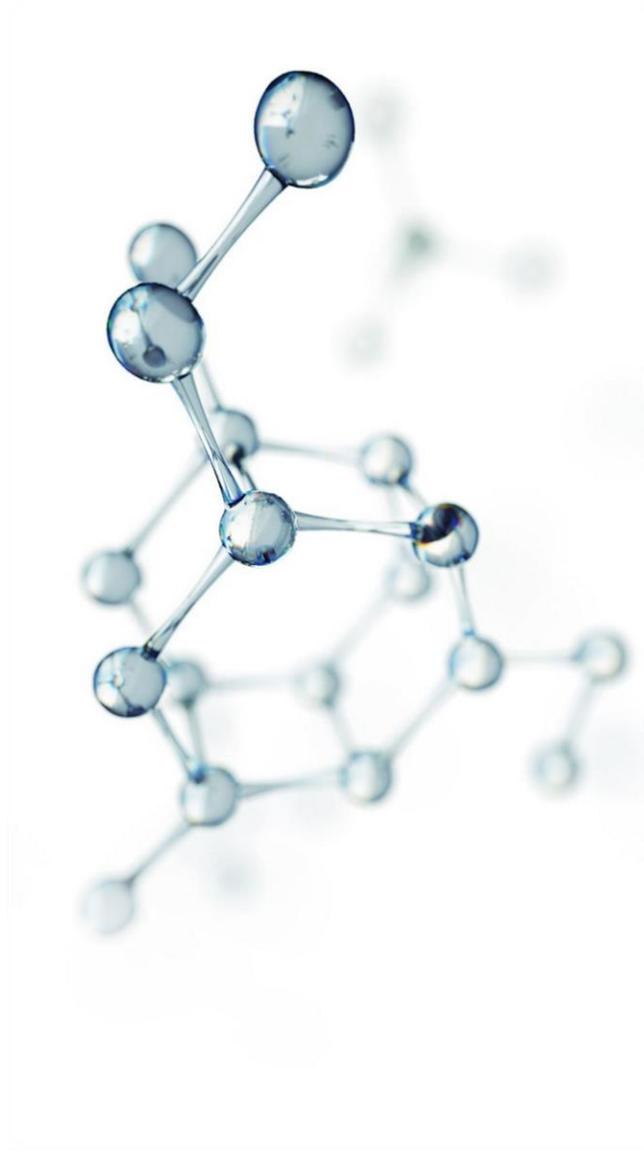


OPERACIONES UNITARIAS 4

PRQ -205



Ing. Hernan G. Peñaranda V.

Sucre - Bolivia

1. INTRODUCCIÓN A RPN

En esta materia los programas serán elaborados para la calculadora HP (en sus modelos 48 y 49) pues es la calculadora que más ampliamente se emplea en las diferentes carreras de la Facultad de Tecnología.

Esta calculadora emplea la notación RPN (Notación Polaca Inversa) siendo sus lenguajes de programación *HP Basic* (o modo Algebraico), *User RPN*, *System RPN* y el *lenguaje de máquina*. La facilidad con la que se puede elaborar un programa, al menos en primera instancia, sigue este mismo orden, es decir es más fácil elaborar un programa en *HP Basic*, le sigue *User RPN*, luego *System RPN* y finalmente el *lenguaje de máquina*, sin embargo, la eficiencia y velocidad de un programa sigue el orden inverso, es decir es mucho más eficiente y rápido un programa elaborado en lenguaje de máquina que uno elaborado en *HP Basic*.

Puesto que la eficiencia (uso de memoria) y la velocidad, son factores muy importantes cuando se trabaja con los limitados recursos de una calculadora, no resulta práctico elaborar programas en *HP Basic* (modo algebraico) pues consumen demasiada memoria y genera programas muy lentos. Por otra parte el *lenguaje de máquina* tampoco es muy práctico pues elaborar programas en el mismo requiere mucho tiempo.

Por estas razones los programas de uso frecuente serán elaborados en *System RPN*, mientras que los programas de uso específico en *User RPN* (aunque en ocasiones también en *System*).

1.1. User y System RPN

User RPN es el lenguaje que viene por defecto en la calculadora, por lo que puede ser programado directamente en la misma. Se trata de un subconjunto de *System RPN* y cuenta con alrededor de 800 comandos, mientras que *System RPN* cuenta con alrededor de 5000 comandos (incluidos los comandos de *User*).

Pero si *User* es un subconjunto de *System* ¿Por qué es más lento? La respuesta se encuentra en la forma en que se ejecutan los comandos de *User*. Todos los comandos de *User* son "seguros" pues antes de ejecutarse verifican que existan los datos y tipos de datos adecuados, de esta manera impiden la escritura de datos erróneos en memoria.

Por el contrario, los comandos de *System* no verifican nada y realizan las operaciones existan o no los datos adecuado. Al no perder tiempo en verificar la existencia de datos y tipos de datos adecuados, estos comandos se ejecutan con mayor rapidez. La verificación de los datos y tipos de datos queda en manos del programador, razón por la cual la programación en *System RPN* requiere de mayor cuidado por parte del programador, pues un error generalmente ocasiona un fallo del sistema y la pérdida de la información almacenada en memoria.

Puesto que *System RPN* no es el lenguaje por defecto de las calculadoras no puede ser programado directamente en las mismas. Para su programación se requieren de herramientas adicionales, las cuales son gratuitas y pueden ser bajadas de www.hpcalc.org.

Para instalar cualquier librería en la calculadora, una vez bajada de hpcalc.org, se la pasa a la calculadora (con el programa de conexión: HP connectivity kit, que también puede ser bajado de hpcalc.org), entonces se almacena en uno de los puertos (0 STO, 1 STO o 2 STO) y se reinicia la calculadora (presionando ON+C) para que la librería sea instalada.

Para la HP48 todas las herramientas necesarias se encuentran en la librería *JAZZ*. Para que *JAZZ* funcione se requiere instalar también la librería de fuentes *UFL3* (que viene conjuntamente la librería *JAZZ*). Puesto que la HP48 no cuenta con un administrador de archivos (un *filer*) es recomendable instalar una de las librerías *FILER48*, la versión más completa (*F48F*) requiere alrededor de 19 kb y la más compacta (*F48U*) alrededor de 10 kb, con las versiones más completa es posible examinar todos los puertos de la calculadora.

Para la HP49 se requieren las librerías *EMACS*, *SDIAG* y *EXTABLE* (las cuales vienen conjuntamente *eMacs*), adicionalmente, en la HP49 es conveniente instalar las librerías *NOSY*, *OT49* y *KEYMAN*. Estas herramientas permiten elaborar o editar programas tanto en *System RPN* como en *lenguaje de máquina* (más propiamente en ensamblador).

La principal limitante de la HP48 es su memoria, pues sólo dispone de 128 kb y la librería *JAZZ* ocupa alrededor de 70 kb. Aunque es posible instalar una versión más limitada de *JAZZ*, de alrededor de 50 kb, sigue siendo una librería muy grande para la memoria disponible en la HP48.

Por otra parte la principal dificultad en la HP49 es su depurador (*DEBUG*), el cual provoca fallos del sistema al depurar algunos comandos de *System* o *Ensamblador*, ello impide en la práctica la depuración de programas escritos en estos lenguajes. Para solucionar este problema se puede instalar la librería *JAZZ* en su versión HP49, pero esta librería sólo puede ser instalada en el puerto 0, lo que consume 70 kb de los 256 kb disponibles en el mismo, adicionalmente se requieren otros 40 kb para la tabla de direcciones de memoria (y 4 Kb para *UFL*), la cual sin embargo puede ser instalada en los puertos 1 o 2. No obstante, aún con *JAZZ* existen todavía algunas limitaciones pues no tiene soporte para los comandos *FLASH*, por lo que dichos comandos no pueden ser depurado paso a paso.

A pesar de las dificultades mencionadas, es conveniente contar con estas librerías (sobre todo en la HP49) principalmente para elaborar programas no muy largos o modificar programas existentes.

En general, sin embargo, es más cómodo y seguro elaborar y probar los programas en una computadora y pasarlos luego a la calculadora. Para ello existen varias herramientas siendo la más completa *DEBUG4X*, que también puede ser bajada de *hpcalc.org*. Emplearemos esta herramienta para elaborar y depurar los programas por lo que deberá bajarla e instalarla en una computadora. *DEBUG4X* incluye emuladores para los modelos HP48 y HP49 (incluido HP49G+), los cuales incorporan todas las funciones de las calculadoras reales lo que nos permite utilizarlas aún sin contar realmente con las mismas (es una buena idea instalar estos emuladores y emplearlos en lugar de la calculadora estándar de Windows).

1.2. Notación RPN

Como se dijo la calculadora HP emplea la Notación Polaca Inversa, este es probablemente el aspecto de la calculadora al que más cuesta acostumbrarse. En RPN se escribe primero él o los datos y luego el comando u operación que actúa sobre los mismos, por ejemplo para sumar 4.5 y 6.7 se escribe:

4.5 6.7 +

En lugar de 4.5+6.7 como ocurre con otras calculadoras (y lenguajes). Para calcular el seno de 45.6 escribimos:

45.6 SIN

En lugar de SIN(45.6). En RPN no existen paréntesis pues no se necesitan. Por ejemplo para calcular el valor de la siguiente expresión:

$$\sqrt{\frac{(3)(4.5)+(7.2)(8^{2.2})}{6.2+9.5e^{4.65}}}$$

Escribimos:

3 4.5 * 7.2 8 2.2 ^ * + 6.2 9.5 4.65 EXP * + / √

En lugar de: $\text{SQRT}((3*4.5+7.2*8^{2.2})/(6.2+9.5*EXP(4.65)))$, como ocurre en otras calculadoras y lenguajes.

Entonces siempre que se escribe una expresión matemática en RPN se deben tomar en cuenta dos aspectos: a) Escribir él o los datos antes del comando u operación y b) No emplear paréntesis (pues no existen).

La única forma de acostumbrarse a la notación RPN es la práctica, mientras más se emplee la calculadora (o su emulador) en la resolución de problemas más fácil y natural resulta. Con la práctica la notación RPN no sólo resulta tan natural como la notación directa sino que en muchos aspectos es inclusive más sencilla.

1.3. La pila (Stack)

Otro aspecto que siempre se debe tener en mente al trabajar con la calculadora HP es la *pila*. La pila es el sector de memoria donde se almacena temporalmente la información. Visualmente la pila se muestra en el sector medio de la pantalla y tiene números al lado izquierdo, tal como se muestra en la figura:



Se denomina pila (o montón) porque la información que se guarda en la misma se va apilando (o amontando). El último dato guardado es siempre el número uno y este es también el primer dato que se puede sacar de la pila, se trata entonces de una pila *LIFO* (Last In First Out).

La pila es muy importante porque todos los comandos y operadores de la HP toman sus datos y guardan sus resultados en la misma. Así por ejemplo si queremos sumar los números 10 y 30 dichos números deben ser colocados primero en la pila, para ello simplemente escribimos 10 30 y presionamos la tecla *Enter* con lo que la pila queda en la forma:



Entonces al presionar la tecla *+*, se ejecuta el operador de la suma el cual toma los dos primeros datos de la pila (los números 30 y 10), los suma y coloca el resultado en la pila, con lo que la misma queda en la forma:

```

DEG XYZ DEC R~ 'X'
[HOME]
7:
6:
5:
4:
3: * + X * X 3. ^ 2. X SQ * + * *
2: [ 3. 4. 5. 6. 4. ]
1:
40.
PPAR PRGM Filer CASDI

```

Cada número identifica un elemento de la pila y existen varios comandos que nos permiten trabajar directamente con la pila, algunos de ellos emplean estos números como referencia, así por ejemplo para hacer una copia de cualquier elemento de la pila se coloca el número o nivel en el que se encuentra seguido del comando PICK, por ejemplo para copiar la lista que se encuentra en el nivel 4 de la pila, escribimos 4 PICK, con lo que la pila queda en la forma.

```

DEG XYZ DEC R~ 'X'
[HOME]
7:
6:
5:
4: * + X * X 3. ^ 2. X SQ * + * *
3: [ 3. 4. 5. 6. 4. ]
2:
1:
40.
PPAR PRGM Filer CASDI

```

Las operaciones y comandos se ejecutan más rápidamente cuando se trabaja directamente con los elementos de la pila (en lugar de trabajar con variables). Por esta razón es frecuente el uso de los comandos que manejan la pila en los programas, sin embargo, no es recomendable abusar de ellos pues los programas se tornan ilegibles y consecuentemente resulta extremadamente difícil modificarlos o corregirlos, actividades que casi siempre se realizan tarde o temprano cuando se elabora un programa.

En la pila se puede guardar cualquier tipo de información, desde un simple número hasta una librería, en realidad sin embargo, lo único que se almacena la pila son direcciones de memoria (cada una de las cuales ocupan 2.5 bytes o lo que es lo mismo 20 bits). Puesto que una dirección de memoria es sólo un número no sería de mucha utilidad ver en pantalla una serie de números, por esta razón el sistema operativo de la calculadora muestra en pantalla el contenido de las direcciones de memoria que se encuentran en la pila, no obstante esto es sólo para efectos de presentación, pues como se dijo en la pila únicamente existen direcciones de memoria.

1.4. Algoritmos

Los algoritmos describen la secuencia lógica de acciones que deben llevarse a cabo para resolver un determinado problema. Existen muchas formas en las que se puede expresar un algoritmo, de ellas emplearemos en esta materia los *diagramas de actividades*, los cuales son parte de UML (Lenguaje de Modelado Unificado) que es el lenguaje gráfico más utilizado actualmente para el modelado de software.

El inicio de un diagrama de actividades se representa con un círculo relleno:



El final del diagrama de actividades se representa con un círculo que contiene un círculo relleno en su interior:



Una acción (o sentencia) se representa con un rectángulo con sus bordes redondeados:



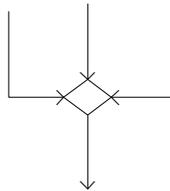
Un flujo, que es el símbolo empleado para unir los elementos del diagrama y señalar la dirección en que deben seguir las acciones, se representa por una flecha continua abierta:



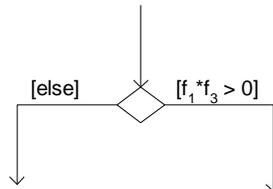
Una unión o bifurcación se representa por un pequeño rombo:



Cuando se emplea como unión llegan dos o más flujos y sale uno.



Cuando se emplea como bifurcación ingresan uno o más flujos y salen dos o más flujos acompañados de alguna condición.



Una condición, tal como se puede observar en la anterior figura, se representa como texto encerrado entre corchetes y siempre está relacionado a algún flujo:



El texto de la condición puede ser una expresión matemática, lógica o literal. Para la condición por defecto se puede emplear `[else]` (caso contrario).

Una actividad representa un conjunto de acciones (o subprograma) que se detalla luego por separado empleando otro diagrama de actividades. Se representa como una acción con un icono de una acción llamando a otra en la parte inferior derecha:



Las notas se emplean para comentar y documentar los diagramas de actividades. Se representan como una hoja con una esquina doblada y asociada a cualquiera de los elementos de un diagrama de actividades mediante una línea discontinua:



Al elaborar un diagrama de actividades se debe tener presente que se trata de un lenguaje y como tal es necesario respetar su sintaxis (es decir los símbolos) pues cada uno de ellos tiene su propio significado y si se cambia, cambia también la lógica del algoritmo. Por ejemplo, para representar un flujo siempre se debe emplear la flecha continua abierta, no una flecha cerrada y rellena: \longrightarrow , una flecha cerrada no rellena: $\longrightarrow\triangleright$, o una flecha discontinua abierta: $\cdots\longrightarrow$ pues cada una de ellas tienen un significado muy diferente (mensaje, herencia y dependencia).

1.5. Programación en User y System RPN

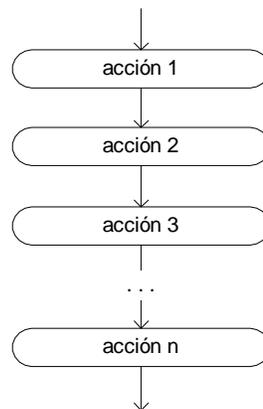
La forma más sencilla y rápida de aprender un nuevo lenguaje de programación es por comparación con otro lenguaje conocido. Puesto que en la Facultad de Tecnología todas las carreras de ingeniería estudian, en Informática I, el lenguaje de programación PASCAL, emplearemos este lenguaje como base para el estudio de los lenguajes *User RPN* y *System RPN*.

Como ya se dijo *User RPN* es un subconjunto de *System RPN*, de manera que todos los comandos y sentencias de *User* puede ser empleados también en *System* (para ello simplemente se antepone una *x* minúscula al nombre del comando *User*). No obstante como los programas elaborados con los comandos y sentencias exclusivos de *System* son más rápidos, son muy raras las ocasiones donde se emplee un comando de *User* en un programa elaborado en *System*.

Al contar tanto *User* como *System* con sentencias y comandos estructurados, abordaremos su estudio describiendo la manera en que se implementan las estructuras estándar de PASCAL en estos lenguajes.

1.5.1. Secuencia

La estructura básica para la elaboración de cualquier programa es la *secuencia*. Una secuencia es una serie de sentencias que se ejecutan una a continuación de otra. El diagrama de actividades de una secuencia se ve aproximadamente como se muestra en la siguiente figura:



Donde las acciones pueden ser acciones simples o acciones compuestas correspondientes a otras estructuras. Recordemos que en PASCAL una secuencia se implementa encerrando las sentencias entre las palabras *begin* y *end* y separando las sentencias con puntos y comas, es decir:

```

begin
  acción 1;
  acción 2;

```

```

acción 3;
...
acción n;
end;

```

Como también se recordará todo programa, procedimiento o función en PASCAL es una secuencia, razón por la cual siempre comienzan con *begin* y terminan con *end*.

En *User RPN* igualmente *todo programa o subprograma es una secuencia*, pero en *User* en lugar de emplear *begin* y *end* se emplean comillas francesas: « », además los comandos se separan con saltos de línea o espacios en blanco:

```

«
acción 1
acción 2
acción 3
...
acción n
»

```

En *System RPN* igualmente *todo programa o subprograma es una secuencia*, pero en lugar de emplear *begin* y *end* se emplean cuatro puntos (::) y punto y coma (;). Al igual que en *User* los comandos se separan con saltos de línea o espacios en blanco:

```

::
acción 1
acción 2
acción 3
...
acción n
;

```

1.5.2. Variables

Al igual que en PASCAL, en *User* y *System RPN* se pueden declarar variables tanto globales como locales, sin embargo, en *User* y *System* la declaración de una variable es una sentencia, por lo que debe formar parte de una secuencia, es decir estar dentro de comillas francesas (para *User*) o entre :: y ; (para *System*).

Como sucede con PASCAL, en *User* y *System* se prefiere el uso de variables locales, pues no interfieren con las variables declaradas en otros programas o módulos, además en *User* y *System RPN* podemos trabajar también con la pila (stack).

A diferencia de PASCAL, donde las variables pueden almacenar diferentes tipos de datos, en *User* y *System RPN* todas las variables almacenan únicamente direcciones de memoria, las cuales como sabemos ocupan 2.5 bytes. Se trata en consecuencia de variables tipo puntero. Afortunadamente la asignación y liberación de la memoria ocupada por estas variables es realizada automáticamente por el sistema operativo, lo que facilita considerablemente su uso.

Contrariamente a lo que se podría pensar, el que las variables almacenen únicamente direcciones de memoria no limita sus posibilidades de almacenamiento, sino que por el contrario las amplía, pues la dirección que se guarda en una variable puede apuntar a cualquier tipo de dato. En consecuencia desde el punto de vista práctico en una variable de *User* o *System RPN* se puede guardar cualquier tipo de información, desde un simple número hasta un programa completo.

Lo anterior es posible debido a la forma en que maneja la información la HP. Todo dato en la HP, desde un número hasta un programa, es almacenado en memoria empleando una estructura denominada *objeto*. Un *objeto* en la HP está conformado por un prólogo (de 20 bits) que identifica al objeto, seguido por el cuerpo del objeto (que son los datos del objeto propiamente) y terminando con un epílogo que puede ser nulo o una dirección de memoria (de 20 bits). De esa manera para acceder a cualquier tipo de dato sólo es necesario conocer su dirección, pues en esa dirección se encuentra toda la información necesaria para trabajar con el mismo.

Debido a esta característica de las variables, todos los programas o subprogramas en la HP, al igual que cualquier otro tipo de dato, son guardados en variables simples (o en librerías).

En cuanto a los nombres, las variables pueden tener nombres conformados por cualquier combinación de letras y números. Se permiten también algunos símbolos especiales como el de subrayado ().

Algo que se debe tomar muy en cuenta al momento de programar y utilizar programas es que *User* y *System RPN* (a diferencia de PASCAL) diferencian entre mayúsculas y minúsculas, así *Var1* es una variable diferente a *VAR1*, *var1*, *vAr1*, *vaR1*, *VAR1*, *vAR1*, etc.

1.5.2.1. Variables locales

1.5.2.1.1. User RPN

Para declarar variables locales en *User RPN* se emplea el carácter correspondiente a la flecha derecha: \rightarrow . Los datos que se asignan a las variables se escriben a la izquierda de la flecha (o deben encontrarse en la pila) y los nombres de las variables a la derecha. Como ya se dijo, en *User* y en *System* la declaración de variables es una sentencia, por lo que debe encontrarse dentro de una secuencia.

Por ejemplo para crear 5 variables locales: *a*, *b*, *c*, *d* y *e*, con los valores 10, 20, 30, 40 y 50, escribimos lo siguiente:

```
10 20 30 40 50  $\rightarrow$  a b c d e
```

De esta manera "a" apuntará al número 10, "b" al número 20, "c" al 30, "d" al 40 y "e" al 50.

Para crear 4 variables locales: *var1*, *var2*, *var3* y *var4*, donde los valores de *var1* y *var2* se encuentran en la pila, el valor de *var3* es [1 2 3] y el de *var4* { 4.5 6.7 8.9 }, escribimos:

```
[ 1 2 3 ] { 4.5 6.7 8.9 }  $\rightarrow$  var1 var2 var3 var4
```

Como se puede observar en este caso no sólo existen datos para dos de las variables (*var3* y *var4*) los dos datos faltantes deben encontrarse en la pila. En realidad cuando en un programa se escribe un dato (de cualquier tipo) el mismo (más propiamente su dirección) es colocado en la pila. Por lo tanto en los anteriores ejemplos los valores que se encuentran a la izquierda de la flecha son colocados primero en la pila y luego recién son asignados a las variables respectivas. Por consiguiente en todos los casos los datos que se asignan a las variables deben encontrarse de una u otra forma en la pila.

Para familiarizarnos con el uso de las variables locales así como la elaboración de programas, crearemos algunos de ellos. Con ese fin debe instalar primero el programa DEBUG4X. Una vez instalado, en la listas de programas (en Inicio) encontrará la carpeta: "HP 48 & 49 Development Kit" y dentro de esta carpeta los emuladores: emu48 (para la HP48), emu49 (para la HP49G) y emu49G+ (para la HP49G+). Si cuenta con una calculadora real elija el emula-

dor correspondiente a su modelo, caso contrario elija emu49. Entonces aparecerá en pantalla la calculadora y podrá trabajar con ella de la misma forma que con la calculadora real.

Por defecto el emulador funciona a la misma velocidad que la calculadora real, si desea trabajar a la velocidad de la computadora entonces debe ir al menú *File* -> *Settings* y en la ventana que aparece desmarcar la opción correspondiente a *Authentic Calculator Speed*. Es conveniente también que estén desmarcadas las opciones: *Show Load Object Warning* y *Always Show KML Compilation Result* y que por el contrario estén marcadas las opciones: *Automatically Save Files* y *Automatically Save Files on Exit*.

Toda información que se guarda en la memoria de la calculadora (en el emulador) queda almacenada en disco duro en archivos que tienen la extensión .E49 (o .E48 para la HP48) y que por defecto tienen el nombre *Default.E49* (o *Default.E48*). Se pueden guardar tantas copias de la memoria de la calculadora como se quiera. Para ello debe ir al menú *File* -> *Save As...*, ingresar al directorio donde desea guardar la copia, escribir un nombre para la copia (por ejemplo la sigla de la materia) y hacer clic en guardar. Cuando quiera trabajar con una de las copias guardadas debe ir al menú *File* -> *Open*, ingresar al directorio donde guardó la copia y hacer doble clic en la misma.

Estas copias (que ocupan 1 MB) son el modo más cómodo y seguro de llevar la información del emulador de una computadora a otra y es la forma que deben emplear para presentar los trabajos que elaboren en *User RPN*. Estas copias no sólo sirven para transportar la información, sino que constituyen también un respaldo, pues en ocasiones, al igual que sucede con la calculadora real, la memoria del emulador puede quedar corrompida. Por ello es conveniente tener al menos dos copias de la memoria de la calculadora.

En cuanto a su manejo, todos los comandos y funciones que se ven en el teclado de la calculadora (tanto en el emulador como en la calculadora real) pueden ser utilizados de la siguiente manera: Si el nombre del comando está sobre el teclado, es suficiente presionar la tecla, así para calcular la raíz cuadrada de 2 se escribe el número 2 y luego se presiona la tecla que tiene el símbolo de la raíz cuadrada, obteniéndose: 1.41421356237 (haga la prueba).

Si el nombre del comando o función está en azul (verde para la HP48) o rojo, entonces se debe pulsar primero la tecla de cambio azul o roja:



Denominadas teclas de cambio izquierda y derecha respectivamente, luego se pulsa la tecla donde se encuentra el comando. Así por ejemplo para calcular el logaritmo natural de 4.56, se escribe el número 4.56, se pulsa la tecla de cambio derecha y luego la tecla donde se encuentra la función correspondiente la logaritmo natural (LN), obteniéndose: 1.51732262353 (haga la prueba).

Para escribir las letras que se encuentran en el teclado se pulsa primero la tecla ALPHA:



Y luego la tecla donde se encuentra la letra. Por defecto, el pulsar una vez la tecla ALPHA permite escribir sólo una letra, para escribir más de una letra se debe pulsar la tecla ALPHA dos veces seguidas, con lo que el teclado queda en el modo alfanumérico y en este modo se pueden escribir letras hasta que se pulse otra vez la tecla ALPHA.

Todos los comandos y funciones que se ven en el teclado, así como aquellos que no están en el teclado, pueden ser utilizados también escribiéndolos directamente en la pila y pulsando luego la tecla ENTER:



Así por ejemplo para calcular el logaritmo natural de 4.56 podemos escribir en la pila:

4.56 LN

Y luego pulsar ENTER (haga la prueba). Si desea que el teclado quede en el modo alfanumérico presionando una sola vez la tecla ALPHA, se debe activar la bandera N° 60. Para activar (o desactivar) cualquier bandera se ingresar a MODE, luego a FLAGS, se busca el número de la bandera y se la activa (o desactiva) pulsando la opción CHK. Alternativamente se puede activar cualquier bandera escribiendo su número (como un número negativo) y a continuación el comando SF (Set Flag), así por ejemplo para activar la bandera N° 60 podemos escribir lo siguiente:

-60 SF

Como de costumbre se debe pulsar luego ENTER. Para desactivar cualquier bandera se escribe su número (como número negativo) y a continuación el comando CF (Clear Flag), así para desactivar la bandera N° 60 podemos escribir lo siguiente:

-60 CF

Los programas en *User RPN* se escriben directamente en la pila, para ello colocamos primero las comillas francesas (« ») y en su interior escribimos las instrucciones del programa.

Como primer ejemplo elaboremos un programa que devuelva el valor de la función:

$$f(x) = x^3 + 2x^2 + 3x + 4$$

Puesto que "x" es una variable (y no una constante) su valor debe encontrarse en la pila y en consecuencia sólo debemos declarar la variable "x" (→ x). Luego con la variable se realizan las operaciones correspondientes a la función (x 3 ^ 2 x 2 ^ * + 3 x * + 4 +). Sin embargo, antes de escribir estas operaciones debemos abrir una nueva secuencia, pues en User RPN siempre que se declara una o más variables es necesario iniciar otra secuencia. Entonces el programa queda en la forma:

```
« → x
  « x 3 ^ 2 x 2 ^ * + 3 x * + 4 + »
»
```

Escriba este programa en el emulador y luego cree el directorio 'PRAC1' escribiendo en la pila lo siguiente:

'PRAC1' CRDIR

Ingrese al directorio creado (pulsando la tecla correspondiente al mismo) y guarde el programa en la variable global 'F1', escribiendo:

'F1' STO

Por supuesto en lugar de escribir STO (y luego pulsar ENTER) puede pulsar también la tecla correspondiente a dicho comando. Con ello el programa queda guardado en la variable global 'F1' dentro del directorio 'PRAC1'.

Ahora con el programa creado puede calcular valores de la función para diferentes valores de "x". Así por ejemplo para calcular el valor de la función para x=1, escribimos:

1 F1

Por supuesto en lugar de escribir el nombre de programa puede pulsar la tecla correspondiente al programa. Entonces obtendrá como resultado el número 10. Para calcular el valor de la función para x=4.56, escribimos:

4.56 F1

Con lo que obtenemos el resultado: 154.086016.

Supongamos ahora que queremos modificar este programa pues tenemos la función:

$$f(x) = x^3 + 2x^2 + 3x - 4$$

Que como vemos sólo difiere de la anterior en el último signo. Para modificar un programa debemos colocarlo primero en la pila escribiendo:

'F1' RCL

Como ya se ha señalado reiteradamente, en lugar de escribir el comando RCL (y luego pulsar ENTER), se puede pulsar la tecla equivalente. Alternativamente, se puede pulsar la tecla de cambio derecha y luego la tecla del programa. Siguiendo cualquiera de estos procedimientos tendrá el programa en la pila.

Una vez que se tiene el programa en la pila, se puede editar el mismo pulsando la tecla de cursor hacia abajo:



Entonces se puede corregir o modificar el programa, en este caso se cambia el último signo de "+" a "-", quedando el programa en la forma:

```
« → x
  « x 3 ^ 2 x 2 ^ * + 3 x * + 4 - »
»
```

Guarde este programa con el nombre "F2" ('F2' STO). Entonces calcule el valor de la función para x=2.12 (2.12 F2), con lo que deberá obtener el resultado: 20.876928.

Con frecuencia es conveniente hacer correr un programa paso a paso ya sea para corregir errores como para analizar la forma en que funciona el mismo. Cada vez que se elabora un nuevo programa es una buena práctica hacerlo correr por primera vez paso a paso, de esa manera se detectan los errores y es posible optimizar el código.

Para hacer correr un programa paso a paso, se coloca el programa en la pila (o se escribe el nombre del programa entre apóstrofes), entonces se ingresa al menú PRG, dentro al submenú RUN y dentro se elige la opción DBUG.

Por ejemplo para hacer correr paso a paso el programa F1 para un valor de x=1.5, escribimos 1.5 'F1' y seguimos la secuencia PRG -> RUN -> DBUG (haga la prueba). Entonces en la parte superior de la pantalla aparecerá el mensaje HLT (Halt), indicando que la ejecución del programa ha sido detenida. Para continuar la ejecución con el siguiente comando (o paso) se pulsa la tecla correspondiente a la opción SST o a SST↓. Tanto SST como SST↓ ejecutan el siguiente comando, pero si dicho comando es a su vez un programa SST↓ salta al mismo y la depuración continúa al interior de dicho programa. SST

por el contrario, no ingresa a ningún otro programa y trata los programas (o subprogramas) como si fueran comandos simples.

En nuestro pequeño programa no existen subprogramas, razón por la cual da lo mismo si se emplea SST o SST↓. A medida que vaya depurando el programa (pulsando SST o SST↓), podrá ver en la parte superior de la pantalla el nombre del comando o variable que esté ejecutando, mientras que en la parte inferior de podrá ver como el programa va colocando o quitando valores de la pila. En cualquier momento puede detener la depuración de un programa pulsando la tecla correspondiente a la opción KILL (o escribiendo KILL y pulsando ENTER). Igualmente puede hacer que el programa continúe su ejecución normal pulsando la tecla CONT (o escribiendo CONT y pulsando ENTER).

Como segundo ejemplo elaboraremos un programa que permita evaluar la función:

$$f(k) = \theta_c - \frac{T_0}{k} \left(1 - \cos \left(\sqrt{\frac{k}{I_0}} t \right) \right)$$

Donde:

$$T_0 = T_c + \theta_k k$$

$$T_c = 0.625$$

$$\theta_c = 2.97$$

$$\theta_k = 3.27$$

$$t = 0.0191$$

$$I_0 = 0.0006$$

Como vemos en esta función existen varias constantes. Desde el punto de vista de *User RPN* estas constantes son variables locales estáticas con valores iniciales definidos. Se tiene también una expresión intermedia cuyo resultado se guarda en T_0 , esta es igualmente una variable local estática pero como el valor de la misma se calcula, puede ser inicializada con un valor arbitrario cualquiera, normalmente 0. Finalmente tenemos la variable: k , que no tiene un valor definido, por lo tanto su valor deberá estar en la pila.

En consecuencia la declaración de variables para esta función es: 0.625 2.97 3.27 0.0191 0.0006 0 → k Tc Thc Thk t Io To; donde el cero se coloca para la variable To. Una vez declaradas las variables se evalúa la expresión intermedia (Tc Thk k * +), se guarda ese valor en To ('To' STO) y con este valor se calcula el valor de la función (Thc To k / 1 k Io / √ t * COS - * -). El programa resultante es:

```
« 0.625 2.97 3.27 0.0191 0.0006 0
  → k Tc Thc Thk t Io To
  « Tc Thk k * + 'To' STO
    Thc To k / 1 k Io / √ t * COS - * -
  »
»
```

Guarde este programa en el directorio PRAC1 con el nombre F3 ('F3' STO). Para evaluar esta función la unidad de medida de los ángulos deben estar en radianes. Puede cambiar la unidad de medida ingresando a MODE, bajando a Angle Measure y seleccionando (con CHOOS) *radians*. Alternativamente puede escribir RAD y pulsar ENTER.

Ahora escriba en la pila 3.36 y haga correr el programa paso a paso. Si encuentra algún error, detenga la depuración (con KILL), corrija el error (editando el programa), vuelva a guardar el programa (con 'F3' STO) y vuelva

a hacer correr el programa paso a paso hasta que el mismo finalice sin errores. Si todo va bien al final deberá tener el resultado: .00132605009.

1.5.2.1.2. *System RPN*

En *System RPN*, existen dos tipos de variables locales: las variables locales sin nombre (NULLLAM) y las variables locales con nombre (LAM). Las variables locales sin nombre, son referenciadas por el orden en que son creadas, son más rápidas que las variables locales con nombre, por lo que es más eficiente trabajar con ellas, sin embargo, no se aconseja su uso cuando el programa llama a otros programas (o subprogramas) que a su vez crean otras variables locales. Cuando ello sucede las posiciones relativas de las variables en el programa son afectadas, generando errores y dificultando considerablemente la depuración del programa.

Las variables locales con nombres (LAM) se declaran de forma similar a las variables locales de *User RPN*. Si tienen valores iniciales se colocan primero dichos valores (caso contrario los valores deberán estar en la pila), luego se escribe la lista de variables entre llaves (precedidas por la palabra LAM) y se crean las variables con el comando BIND. A diferencia de *User*, las variables locales de *System* deben ser eliminadas manualmente por el programador (cuando ya no son necesarias), con el comando ABND. La forma general en la que se declaran variables locales con nombre en *System* es:

```
Valores iniciales { lista de variables } BIND
(sentencias donde se utilizan las variables)
ABND
```

Como ya se mencionó anteriormente, los comandos de *System* no verifican nada, por consiguiente el comando BIND crea las variables existan o no los datos para las mismas, sin embargo, si no existen se escribirá en sectores erróneos de la memoria provocando un fallo del sistema. Por ello en *System* se debe verificar previamente que exista el número de datos adecuado, para ello se emplean los comandos CK0NOLASTWD, CK1NOLASTWD, CK2NOLASTWD, CK3NOLASTWD, CK4NOLASTWD, CK5NOLASTWD y CKNNOLASTWD (o CK0, CK1, CK2, CK3, CK4, CK5 y CKN si se trata de una librería), los primeros 6 verifican que existan 0, 1, 2, 3, 4 o 5 datos en la pila, el último permite verificar la existencia de un número arbitrario de datos y debe estar precedido por un número entero binario (BINT).

Los números enteros binarios (BINT) son de uso frecuente en *System* debido a que almacenan 20 bits y como sabemos ese es el tamaño de una dirección de memoria. Los BINT de uso más frecuente (hasta el 64 en la HP48 y hasta el 130 en la HP49) están predefinidos y pueden ser colocados en la pila escribiendo el número en inglés (o BINT seguido del número en la HP49). Así para colocar en la pila el entero binario 25 escribimos:

```
TWENTYFIVE
```

O alternativamente BINT25 en la HP49. Otros BINT no predefinidos pueden ser colocados en la pila escribiendo el símbolo "#" seguido del número en notación hexadecimal, así por ejemplo para colocar en la pila el entero binario 205 escribimos:

```
# CD
```

Para verificar la existencia de 30 datos en la pila escribimos:

```
THIRTY CKNNOLASTWD
```

O THIRTY CKN si estamos trabajando en una librería.

Por otra parte es importante comprobar que los datos sean del tipo correcto, pues a diferencia de *User*, todas las funciones y comandos en *System* son específicos para un determinado tipo de dato, así por ejemplo para sumar dos números reales el comando es `%+`, para sumar dos números enteros binarios es `#+`, para sumar (o concatenar) dos cadenas (strings) es `&$`, mientras que en *User* para todos ellos es `+`. En *System* la mayoría de los comandos, operadores y funciones que trabajan con números enteros binarios (BINT) están precedidos por el signo `#`, los que trabajan con números reales por `%`, los que trabajan con números reales de doble precisión (que no existen en *User*) por `%%`, los que trabajan con números complejos por `C%` y los que trabajan con números complejos de doble precisión por `C%%`.

Para comprobar los tipos de datos se emplea el comando `CK&DISPATCH0` o `CK&DISPATCH1`. Ambos tienen el siguiente formato general:

```
CK&DISPATCH1
#tipos1 acción1
#tipos2 acción2
. . .
#tiposn acciónn
;
```

Donde `#tipos1`, `#tipos2`, etc., son números enteros binarios (BINT) que tienen el formato `# nnnnn`, donde cada `n` es un número hexadecimal que identifica a un tipo de dato en particular. Por lo tanto cada BINT puede verificar los tipos de hasta 5 datos de la pila (menos para aquellos tipos de datos que se identifican con dos números hexadecimales). La forma en que funcionan estos comandos es la siguiente: primero comparan los tipos de datos de la pila con el BINT `#tipos1` y si son iguales ejecutan la `acción1`, luego de lo cual el programa continúa después del `;`, caso contrario comparan con el BINT `#tipos2` y si son iguales ejecuta la `acción2`, luego de lo cual el programa continúa después del `;` y así sucesivamente. Si ninguno de los BINTS concuerda con los tipos de datos de la pila se genera el error "Bad Argument Type" (tipo de argumento erróneo).

Como se mencionó cada tipo de dato se identifica por un número. Los números para los datos de uso más frecuente son los siguientes:

- 0 : Cualquier tipo de dato
- 1 : Número real
- 2 : Número complejo
- 3 : Cadena (String)
- 4 : Vector o matriz (Array)
- 5 : Lista
- 6 : Variable global (ID)
- 7 : Variable local (LAM)
- 8 : Programa
- 9 : Simbólico
- C : Objeto gráfico (GROB)
- D : Objeto etiquetado (TAG)
- E : Objeto de unidad
- 1F : Entero binario (BINT)
- 2F : Directorio
- 3F : Real extendido (%%)
- 4F : Complejo extendido (C%%)

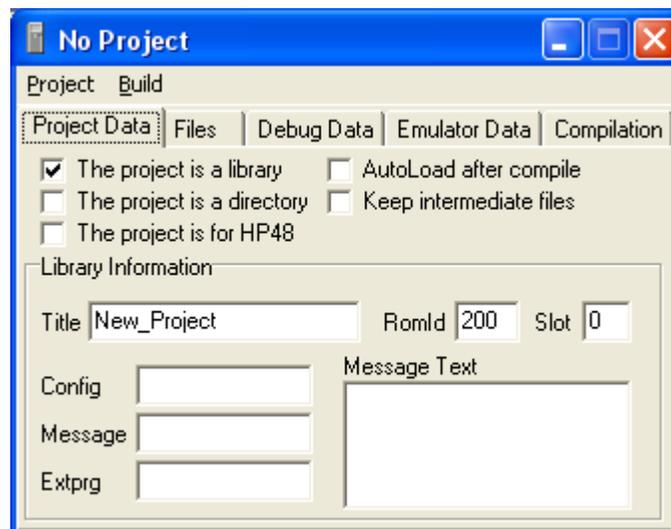
Así por ejemplo para verificar que en la pila existe un programa en el quinto nivel, una lista en el cuarto, un vector o matriz en el tercero, una cadena en el segundo y un número real en el primero, el número entero binario sería: `# 85431`.

La diferencia entre CK&DISPATCH0 y CK&DISPATCH1 es que el segundo si no encuentra concordancia con ninguno de los BINT, convierte los enteros largos (ZINT) y los objetos etiquetados (TAG) a números reales y realiza una segunda pasada.

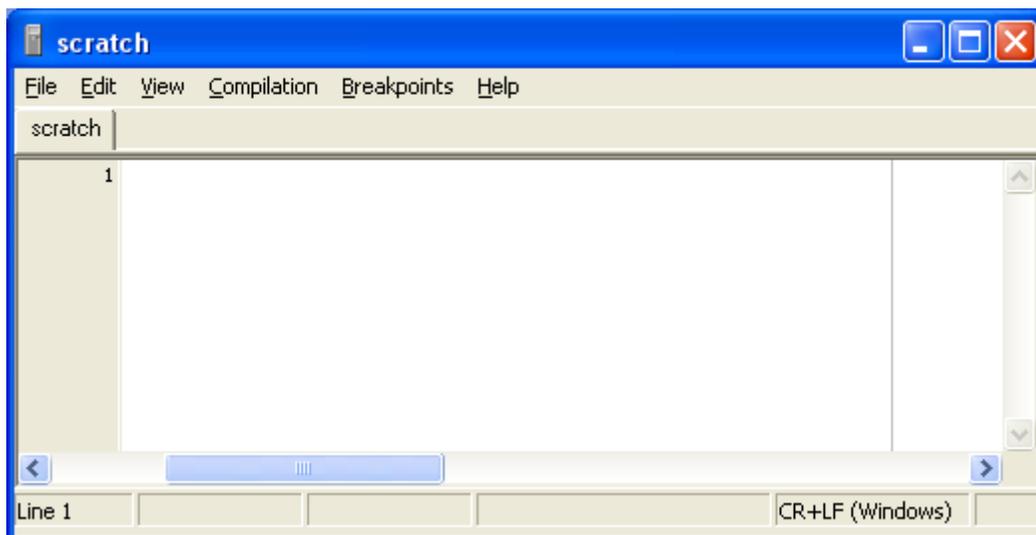
Los programas en *System RPN* serán elaborados con Debug4x, que como los emuladores, se encuentra dentro de la carpeta "HP 48 & 49 Development Kit". Al ingresar al programa tendrá una ventana como la siguiente:



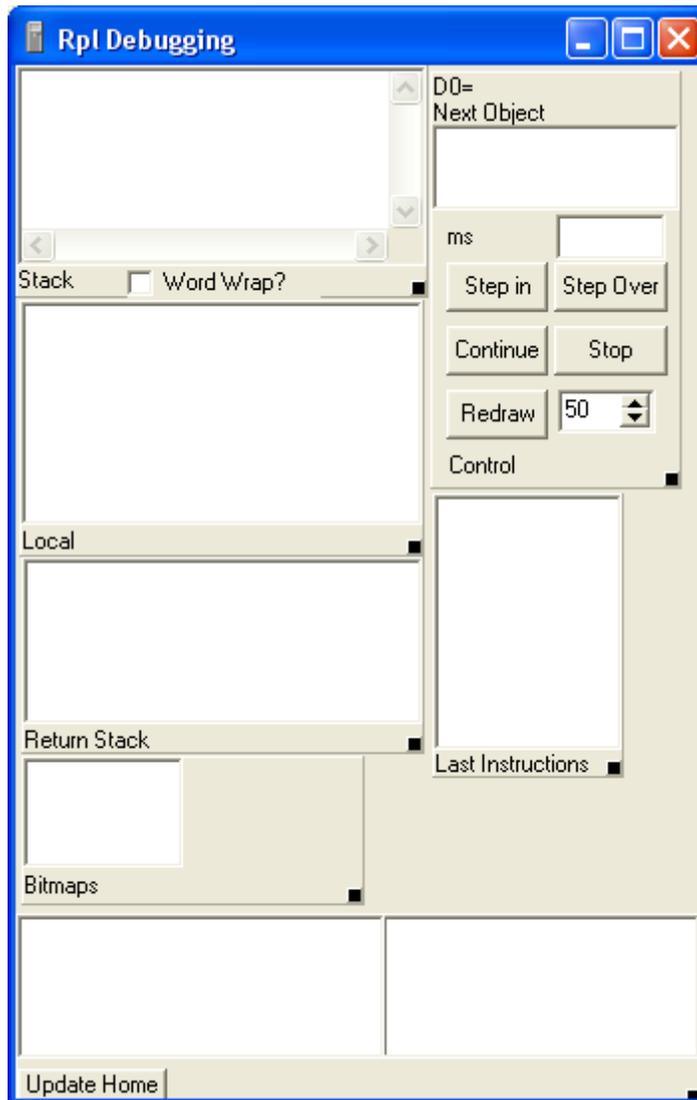
Si no está visible la ventana de proyectos vaya al menú *View -> Project*, con lo que deberá aparecer la siguiente ventana:



De la misma manera, si no está visible la ventana de edición vaya al menú *View -> Editor*, con lo que aparecerá una ventana similar a la siguiente:



Si no está abierta la ventana del depurador vaya al menú DebugView -> Debugger RPL, con lo que aparecerá una ventana similar a la siguiente:



Finalmente para ver el emulador de la calculadora vaya al menú *Emulator* -> *Emulate*.

Como primer ejemplo elaboraremos un programa en *System RPN* para evaluar la función que programamos en *User*:

$$f(x) = x^3 + 2x^2 + 3x + 4$$

En la ventana de proyectos vaya al menú *Project* -> *New Project*. En la ventana que aparece cree en el directorio raíz del disco duro el directorio RPN, dentro de este el directorio PRAC1 y dentro guarde el proyecto con el nombre *Project01*.

Como en este caso crearemos un programa (no una librería) desmarque la opción "The project es a library" y asegúrese que esté marcada la opción "Autoload after compile" para que el programa, una vez compilado, aparezca en el emulador. Si está utilizando una calculadora HP48, entonces marque también la opción "The project is for HP48". En *Path* escriba: "PRAC1 f1", que es el nombre del directorio y de la variable donde se guardará el programa en el emulador, observe que "f1" está en minúsculas.

Ahora, en la ventana de proyectos haga clic en la pestaña "Files" y vaya al menú *Project -> New Source File*. En la ventana que aparece escriba como nombre del archivo "f1" y haga clic en el botón "Guardar".

Entonces en la ventana de edición aparece un encabezado y se puede comenzar a escribir el programa debajo del mismo. Las consideraciones para este programa son esencialmente las mismas que para su equivalente en *User*, excepto que ahora se debe verificar el número de datos (1) y el tipo de los mismos. Puesto que las funciones normalmente se evalúan para números reales, este será el tipo de dato que reciba el programa, pero para permitir además números enteros (ZINT) y objetos etiquetados (TAG) se verificará el tipo con CK&DISPATCH1. Tomando en cuentas estas consideraciones se ha escrito el siguiente programa (transcriba el mismo en la ventana de edición):

```
::
  CK1NOLASTWD
  CK&DISPATCH1
  ONE
  ::
    { LAM x }
  BIND
  LAM x %3 %^ %2 LAM x %2 %^ %* %+ %3 LAM x %* %+ %4 %+
  ABND
;
;
```

Una vez escrito el programa guárdelo (*File -> Save*) y compile el proyecto (*Compilation -> Build Project*), entonces, al cabo de algunos segundos aparecerá en el emulador, en el directorio "PRAC1", el archivo "f1". Para hacer correr este programa paso a paso, que es lo recomendable la primera vez que se hace correr un programa, haga clic a la izquierda del número que aparece en la línea donde se encuentra { LAM x }, entonces verá que aparece un punto rojo, este es un punto de ruptura (breakpoint) y se coloca de esta manera en él o los lugares donde uno quiere detener la ejecución de un programa.

Ahora en el emulador escriba el número 1 y pulse la tecla correspondiente al programa "f1". Entonces el programa comenzará a correr y se detendrá en la línea donde se puso el punto de ruptura (quedando remarcada con un fondo celeste). Para continuar la ejecución paso a paso haga clic en el botón "Step over" (equivalente a SST) en la ventana de depuración, si quiere ingresar a los subprogramas deberá hacer clic en "Step in" (equivalente a SST↓). A medida que vaya ejecutando el programa paso a paso (haciendo clic en "Step over" o "Step in") podrá ver el contenido de la pila (en Stack), las variables locales (en Local), el siguiente comando (en Next Object) e información adicional en los otros recuadros. Finalmente cuando se ejecute el comando ABND termine el programa haciendo clic en el botón "Continue". Si no ha cometido ningún error obtendrá (en el emulador) el resultado 10, que por supuesto es el mismo resultado calculado con el programa equivalente en *User*.

Una vez probado el programa, elimine el punto de ruptura (volviendo a hacer clic en el mismo) y haga correr el programa para $x=4.56$, escribiendo en el emulador:

4.56 f1

Con lo que obtendrá el resultado: 154.086016.

El modificar o corregir un programa con *Debug4x* es incluso más sencillo que en *User*. Por ejemplo si se desea modificar el proyecto (sin guardar el proyecto original) para evaluar la función:

$$f(x) = x^3 + 2x^2 + 3x - 4$$

Que difiere de la anterior sólo en el último signo, debe proceder de la siguiente manera: en la ventana de proyectos (en Path) cambie "PRAC1 f1" por "PRAC1 f2" de manera que el nuevo programa se guarde en la variable "f2". En la ventana de edición modifique el programa (cambie el último signo + por -) y compile el proyecto (*Compilation -> Build project*). Entonces en el emulador aparecerá un nuevo programa "f2" que se el programa modificado. Al hacer correr el programa con $x=2.12$ (*2.12 f2*), obtendrá el resultado 20.876928.

Como segundo ejemplo elaboremos un programa en *System RPN* para evaluar la función que programamos como segundo ejemplo en *User*:

$$f(k) = \theta_c - \frac{T_0}{k} \left(1 - \cos \left(\sqrt{\frac{k}{I_0}} t \right) \right)$$

Donde:

$$T_0 = T_c + \theta_k k$$

$$T_c = 0.625$$

$$\theta_c = 2.97$$

$$\theta_k = 3.27$$

$$t = 0.0191$$

$$I_0 = 0.0006$$

Para crear este programa elaboraremos un nuevo proyecto (*Project -> New Project*) con el nombre *Proyecto2*. Recuerde desmarcar la opción "*The project is a library*" y marcar la opción "*Autoload after compile*". El programa será guardado con el nombre "f3" (en minúscula), por lo tanto en *Path*, en la ventana de proyectos escriba: "PRAC f3" y en la pestaña "*Files*" añada el archivo "F3.S" (*Project -> New Source File*).

Las consideraciones para este programa son las mismas que en *User*, con el cuidado adicional de controlar el número (1) y tipo de datos (real). El programa, que debe ser escrito en la ventana de edición, es el siguiente:

```
::
CK1NOLASTWD
CK&DISPATCH1
ONE
::
% 0.625 % 2.97 % 3.27 % 0.0191 % 0.0006 %0
{ LAM k LAM Tc LAM Thc LAM Thk LAM t LAM Io LAM To }
BIND
LAM Tc LAM Thk LAM k %* %+
' LAM To STO
LAM Thc LAM To LAM k %/ %1 LAM k LAM Io %/
%SQRT LAM t %* %COS %- %* %-
ABND
;
;
```

Observe que todos los números, operadores y funciones están precedidos por el símbolo "%" porque se trabaja con números reales (la raíz cuadrada se calcula con %SQRT). Observe también que para guardar el resultado en la variable "To", se escribe un apóstrofe delante de la variable, esto permite colocar la variable (y no su contenido) en la pila. Esto es algo que se debe hacer siempre que sea necesario colocar un objeto en la pila sin evaluarlo.

Guarde el programa (*File -> Save All*), compile el proyecto (*Compilation -> Build project*) y si existen errores corríjalos, luego coloque un punto de ruptura en la línea donde se encuentran las constantes. En el emulador escriba el número 3.36 y haga correr el programa paso a paso. Si todo está correcto, al final obtendrá el resultado: .00132605009.

Finalmente quite el punto de ruptura y haga correr el programa con $k=3.5$ (3.5 f3) con lo que deberá obtener el resultado: -.09310507507.

Como se dijo, en *System* se pueden declarar también variables locales sin nombre (NULLLAM). Estas variables se declaran de manera similar a las variables locales con nombre, excepto que todas las variables tienen el nombre "NULLLAM". Así por ejemplo para declarar 7 variables locales sin nombre (todas con valores iniciales) es escribe:

```
%1 %2 %3 %4 %5 %6 %7
{ NULLLAM NULLLAM NULLLAM NULLLAM NULLLAM NULLLAM NULLLAM }
BIND
```

Se puede evitar escribir "n" veces NULLLAM, si se coloca un "NULLLAM" en la pila con " ' NULLLAM " y se duplica n veces este valor escribiendo el número como entero binario seguido del comando "NDUPN", luego se crean las variables con el comando DOBIND (no BIND porque no se trata de una lista). Así las 7 variables anteriores pueden ser creadas de la siguiente manera:

```
%1 %2 %3 %4 %5 %6 %7
' NULLLAM SEVEN NDUPN
DOBIND
```

En *System RPN*, muchos comandos son sólo combinaciones de otros y su único fin es el ahorro de memoria. Por ejemplo para crear una variable local se puede emplear el comando "1LAMBIND" en lugar de "{ NULLLAM } BIND" o "'NULLLAM ONE DOBIND".

Una vez creada una variable local se accede a la misma en base al orden de creación, por ejemplo para acceder a la primera variable local se escribe 1GETLAM, para la cuarta 4GETLAM y de la misma manera para cualquier otra variable. Para guardar nuevos valores en las variables locales se emplea el mismo procedimiento, así para guardar un nuevo valor en la primera variable se escribe 1PUTLAM, para guardar uno en la cuarta 4PUTLAM, etc.

Las variables locales sin nombre se enlazan a sus valores de manera diferente a las variables locales con nombre, la primera variable local (1GETLAM) toma el primer valor de la pila, la segunda (2GETLAM) el segundo y así sucesivamente. Por lo tanto, para el ejemplo 1GETLAM tiene el número %7, 2GETLAM el número %6, 3GETLAM el número %5 y así sucesivamente.

Como primer ejemplo del uso de variables locales sin nombre volveremos a programar la función: $f(x)=x^3+2x^2+3x+4$.

Para ello creamos un nuevo proyecto con el nombre *Proyecto4* (*Project -> New Project*), recordando desmarcar la opción "The project is a library" y marcar la opción "Autoload after compile". El programa se llamará *fun1*, por lo que en *Path* se escribe: "PRAC1 fun1". En *Files* añade el archivo *FUN1.S* (*Project -> New Source File*), recuerde también visualizar el emulador (*Emulator -> Emulate*). Entonces en la ventana de edición escriba el siguiente programa:

```
::
CK1NOLASTWD
CK&DISPATCH1
ONE
::
```

```

1LAMBIND
1GETLAM %3 %^ %2 1GETLAM %2 %^ %* %+ %3 1GETLAM %* %+ %4 %+
ABND

```

```
;
```

Guarde las modificaciones (*File -> Save All*), compile el proyecto (*Compilation -> Build Project*), coloque un punto de ruptura en la línea donde se encuentra el comando 1LAMBIND y haga correr el programa paso a paso con $x=4.56$ (*4.56 fun1*). Si todo va bien obtendrá el resultado 154.086016.

Como segundo ejemplo del uso de variables locales sin nombre volveremos a programar la función:

$$f(k) = \theta_c - \frac{T_0}{k} \left(1 - \cos \left(\sqrt{\frac{k}{I_0}} t \right) \right)$$

Donde:

$$T_0 = T_c + \theta_k k$$

$$T_c = 0.625$$

$$\theta_c = 2.97$$

$$\theta_k = 3.27$$

$$t = 0.0191$$

$$I_0 = 0.0006$$

Para ello creamos un nuevo proyecto con el nombre *Proyecto5 (Project -> New Project)*, recordando desmarcar la opción "The project is a library" y marcar la opción "Autoload after compile". El programa se llamará *fun3*, por lo que en *Path* se escribe: "PRAC1 fun3". En *Files* añade el archivo *FUN3.S (Project -> New Source File)*, recuerde también visualizar el emulador (*Emulator -> Emulate*). Entonces en la ventana de edición escriba el siguiente programa:

```

::
CK1NOLASTWD
CK&DISPATCH1
ONE
:: ( k=7 Tc=6 Thc=5 Thk=4 t=3 Io=2 To=1 )
% 0.625 % 2.97 % 3.27 % 0.0191 % 0.0006 %0
' NULLLAM SEVEN NDUPN
DOBIND
6GETLAM 4GETLAM 7GETLAM %* %+ 1PUTLAM ( To=Tc+Thk*k )
5GETLAM 1GETLAM 7GETLAM %/
%1 7GETLAM 2GETLAM %/ %SQRT
3GETLAM %* %COS %- %* %- ( Thc-To/k*[1-COS[[k/Io]^0.5*t]] )
ABND
;
;

```

Donde las expresiones que están entre paréntesis son comentarios, los cuales se emplean para hacer el código más entendible pero que no forman parte realmente del programa, por lo que en realidad pueden ser omitidos. Sin embargo se recomienda su uso, sobre todo cuando se trabaja con variables locales sin nombre, pues sólo de esa manera el código es entendible y es posible corregir y modificar estos programas. Por ejemplo en el primer comentario se colocan los números que son equivalentes a las variables, así la variable "t" está representada por el número 3, por lo tanto para obtener su

valor se emplea 3GETLAM y para guardar un valor en la misma se emplea 3PUTLAM.

Una vez escrito el programa, guarde las modificaciones (*File -> Save All*), compile el proyecto (*Compilation -> Build Project*), coloque un punto de ruptura en la línea donde se encuentran los números y haga correr el programa paso a paso con $k=3.36$ (*3.36 fn3*). Si todo va bien obtendrá el resultado: .00132605009.

Alternativamente es posible darles nombres temporales a las variables sin nombre. Ello permite trabajar con las variables sin nombre de manera similar a como se trabaja con las variables locales de *User*, sin embargo se debe recordar que estos nombres son sólo temporales pero que no existen realmente en el programa.

Para darles nombres temporales a las variables sin nombre, se declaran colocando los nombres entre dobles llaves:

```
{ { Nombres temporales } }
```

Para utilizar una de estas variables simplemente se escribe el nombre temporal y para guardar un valor se escribe el nombre temporal precedido del signo de admiración (!).

Como ejemplo programaremos nuevamente la función $f(x)=x^3+2x^2+3x+4$, pero empleando nombres temporales para las variables locales sin nombre.

Con ese fin crearemos un nuevo proyecto con el nombre "Proyecto7" (*Project -> New Project*), recordando desmarcar la opción "The project is a library" y marcar la opción "Autoload after compile". El programa se llamará *fn3*, por lo que en *Path* se debe escribir: "PRAC1 *fn3*". En *Files* añada el archivo *FN3.S* (*Project -> New Source File*), recuerde también visualizar el emulador (*Emulator -> Emulate*). Entonces en la ventana de edición escriba el siguiente programa:

```
::
CK1NOLASTWD
CK&DISPATCH1
ONE
::
  { { x } }
  x %3 %^ %2 x %2 %^ %* %+ %3 x %* %+ %4 %+
  ABND
;
;
```

Guarde las modificaciones (*File -> Save All*), compile el proyecto (*Compilation -> Build Project*), coloque un punto de ruptura en la línea donde se encuentra el nombre temporal de la variable y haga correr el programa paso a paso con $x=4.56$ (*4.56 fn1*). Entonces podrá observar que no existe realmente la variable "x", pues en "Next object" verá "1GETLAM" en lugar de "x". Al final obtendrá el resultado 154.086016.

Como se puede observar la elaboración de programas es más sencilla cuando se emplean nombres temporales, lamentablemente, debido a un error en DE-BUG4X, sólo es posible emplear nombres temporales en programas individuales, como los que hemos estado elaborando hasta el momento, pero no en directorios o librerías, por lo que en las mismas nos vemos obligados a trabajar (hasta que se corrija ese error) con los comandos GETLAM y PUTLAM.

Como segundo ejemplo de la aplicación de nombres temporales volveremos a programar la función:

$$f(k) = \theta_c - \frac{T_0}{k} \left(1 - \cos \left(\sqrt{\frac{k}{I_0}} t \right) \right)$$

Donde:

$$T_0 = T_c + \theta_k k$$

$$T_c = 0.625$$

$$\theta_c = 2.97$$

$$\theta_k = 3.27$$

$$t = 0.0191$$

$$I_0 = 0.0006$$

Para ello creamos un nuevo proyecto con el nombre *Proyecto8* (*Project -> New Project*), recordando desmarcar la opción "The project is a library" y marcar la opción "Autoload after compile". El programa se llamará *fn3*, por lo que en *Path* se escribe: "PRAC1 *fn3*". En *Files* añade el archivo *FN3.S* (*Project -> New Source File*), recuerde también visualizar el emulador (*Emulator -> Emulate*). Entonces en la ventana de edición escriba el siguiente programa:

```
::
CK1NOLASTWD
CK&DISPATCH1
ONE
::
% 0.625 % 2.97 % 3.27 % 0.0191 % 0.0006 %0
{{ k Tc Thc Thk t Io To }}
Tc Thk k %* %+
!To
Thc To k %/ %1 k Io %/
%SQRT t %* %COS %- %* %-
ABND
;
;
```

Una vez escrito el programa, guarde las modificaciones (*File -> Save All*), compile el proyecto (*Compilation -> Build Project*), coloque un punto de ruptura en la línea donde se encuentran los números y haga correr el programa paso a paso con $k=3.36$ (*3.36 fun3*). Si todo va bien obtendrá el resultado: .00132605009.

Al elaborar un programa en System RPN no se deben combinar las variables locales con nombre (LAM) con las variables locales sin nombre (NULLLAM), pues la nomenclatura empleada en las variables locales sin nombre (GETLAM y PUTLAM) funciona también con las variables locales con nombre, así el siguiente programa coloca el número %5 en la pila (haga la prueba):

```
::
%3 %2 %0
{ LAM a LAM b LAM c }
BIND
LAM a LAM b %+ ' LAM c STO
1GETLAM
ABND
;
```

De manera que el combinar ambos tipos de variables puede dar lugar a errores difíciles de corregir.

1.5.2.2. Variables globales

Las variables globales se emplean principalmente para guardar programas y algunos datos de importancia en la memoria del usuario (puerto 0). Como ya se dijo en los programas se prefiere trabajar con variables locales porque no interfieren con otros programas y son por lo tanto más seguras.

1.5.2.2.1. User RPN

En *User RPN* ya hemos empleado variables globales para guardar los programas que hemos elaborado. El formato general para guardar un objeto de cualquier tipo, en una variable global (tanto en un programa como en la pila) es el siguiente:

'Nombre de la variable' STO

Donde el objeto a guardar debe encontrarse en la pila. Este comando debe ser utilizado con cierto cuidado, pues no verifica previamente si existe otra variable con el mismo nombre. Por lo tanto si se crea una variable global con un nombre ya existente, se pierde para siempre el contenido de la variable original.

Para eliminar una variable global se emplea el comando *PURGE* de acuerdo al siguiente formato:

'Nombre de la variable' PURGE

O si se quiere eliminar más de una variable, se escriben los nombres de las variables en una lista:

{ Nombres de las variables } PURGE

Igualmente este comando debe ser utilizado con precaución pues por defecto no pide confirmación para eliminar la o las variables.

1.5.2.2.2. System RPN

Las variables globales en *System RPN (IDs)*, se crean de la misma forma en que se guardan nuevos valores en una variable local (con *STO*), sólo que el nombre de la variable debe estar precedido por *ID* en lugar de *LAM*. El formato general para crear (o guardar un nuevo valor) en una variable global en *System* es el siguiente:

' ID Nombre de la variable STO

Donde el objeto a guardar debe encontrarse en la pila.

Para eliminar una variable global se emplea el comando *PURGE* de acuerdo al siguiente formato:

ID Nombre de la variable PURGE

Al trabajar con variables globales en *System* se debe tener incluso más cuidado que en *User*, pues los comandos de *System* suelen tener un alcance más general. Por ejemplo el comando *PURGE* de *System* busca la variable a eliminar en el directorio actual y si no la encuentra busca la variable en el directorio padre y así sucesivamente hasta llegar al directorio raíz (*HOME*). Por lo tanto si en *System* se escribe mal un nombre es posible que se esté eliminando una variable (que no quería eliminarse) en otro directorio.

1.5.3. Preguntas y Ejercicios

Tanto los ejemplos como los ejercicios que se dan al final de las preguntas, deben ser presentados en un CD. Para este fin, una vez realizados los

ejemplos y ejercicios, guarde una copia de la memoria del emulador con el nombre "PRlnn_nnn"; donde "nn_nnn" es el número de su carnet universitario. Entonces, en el CD cree el directorio PRAC1 y dentro de él copie el archivo PRlnn_nnn así como todos los archivos que se encuentran en el directorio RPN\PRAC1 de su disco duro.

1. ¿Con qué lenguaje de programación se pueden lograr los programas más rápidos y eficientes?
2. ¿Por qué no es práctico elaborar programas en el modo algebraico?
3. ¿Por qué los programas escritos en System RPN son más rápidos que los escritos en User RPN?
4. ¿Por qué la programación en System requiere mayor cuidado por parte del programador?
5. ¿Qué aspectos se deben tomar en cuenta al emplear la notación RPN?
6. ¿Qué es la pila?
7. ¿Por qué es importante la pila?
8. Si en la pila se encuentran un programa y un número ¿Cuál de ellos ocupa más memoria en la pila?
9. ¿Por qué en los programas se suele emplear comandos que manejan directamente la pila?
10. ¿Qué son los algoritmos?
11. ¿Por qué al elaborar un diagrama de actividades se debe respetar la sintaxis?
12. ¿Qué es una secuencia?
13. ¿Cómo se implementa una secuencia en User RPN?
14. ¿Cómo se implementa una secuencia en System RPN?
15. ¿Qué tipo de información almacena una variable en User y System RPN?
16. ¿Cuándo se depura un programa cuál es la diferencia entre SST y SST!?
17. Elabore un programa en User RPN para evaluar la función que se presenta a continuación. El programa debe ser guardado con el nombre F4 y probado con f=0.009 (siendo el resultado: .00851274051)

$$f(f) = \frac{1}{\sqrt{f}} - \left(\frac{1}{k}\right) \ln(R_e \sqrt{f}) + \left(14 - \frac{5.6}{k}\right)$$

donde:

$$k = 0.28$$

$$R_e = 37.50$$

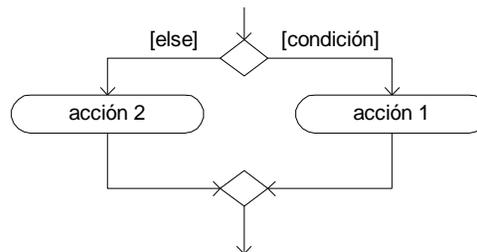
18. Repita el ejercicio anterior pero elaborando el programa en *System RPN* y guardando el programa en la variable "f4".
19. Repita el ejercicio anterior pero empleando variables locales sin nombre y guardando el programa en la variable "fun4".
20. Repita el ejercicio anterior pero empleando nombres temporales para las variables locales sin nombre y guardando el programa en la variable "fn4".

1.5.4. Selección

Sólo los problemas más sencillos, como los estudiados hasta ahora, pueden ser resueltos empleando únicamente la secuencia, en general la mayoría de los problemas prácticos tienen dos o más alternativas que se siguen dependiendo de ciertas condiciones que deben cumplirse. Para este fin se recurre a las estructuras selectivas y tanto en *User* como en *System RPN* se pueden implementar las estructuras selectivas *IF-THEN-ELSE* y *CASE* de PASCAL.

1.5.4.1. Estructura *IF-THEN-ELSE*

La estructura *IF-THEN-ELSE* tiene el siguiente diagrama de actividades:



Y recordemos que en PASCAL se implementa de la siguiente manera:

```
if condición then acción 1 else acción 2;
```

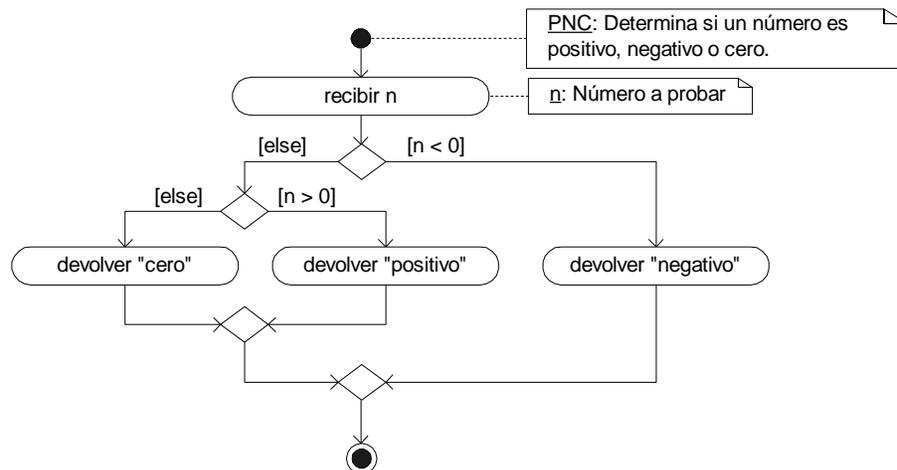
Donde *condición* es cualquier expresión lógica que devuelve un resultado falso o verdadero. Si *acción 1* y/o *acción 2* involucran más de una sentencia se trata como una secuencia, es decir encerrando las sentencias entre *Begin* y *End*.

En *User RPN* esta estructura se implementa de la siguiente manera:

```
IF condición THEN acción 1 ELSE acción 2 END;
```

Si *acción 1* y *acción 2* constan de más de una sentencia no se requiere una secuencia (« »). Al igual que en PASCAL el caso contrario (*ELSE*) es opcional, es decir que puede existir o no. En User el valor verdadero está representado por cualquier número diferente de cero (generalmente 1), mientras que el valor falso está representado por el número 0.

Como ejemplo elaboraremos un programa que determine si un número es positivo, negativo o cero. El algoritmo a seguir es el siguiente:



Como se puede observar en el algoritmo existen dos estructuras *IF-THEN-ELSE* (una anidada). La implementación en *User RPN* es la siguiente (recuerde trabajar en el directorio PRAC1):

```
« → n
« IF n 0 <
  THEN "NEGATIVO"
  ELSE
    IF n 0 >
      THEN "POSITIVO"
      ELSE "CERO"
    END
  END
END
»
»
```

Guarde el programa con el nombre "PNC" ('PNC' STO) y haga correr el mismo (paso a paso) con un número positivo, uno negativo y cero.

En *System RPN* la estructura *IF-THEN-ELSE* se implementa de la siguiente manera:

condición ITE acción 1 acción 2

En *System*, al igual que en PASCAL, si *acción 1* o *acción 2* constan de más de una sentencia deben ser escritas como secuencias (es decir deben estar entre ":" y ";"). Cuando no se quiere el caso contrario (ELSE), en *System* se escribe:

condición IT acción 1

En *System RPN* (al igual que en PASCAL) el valor lógico verdadero es TRUE y el valor lógico falso es FALSE.

Como ejemplo implementemos el algoritmo que determina si un número es positivo negativo o cero en *System RPN*. Con ese fin y siguiendo el procedimiento explicado en los anteriores ejemplos, cree el programa "pnc" en el proyecto "PNC", y escriba el siguiente programa en el archivo PNC.S:

```
::
CK1NOLASTWD
CK&DISPATCH1
ONE
::
  1LAMBIND
  1GETLAM %0 %<
  ITE "NEGATIVO"
  ::
    1GETLAM %0 %>
    ITE "POSITIVO" "CERO"
  ;
  ABND
;
;
```

Finalmente compile el proyecto y haga correr el programa (paso a paso) con un número positivo, otro negativo y cero.

1.5.4.2. Estructura CASE

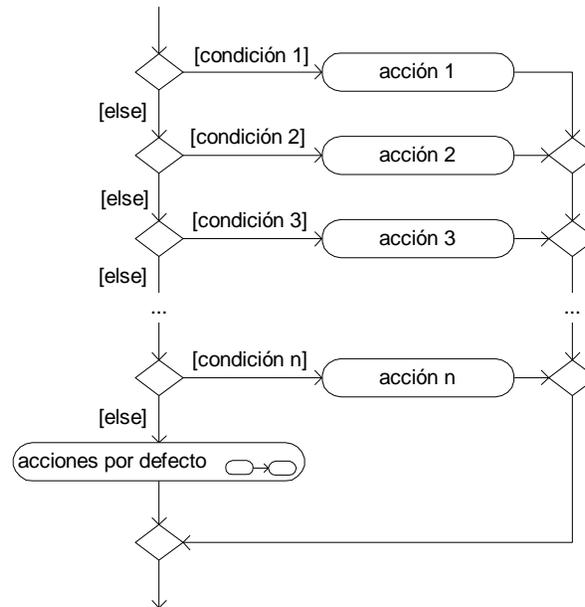
La estructura CASE tiene el diagrama de actividades que se muestra en la siguiente página. Recordemos que la forma de implementar esta estructura en PASCAL es la siguiente:

```

case valor o expresión ordinal of
  caso 1: acción 1;
  caso 2: acción 2;
  caso 3: acción 3;
  . . .
  caso n: acción n;
else
  acciones por defecto;
end;

```

Que está limitada a valores o expresiones ordinales y donde los casos (o condiciones) sólo pueden ser igualdades o intervalos.



En *User RPN* la estructura *CASE* es más completa que en *PASCAL* pues las condiciones pueden ser cualquier expresión lógica y no están limitadas únicamente a igualdades, intervalos y valores ordinales. La forma de implementar esta estructura en *User RPN* es la siguiente:

```

CASE
  condición 1 THEN acción 1 END
  condición 2 THEN acción 2 END
  condición 3 THEN acción 3 END
  . . .
  condición n THEN acción n END
  acciones por defecto
END

```

Donde *acción 1*, *acción 2*, etc., pueden ser una o más sentencias y no necesitan estar en una secuencia (« »). Al igual que en *PASCAL*, las acciones por defecto son opcionales.

Como ejemplo volvamos a resolver el problema de determinar si un número es positivo, negativo o cero, pero ahora empleando la estructura *CASE*. El diagrama de actividades (PNC2), que en cuanto a su lógica es igual al elaborado para la estructura *IF-THEN-ELSE*, ha sido modificado de manera que se identifique mejor la estructura *CASE* y se presenta en la siguiente página.

La implementación de este diagrama en *User RPN* es la siguiente:

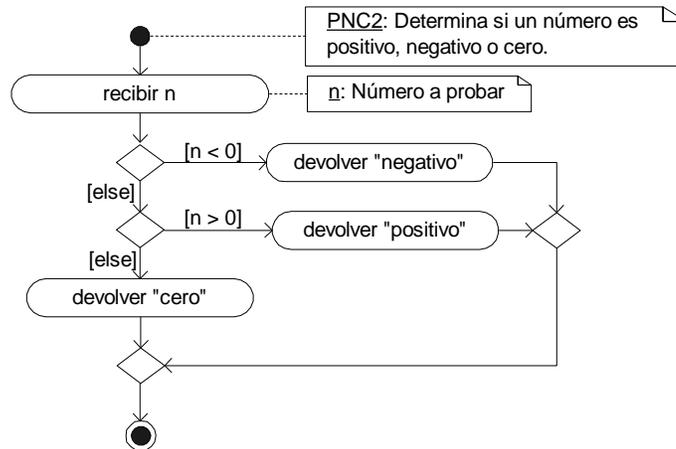
« → n

```

« CASE
  n 0 < THEN "NEGATIVO" END
  n 0 > THEN "POSITIVO" END
  "CERO"
END
»
»

```

Guarde este programa con el nombre "PNC2" (en el directorio PRAC1) y haga correr el mismo (paso a paso) con un número positivo, otro negativo y cero.



En *System RPN*, la estructura *CASE* debe ser implementada dentro de una secuencia de la siguiente manera:

```

::
  condición 1 case acción 1
  condición 2 case acción 2
  condición 3 case acción 3
  . . .
  condición n case acción n
  acciones por defecto
;

```

Donde si *acción 1*, *acción 2*, etc., constan de más de una sentencia deben estar encerradas dentro de una secuencia.

Resolvamos el problema de determinar si un número es positivo, negativo o cero empleando la estructura *case* de *System*. Para ello cree un proyecto con el nombre "PNC2", dé al programa el nombre "pnc2", cree el archivo "PNC2.S" y escriba el siguiente código en la ventana de edición.

```

::
CK1NOLASTWD
CK&DISPATCH1
ONE
::
  1LAMBIND
  ::
    1GETLAM %0< case "NEGATIVO"
    1GETLAM %0> case "POSITIVO"
    "CERO"
  ;
ABND
;
;

```

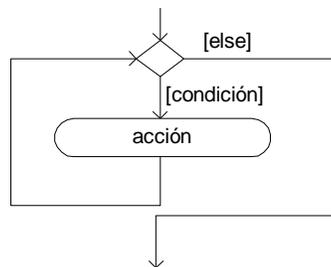
Compile el programa y haga correr el mismo (paso a paso) con un número positivo, uno negativo y cero. Al hacer correr el programa paso a paso, para ingresar a la estructura *case* debe emplear el botón "Step in".

1.6. Iteración

Tanto en *User* como en *System* se pueden implementar las tres estructuras iterativas estándar de PASCAL: *While*, *Until* y *For*. Además en *System* existe un comando especializado que permite crear una estructura infinita.

1.6.1. Estructura *While*

La estructura *While* (*mientras*) tiene el siguiente diagrama de actividades:



Recordemos que en esta estructura la acción se repite *mientras* la condición sea verdadera y termina cuando es falsa. En PASCAL esta estructura se implementa de la siguiente manera:

```
while condición do acción;
```

Donde si la acción involucra más de una sentencia se emplea una secuencia (*begin - end*).

En *User RPN* la estructura *while* se implementa de la siguiente manera:

```
WHILE condición
REPEAT
  acción
END
```

Donde la acción puede ser una o más sentencias y no se requiere una secuencia (pues el final está señalado por la palabra *END*).

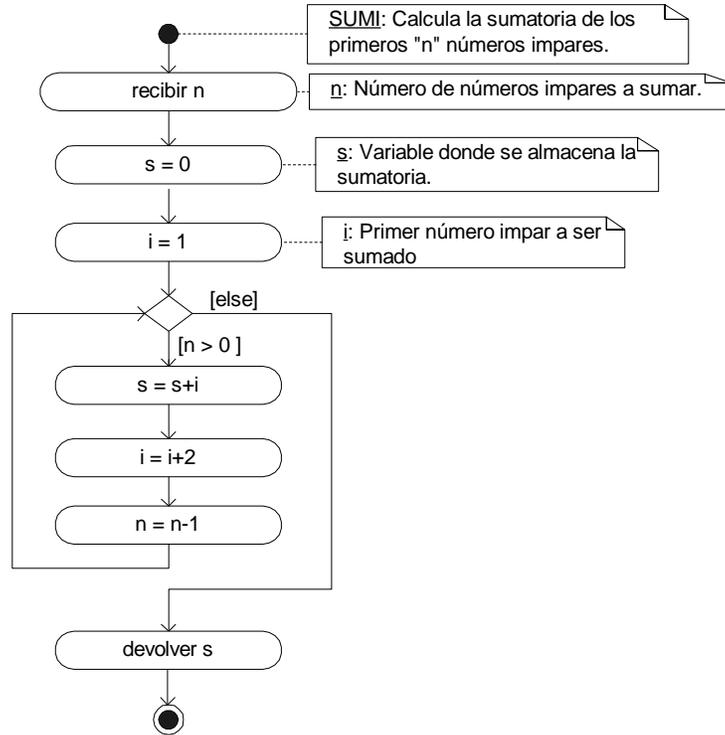
Como ejemplo elaboraremos un programa que calcule la sumatoria de los primeros "n" números impares. El diagrama de actividades (algoritmo) para resolver este problema se presenta en la siguiente página (SUMI).

El código que implementa el algoritmo en *User RPN* es el siguiente:

```
« 0 1 → n s i
« WHILE n 0 >
  REPEAT
    i 's' STO+
    2 'i' STO+
    'n' 1 STO-
  END
  s
»
»
```

Donde los comandos *STO+*, *STO-* (así como *STO** y *STO/*) realizan la operación asociada y almacenan el resultado en la variable respectiva. Así por

ejemplo "i 's' STO+" es equivalente a " i s + 's' STO", "'n' 1 STO-" es equivalente a " n 1 - 'n' STO". Guarde el programa con el nombres "SUMI" (en el directorio PRAC1) y haga correr el mismo (paso a paso) con el número 5, como resultado obtendrá el número 25.



En *System RPN* la estructura *while* se implementa de la siguiente manera:

```

BEGIN
  condición
WHILE
  acción
REPEAT
  
```

Donde si la *acción* involucra más de una sentencia, las mismas deben encontrarse en una secuencia (:: ;). *DEBUG4X* coloca automáticamente el secundario, de manera que cuando se programa en *DEBUG4X* no se debe escribir manualmente la secuencia.

Como ejemplo implementaremos el algoritmo que suma números impares (*SUMI*) en *System RPN*. Para ello cree el proyecto "*SUMI*", dé al programa el nombre "*sumi*", cree el archivo "*SUMI.S*" y escriba el siguiente código en la ventana de edición.

```

::
CK1NOLASTWD
CK&DISPATCH0
ONE
::
%0 %1
{{ n s i }}
BEGIN
  n %0 %>
  WHILE
    s i %+ !s
    i %2 %+ !i
  
```

```

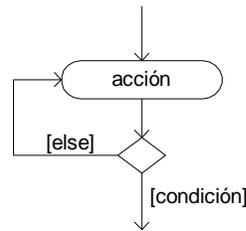
    n %1 %- !n
  REPEAT
  s
  ABND
;
;

```

Compile el proyecto y haga correr el programa (paso a paso) con el número 5. Si todo está bien obtendrá como resultado el número 25.

1.6.2. Estructura UNTIL

La estructura *Until* (*hasta*) tiene el siguiente diagrama de actividades:



Recordemos que en esta estructura la *acción* se repite *hasta* que la *condición* es verdadera y que se prefiere esta estructura (en lugar de *WHILE*) cuando el proceso debe repetirse al menos una vez. En PASCAL esta estructura se implementa de la siguiente manera:

```

repeat
  acción
until condición;

```

Donde la *acción* puede constar de una o más sentencias sin que sea necesario emplear una secuencia (*begin - end*).

En *User RPN* esta estructura se implementa de la siguiente manera:

```

DO
  acción
UNTIL
  Condición
END

```

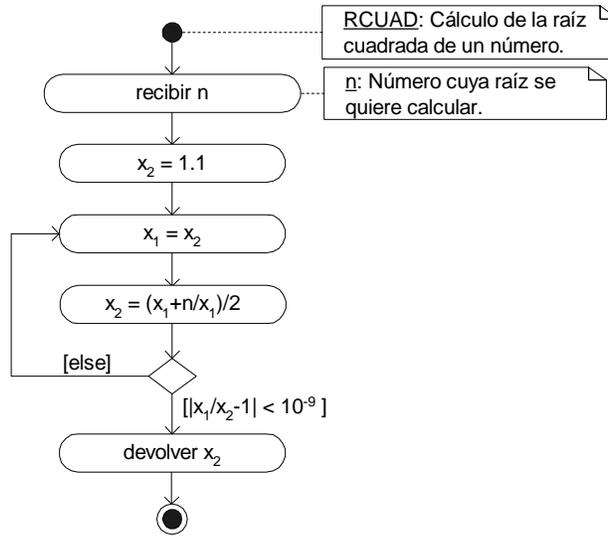
Donde *acción*, al igual que en PASCAL puede constar de una o más sentencias y no se requiere una secuencia (« »).

Como ejemplo elaboraremos un programa que calcule la raíz cuadrada de un número "n" empleando la ecuación de *Newton*:

$$x_2 = \frac{1}{2} \left(x_1 + \frac{n}{x_1} \right)$$

Para resolver esta ecuación se asume un valor inicial de x_1 , usualmente 1.1 y con ese valor se calcula x_2 , si x_1 y x_2 son aproximadamente iguales el proceso concluye, siendo la raíz cuadrada valor x_2 , caso contrario x_1 toma el valor de x_2 y el proceso se repite *hasta* que los dos valores son aproximadamente iguales.

El diagrama de actividades (el algoritmo) para la resolución de este problema se presenta en la siguiente página y en el mismo la condición: $|x_1/x_2 - 1| < 10^{-9}$, que se explicará posteriormente, es verdadera si x_1 y x_2 son iguales en los primeros 9 dígitos.



La implementación del algoritmo en *User RPN* es la siguiente:

```

« 1.1 0 → n x2 x1
« DO
  x2 'x1' STO
  x1 n x1 / + 2 / 'x2' STO
UNTIL
  x1 x2 / 1 - ABS 1E-9 <
END
x2
»
»

```

Guarde el programa con el nombre *RCUAD* (en el directorio *PRAC1*) y haga correr el mismo (paso a paso) con el número 2. Entonces si no ha cometido ningún error obtendrá el resultado: 1.41421356238, que es la raíz cuadrada del número 2 con 9 dígitos de exactitud. Cabe aclarar que los programas elaborados hasta el momento son ideales, pues no toman en cuenta aspectos que si deben ser considerados en un programa real, por ejemplo en el cálculo de la raíz cuadrada se debe verificar que el número sea positivo y en caso contrario generar un error, asimismo no se deben realizar los cálculos si el número es cero o uno, pues para los mismos ya se conoce la respuesta.

En *System RPN* la estructura *UNTIL* se implementa de la siguiente manera:

```

BEGIN
  acción
  condición
UNTIL

```

Donde, al igual que en PASCAL, *acción* puede constar de una o más sentencias y no requiere estar dentro de una secuencia (:: ;). Como ejemplo implementemos el algoritmo que calcula la raíz cuadrada en *System RPN*. Para ello cree el proyecto "RCUAD", dé al programa el nombre "rcuad" (en minúsculas), cree el archivo "RCUAD.S" y en la ventana de edición escriba el siguiente código:

```

::
CK1NOLASTWD
CK&DISPATCH1
ONE
::

```

```

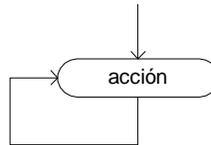
% 1.1 %0
{{ n x2 x1 }}
BEGIN
  x2 !x1
  x1 n x1 %/ %+ %2 %/ !x2
  x1 x2 %/ %1- %ABS % 1E-9 %<
UNTIL
  x2
  ABND
;
;

```

Compile el proyecto y haga correr el programa (paso a paso) con el número 2, entonces deberá obtener el resultado: 1.41421356238.

1.6.3. Cielo infinito

Un ciclo infinito es aquel que se repite por siempre. El diagrama de actividades de un ciclo infinito es el siguiente:



Por supuesto en programación un ciclo infinito como tal no tiene utilidad práctica, por el contrario es uno de los errores que se debe evitar al momento de escribir un programa. Para que tenga utilidad práctica un ciclo infinito debe contar con alguna condición de finalización, pero la misma no está al principio (como en *WHILE*) ni al final (como en *UNTIL*) sino en alguna parte intermedia del ciclo.

En PASCAL no existe un comando específico para crear ciclos infinitos, por lo que los mismos se implementan con la estructura *WHILE* o *UNTIL* escribiendo en lugar de la condición un valor lógico verdadero (true) o falso (false) de manera que el ciclo se repita siempre. Así con la estructura *WHILE* el ciclo infinito se crea de la siguiente manera:

```
while TRUE do acción;
```

Y con la estructura *UNTIL* de la siguiente manera:

```
repeat acción until FALSE;
```

Para salir de un ciclo infinito en PASCAL se utilizan los comandos *break*, *exit* o *goto*.

En *User RPN*, al igual que en PASCAL, no existe un comando para crear ciclos infinitos, por lo que se emplean las estructuras *WHILE* o *UNTIL*, sin embargo como en *User RPN* no existen comandos equivalentes a *break*, *exit* o *goto*, en lugar de la condición se coloca en la pila el valor lógico verdadero (1) o falso (0). El valor lógico se coloca generalmente mediante una estructura *IF-THEN-ELSE* ubicada en el interior del ciclo. Así con la estructura *WHILE* el ciclo infinito se crea de la siguiente manera:

```

1
WHILE
REPEAT
  acciones1
  IF condición
  THEN acciones2 0

```

```

ELSE acciones3 1
END
END

```

Donde el número 1 (verdadero) antes del ciclo se coloca para ingresar por primera vez al ciclo. Si la condición es verdadera se ejecutan las *acciones2* y para salir del ciclo se coloca en la pila el número 0 (falso), caso contrario se ejecutan las *acciones3* y para repetir el ciclo se coloca en la pila el número 1 (verdadero).

Con la estructura *UNTIL* el ciclo infinito se crea de la siguiente manera:

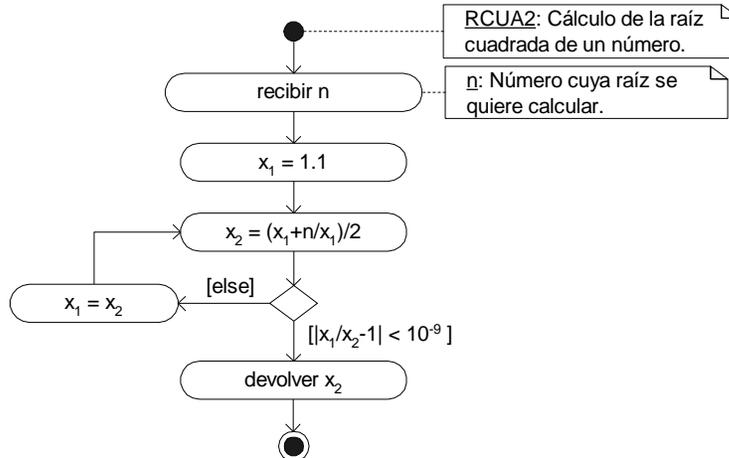
```

DO
  acciones1
  IF condición
  THEN acciones2 1
  ELSE acciones3 0
  END
UNTIL
END

```

En este caso si la condición es verdadera se ejecutan las *acciones2* y para salir del ciclo se coloca en la pila el número 1 (verdadero), caso contrario se ejecutan las *acciones3* y para repetir el ciclo se coloca en la pila el número 0 (falso).

Como ejemplo volveremos a resolver el problema de calcular la raíz cuadrada de un número real, pero en este caso seguiremos la lógica natural o directa que se muestra en el siguiente diagrama de actividades:



Este algoritmo sigue el razonamiento que naturalmente se seguiría para resolver el problema: se asume un valor para x_1 , se calcula x_2 , se compara x_1 con x_2 y si son aproximadamente iguales el proceso concluye, caso contrario x_1 toma el valor de x_2 y el proceso se repite.

La implementación de este algoritmo en *User RPN* empleando la estructura *WHILE* es la siguiente:

```

« 1.1 0 → n x1 x2
« 1
  WHILE
  REPEAT
    x1 n x1 / + 2 / 'x2' STO
    IF x1 x2 / 1 - ABS 1E-9 <
    THEN x2 0
    ELSE x2 'x1' STO 1
  
```

```

    END
  END
  »
  »

```

Guarde este programa con el nombre "RCUA2" y haga correr el mismo (paso a paso) con el número 2, si todo está correcto obtendrá el resultado: 1.41421356238.

La implementación de este programa empleando la estructura *UNTIL* es la siguiente:

```

« 1.1 0 → n x1 x2
  « DO
    x1 n x1 / + 2 / 'x2' STO
    IF x1 x2 / 1 - ABS 1E-9 <
      THEN x2 1
      ELSE x2 'x1' STO 0
    END
  UNTIL
  END
  »
  »

```

Guarde este programa con el nombre "RCUA3" y haga correr el mismo (paso a paso) con el número 2 (resultado: 1.41421356238).

En *System RPN*, se pueden implementar ciclos infinitos con las estructuras *WHILE* o *UNTIL*, al igual que en *User*, pero además en *System* existe una estructura específica, la estructura *AGAIN*, que tiene la siguiente sintaxis:

```

BEGIN
  acciones
AGAIN

```

Lamentablemente no existe un comando específico para salir de este ciclo, por lo que para salir del mismo se debe manipular directamente la pila. Debido a esto, generalmente esta estructura se coloca dentro de una secuencia, y para salir de la misma se emplea un formato similar al siguiente:

```

::
BEGIN
  acciones1
  condición IT :: acciones2 2RDROP;
  acciones3
AGAIN
;

```

Donde si la *condición* es verdadera se ejecutan las *acciones2* y el programa continúa después de la secuencia (después del ;).

Como ejemplo implementaremos el programa que calcula la raíz cuadrada. Para ello cree el proyecto "RCUA2", dé al programa el nombre "rcua2" y cree el archivo "RCUA2.S", entonces en la ventana de edición escriba el siguiente código:

```

::
CK1NOLASTWD
CK&DISPATCH1
ONE
::
% 1.1 %0
{{ n x1 x2 }}

```

```

::
BEGIN
  x1 n x1 %/ %+ %2 %/ !x2
  x1 x2 %/ %1- %ABS % 1E-9 %<
  IT :: x2 2RDROP ;
  x2 !x1
AGAIN
;
ABND
;
;

```

Compile el proyecto y haga correr el mismo (paso a paso) con el número 2, (resultado: 1.41421356238). Como casi siempre ocurre existe más de una forma de elaborar un programa, así el anterior programa también puede ser escrito de la siguiente manera:

```

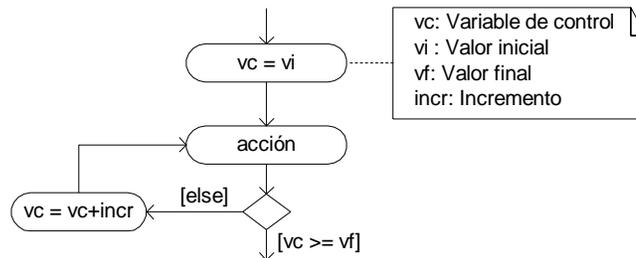
::
CK1NOLASTWD
CK&DISPATCH1
ONE
::
% 1.1 %0
{{ n x1 x2 }}
::
BEGIN
  x1 n x1 %/ %+ %2 %/ !x2 x2
  x1 x2 %/ %1- %ABS % 1E-9 %<
  IT :: 2RDROP ;
  !x1
AGAIN
;
ABND
;
;

```

Donde simplemente se ahorra algo de memoria, al colocar x2 en la pila, antes de la condición (haga la prueba).

1.7. Estructura FOR

La estructura FOR suele tener lógicas ligeramente diferentes según el lenguaje de programación. La lógica más frecuente es la que se muestra en el siguiente diagrama de actividades:



Donde el ciclo se repite, incrementando en cada repetición el valor de la variable de control, hasta que es mayor o igual a un determinado valor final. Generalmente existen dos maneras de implementar este ciclo, una incrementando el valor de la variable de control y otra disminuyendo el valor de la variable de control.

Como se sabe en PASCAL la variable de control (vc) sólo puede ser incrementada (o disminuía) de uno en uno y el ciclo termina cuando su valor es igual al valor final (vf). Además la estructura *FOR* de PASCAL realiza una comprobación previa, de manera que si el valor inicial de la variable de control es mayor al valor final el programa continúa después del ciclo. La implementación de esta estructura en PASCAL, cuando la variable se incrementa de uno en uno es:

```
for vc:=vi to vf do acción;
```

Y cuando disminuye de uno en uno:

```
for vc:=vi downto vf do acción;
```

La estructura *FOR* de *User RPN*, a diferencia de PASCAL, no realiza una comprobación previa al ciclo, de manera que el mismo se ejecuta al menos una vez. La implementación de esta estructura cuando la variable de control se incrementa de uno en uno es la siguiente:

```
vi vf FOR vc
  acción
NEXT
```

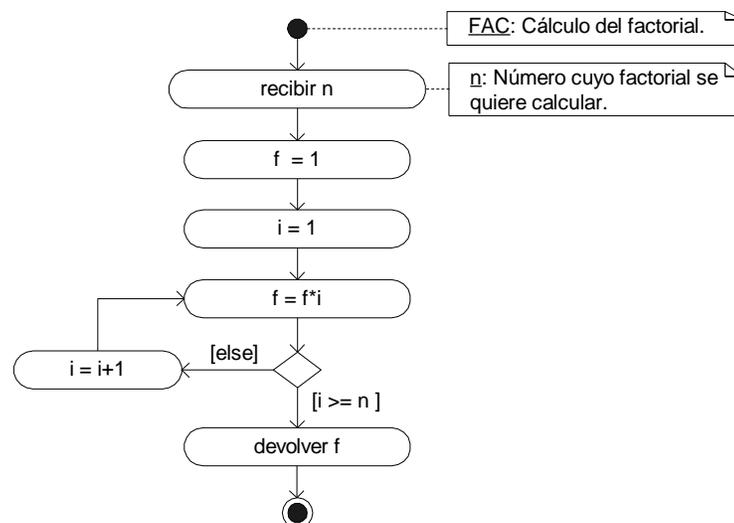
Donde *acción* puede ser una o más sentencias y no se requiere de una secuencia (pues termina con *NEXT*). A diferencia de PASCAL la variable de control en *User RPN* es una variable real y en consecuencia puede tomar valores fraccionarios.

Cuando la variable de control incrementa con valores diferentes a uno, o disminuye en lugar de incrementar, se implementa de la siguiente forma:

```
vi vf FOR vc
  acción
  incr
STEP
```

Donde *incr* es un número real positivo o negativo.

Como ejemplo elaboraremos un programa que calcule el factorial ($n! = 1 * 2 * 3 * \dots * n$). La lógica para resolver este problema se presenta en el siguiente diagrama de actividades:



La implementación de este algoritmo en *User RPN* es la siguiente:

```
« 1 → n f
```

```

« 1 n FOR i
  'f' i STO*
NEXT
f
»
»

```

Guarde este programa con el nombre "FAC" y haga correr el mismo con el número 5 (paso a paso). Obtendrá como resultado el número 120.

Cuando no se requiere el valor de la variable de control dentro del ciclo, la estructura FOR puede ser implementada de la siguiente manera:

```

vi vf START
  acción
NEXT

```

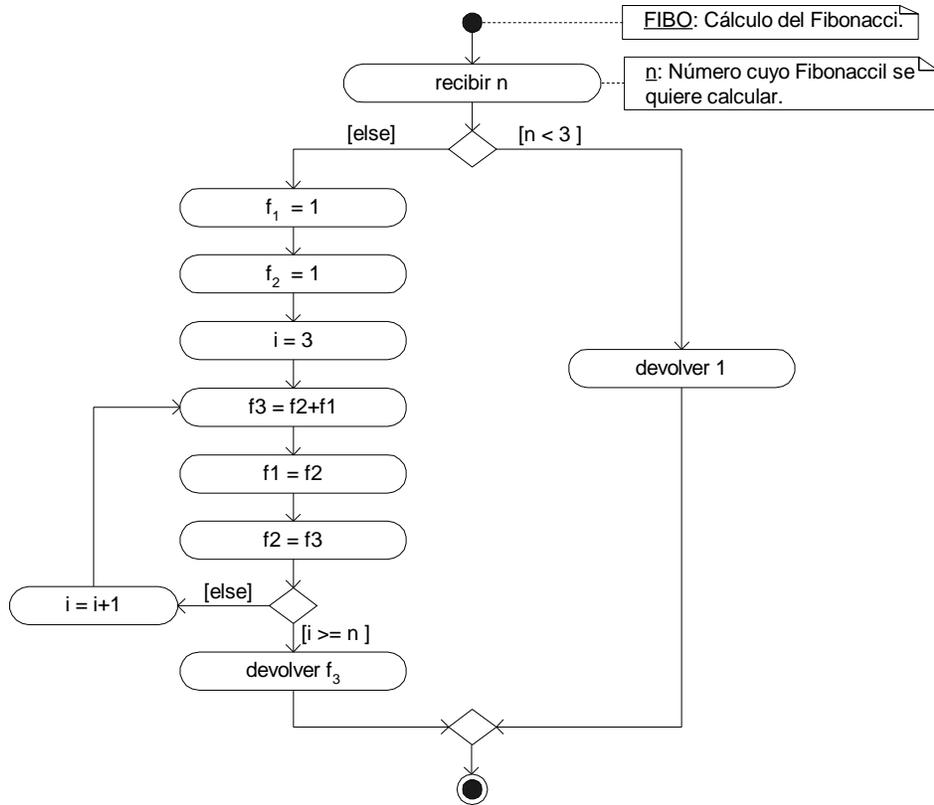
Y cuando el incremento es diferente a uno con:

```

vi vf START
  acción
  incr
STEP

```

Como ejemplo elaboraremos un programa para calcular el Fibonacci. El Fibonacci de un número "n" se calcula con la ecuación: $F_n = F_{n-1} + F_{n-2}$, donde por definición $F_1 = F_2 = 1$. El algoritmo para resolver este problema es el siguiente:



Donde como se puede ver el valor de la variable de control (i) no se utiliza realmente dentro del ciclo, por lo tanto resulta más eficiente emplear START en lugar de FOR:

```

« 1 1 0 → n f1 f2 f3

```

```

« IF n 3 <
  THEN 1
  ELSE
    3 n START
      f2 f1 + 'f3' STO
      f2 'f1' STO
      f3 'f2' STO
    NEXT
      f3
  END
»
»

```

Guarde este programa con el nombre "FIBO" y haga correr el mismo con el número 8. Si no ha cometido ningún error obtendrá el resultado 21.

En *System RPN* la implementación de la estructura *FOR*, cuando la variable de control (*vc*) incrementa de uno en uno es la siguiente:

```

vf+1 vi DO
  acción
LOOP

```

Donde *acción* puede ser una o más sentencias y no se requiere una secuencia (:: ;). Observe que al valor final (*vf*) se le debe sumar 1, pues en *System* el ciclo se repite desde el valor inicial (*vi*) hasta el valor final menos uno (*vf-1*). Tome en cuenta también que en *System* primero se escribe el valor final y luego el valor inicial (al revés de la implementación en *User*).

En la implementación de *System* la variable de control (*vc*) es creada automáticamente y es una variable de tipo entero binario (*BINT*). El valor de la variable de control se recupera con *INDEX@* y se le puede asignar un nuevo valor con *INDEXSTO*. Es posible también recuperar el valor final del ciclo (*vf*) con *ISTOP@* y asignarle un nuevo valor con *ISTOPSTO*.

Cuando existen dos estructuras *FOR* anidadas, la variable de control del ciclo externo se recupera con *JINDEX@* y se le puede asignar un nuevo valor con *JINDEXSTO*. Igualmente el valor final del ciclo externo se recupera con *JSTOP@* y se puede guardar un nuevo valor final con *JSTOPSTO*.

Cuando el incremento (*incr*) de la variable de control es diferente de uno, se implementa de la siguiente forma:

```

vf+1 vi DO
  acción
  incr
+LOOP

```

Donde el incremento (*incr*) debe ser un número entero binario (*BINT*).

Como primer ejemplo implementaremos el algoritmo para calcular el factorial (*FAC*). Para ello cree el proyecto *FAC*, dé al programa el nombre *fact*, cree el archivo *FAC.S* y en la ventana de edición escriba el siguiente código:

```

::
CK1NOLASTWD
CK&DISPATCH1
ONE
::
%1
{{ n f }}

```

```

n COERCE #1+
ONE_DO
  f INDEX@ UNCOERCE %* !f
LOOP
f
ABND
;
;

```

En este programa *COERCE* convierte un número real (*n*) en un entero binario, *#1+* suma uno al resultado de *COERCE*, *ONE_DO* coloca el entero binario *ONE* y luego inicia el ciclo con *DO*, finalmente *UNCOERCE* convierte un entero binario (*INDEX@*) en un número real. Como ya se dijo en *System RPN* muchos de los comandos son simplemente combinaciones de dos o más comandos cuyo único propósito es el de ahorrar memoria, así en este caso en lugar de emplear: *ONE #+ ONE DO*, que son cuatro comandos, se puede emplear *#1+ ONE_DO*, que son dos comandos, o inclusive: *#1+_ONE_DO*, que es un solo comando y que en consecuencia ocupa menos memoria (haga la prueba).

Al compilar el proyecto, debido a un error de *Debug4x*, aparecerá un mensaje de error indicándole que no existe el *DO* para *LOOP*. En este caso ignore el error y haga correr el programa, paso a paso, con el número 5, entonces deberá obtener el resultado 120.

Como segundo ejemplo implementaremos el algoritmo para calcular el Fibonacci (*FIBO*). Para ello cree el proyecto *FIBO*, dé al programa el nombre *fibonacci*, cree el archivo *FIBO.S* y en la ventana de edición escriba el siguiente código:

```

::
CK1NOLASTWD
CK&DISPATCH1
ONE
::
%1 %1 %0
{{ n f1 f2 f3 }}
n 3 %<
ITE
%1
::
  n COERCE #1+ THREE DO
    f2 f1 %+ !f3
    f2 !f1
    f3 !f2
  LOOP
  f3
;
ABND
;
;

```

Compile el proyecto y haga correr el programa con el número 8, entonces deberá obtener el resultado 21.

1.7.1. Preguntas y ejercicios

Al igual que antes, los ejemplos y los ejercicios que se dan al final de las preguntas deben ser presentados en un CD. Para este fin, una vez realizados los ejemplos y ejercicios, guarde una copia de la memoria del emulador con el nombre "PR1Bnn_nnn"; donde "nn_nnn" es el número de su carnet uni-

versitario. Entonces, en el CD cree el directorio PRAC1 y dentro de él copie el archivo PRLnn_nnn así como todos los archivos que se encuentran en el directorio RPN\PRAC1 de su disco duro.

21. En la estructura IF-THEN-ELSE de *User RPN* ¿Las sentencias deben estar dentro de una secuencia?
22. En la estructura IF-THEN-ELSE de *System RPN* ¿Las sentencias deben estar dentro de una secuencia?
23. En la estructura CASE de *User RPN* ¿Las condiciones tienen alguna limitante?
24. En la estructura CASE de *System RPN* ¿Las acciones deben estar dentro de una secuencia?
25. ¿La estructura *WHILE* se repite mientras la condición es falsa?
26. ¿La estructura *HASTA* se repite hasta que la condición sea falsa?
27. ¿En *User RPN* existe un comando específico para crear ciclos infinitos?
28. ¿Cómo se sale de un ciclo infinito en *System RPN*?
29. En la estructura *FOR* de *User RPN* ¿Si el valor inicial es mayor al valor final se ejecuta el ciclo?
30. En la estructura *FOR* de *User RPN* ¿Cuándo se emplea la palabra *START* en lugar de *FOR*?
31. En la estructura *FOR* de *System RPN* ¿La variable de control puede ser un número real?
32. Elabore un programa en *User RPN* que empleando la estructura *IF-THEN-ELSE* determine si un número es par, impar o cero. Guarde el programa con el nombre *PIC*.
33. Resuelva el ejercicio anterior en *System RPN*. Para ello elabore un proyecto con el nombre *PIC* y dé al programa el nombre *pic*.
34. Elabore un programa en *User RPN* que empleando la estructura *CASE* determine si un número es par, impar o cero. Guarde el programa con el nombre *PIC2*.
35. Resuelva el ejercicio anterior en *System RPN*. Para ello elabore un proyecto con el nombre *PIC2* y dé al programa el nombre *pic2*.
36. Elabore un programa en *User RPN* que empleando la estructura *WHILE* calcule la productoria de los primeros "n" números pares. Guarde el programa con el nombre *PROP*.
37. Resuelva el ejercicio anterior en *System RPN*. Para ello elabore un proyecto con el nombre *PROP* y dé al programa el nombre *prop*.
38. Elabore un programa en *User RPN* que empleando la estructura *UNTIL* calcule la raíz cúbica de un número "n" empleando la ecuación de Newton: (guarde el programa con el nombre *RCUB*)

$$x_2 = \frac{1}{3} \left(2x_1 + \frac{n}{x_1^2} \right)$$

39. Resuelva el ejercicio anterior en *System RPN*. Para ello elabore un proyecto con el nombre *RCUB* y dé al programa el nombre *rcub*.
40. Elabore un programa en *User RPN* que resuelva el ejercicio 38, empleando un ciclo infinito implementado con la estructura *WHILE* (guarde el programa con el nombre *RCUB2*).

41. Elabore un programa en *User RPN* que resuelva el ejercicio 38, empleando un ciclo infinito implementado con la estructura *UNTIL* (guarde el programa con el nombre *RCUB3*).
42. Elabore un programa en *System RPN* que resuelva el ejercicio 38, empleando un ciclo infinito (cree el proyecto *RCUB2* y guarde el programa con el nombre *rcub2*).
43. Elabore un programa en *User RPN* que empleando la estructura *FOR* calcule la sumatoria de los primeros "n" números enteros: $\sum_{i=1}^n i$ (guarde el programa con el nombre *SUMN*).
44. Elabore un programa en *User RPN* que empleando la estructura *FOR* (implementada con *START*), calcule la potencia entera de un número real: $x^n = \prod_{i=1}^n x = x * x * \dots * x$ (*n veces*). Guarde el programa con el nombre *POTE*.
45. Resuelva el ejercicio 43 empleando la estructura *FOR* de *System RPN* (cree el proyecto *SUMN* y guarde el programa con el nombre *sumn*).
46. Resuelva el ejercicio 44 empleando la estructura *FOR* de *System RPN* (cree el proyecto *POTE* y guarde el programa con el nombre *pote*).

2. HUMEDAD

En este capítulo estudiaremos una propiedad importante para varias operaciones industriales: secado, humidificación y enfriamiento, etc. En general cuando hablamos de humedad nos referimos a la humedad del aire, sin embargo las relaciones que emplearemos pueden ser fácilmente extendidas a otras mezclas de vapor - gas.

Comenzaremos dando algunas definiciones y conceptos que se emplean con frecuencia cuando se trabaja con la humedad.

2.1. Humedad absoluta (H)

La humedad absoluta o simplemente humedad, se define como los kilogramos de vapor de agua contenidos en 1 kilogramo de aire seco.

Denominaremos H a los kilogramos de vapor de agua contenidos en el kilogramo de aire seco, entonces si transformamos los kilogramos a kilogramosmol obtenemos:

$$\frac{H \text{ kg de vapor de } H_2O}{1 \text{ kg de aire seco}} * \frac{1 \text{ kgmol de vapor de } H_2O}{18.02 \text{ kg de vapor de } H_2O} * \frac{28.97 \text{ kg de aire seco}}{1 \text{ kgmol de aire seco}}$$

$$= \left(\frac{H * 28.97}{18.02} \right) \frac{\text{kgmol de vapor de } H_2O}{\text{kgmol de aire seco}}$$

Ahora si dividimos el lado derecho de esta ecuación entre los kgmol totales (kgmol de H_2O + kgmol de aire seco), lo que se tiene son las fracciones molares:

$$\left(\frac{H * 28.97}{18.02} \right) \frac{\text{kgmol de vapor de } H_2O}{\text{kgmol de aire seco}} = \frac{\text{fracción molar de } H_2O}{\text{fracción molar de aire seco}}$$

$$\left(\frac{H * 28.97}{18.02} \right) \frac{\text{kgmol de vapor de } H_2O}{\text{kgmol de aire seco}} = \frac{\text{fracción molar de } H_2O}{\text{fracción molar de aire seco}}$$

Pero la fracción molar, según la ley de las presiones parciales de Dalton, es igual a la presión parcial entre la presión total, es decir:

$$\left(\frac{H * 28.97}{18.02} \right) = \frac{\text{fracción molar de } H_2O}{\text{fracción molar de aire seco}} = \frac{\frac{P_A}{P}}{\frac{P_B}{P}}$$

Donde A representa al agua y B al aire. Como en esta mezcla sólo existen dos componentes: $P_B = P - P_A$, por lo tanto:

$$\left(\frac{H * 28.97}{18.02} \right) = \frac{\frac{P_A}{P}}{\frac{P_B}{P}} = \frac{P_A}{P - P_A}$$

Finalmente, despejando H se tiene:

$$H = \frac{18.02}{28.97} \left(\frac{P_A}{P - P_A} \right) \quad (1)$$

Esta ecuación permite calcular la humedad del aire en función a la presión parcial del vapor de agua y la presión total de la mezcla. En esta ecuación 18.02 es el peso molecular del agua y 28.97 es el peso molecular

medio del aire. Cuando se trabaja con otras mezclas se deben cambiar estos pesos moleculares por los del vapor y gas involucrados en la nueva mezcla.

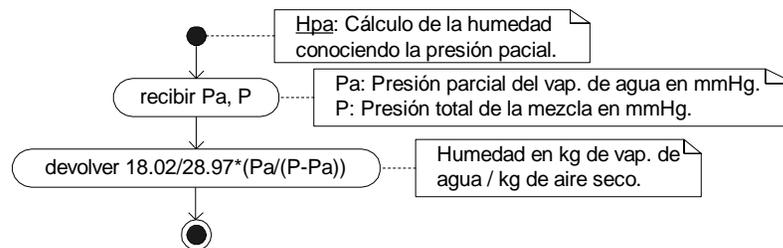
Aun cuando esta ecuación es muy sencilla y puede ser calculada manualmente, se la emplea con relativa frecuencia, razón por lo cual es conveniente contar con un programa que realice la labor automáticamente.

Como ya se mencionó en el primer capítulo, en esta materia los problemas serán resueltos con la ayuda de programas elaborados para la calculadora HP. Los programas de uso más frecuente, como las ecuaciones de definición y métodos de resolución serán elaborados en *System RPN* (por razones de velocidad), mientras que generalmente los programas de aplicación serán elaborados en *User RPN*.

Con este fin crearemos una librería, la librería "PRQ205" número 520 (208 en hexadecimal), la cual iremos modificando continuamente durante el transcurso de la materia.

Comenzaremos esta librería con el programa para calcular la humedad en base a la ecuación de definición (ecuación 1). Si bien la lógica es muy sencilla elaboraremos un diagrama de actividades para recordar luego el orden y las unidades en que deben ser introducidos los datos (parámetros) así como las unidades en que se devuelven los resultados. Además un diagrama de actividades facilita la modificación del código y es general pues puede ser implementado en otros lenguajes.

El diagrama de actividades para la ecuación (1) es el siguiente:



Para implementar este código ingrese a *Debug4x* y cree un nuevo proyecto en el directorio *PRQ205*, dentro del directorio *RPN* con el nombre *PRQ205*. Asegúrese que estén seleccionadas las opciones *The Project is a library* y *AutoLoad Alter Compile*. En el campo de *Title* escriba el título del proyecto: *PRQ205* y en *RomId* el número de la librería (en hexadecimal): *208*. Asegúrese también que el emulador esté activo (*Emulator -> Emulate*). Ahora en *Files* cree un nuevo archivo (*Project -> New Source File*) con el nombre *Humedad*, es en este archivo donde escribiremos los programas de este capítulo.

Como ahora estamos trabajando con librerías, donde tendremos más de un módulo, no podremos emplear las variables locales temporales, pues como ya se dijo, cuando existe más de un módulo, *Debug4x* no compilan bien los módulos que emplean variables temporales, por esa razón nos vemos obligados a trabajar con variables locales sin nombre y variables locales con nombre pero no con variables temporales.

Ahora en el archivo *Humedad* escriba el siguiente código:

```
xNAME Hpa ( Cálculo de la humedad con la presión parcial)
:: ( Datos en la pila Pa=Presión parcial en mmHg; P=Presión total en mmHg)
  CK2NOLASTWD
  CK&DISPATCH1
  # 11
  :: ( Pa=2; P=1 )
    ' NULLLAM TWO NDUPN DOBIND
```

```
% 18.02 % 28.97 %/ 2GETLAM 1GETLAM 2GETLAM %- %/ %*
ABND
```

```
;
;
```

Como puede observar en este código, el módulo tiene un nombre: *Hpa*, que es el mismo nombre empleado en el diagrama de actividades. Para crear un módulo con nombre se emplea la instrucción *xNAME* seguida del nombre del módulo, recuerde que en RPN se diferencia entre mayúsculas y minúsculas, por lo que es importante que escriba los nombres correctamente. Observe también que en este módulo se comprueba que existan dos datos en la pila (*CK2NOLASTWD*) y que ambos sean de tipo real (*CK&DISPATCH1 # 11*).

Una vez escrito el código, guarde las modificaciones (*File -> Save All*) y compile el proyecto (*F9*), entonces, si no ha cometido ningún error la librería se cargará en el emulador y aparecerá dentro de la misma *Hpa*. Una vez compilado el proyecto haga la prueba y haga correr el mismo (paso a paso) con una presión parcial de 20 mm de Hg y una presión total de 760 mm de Hg. Si todo va bien obtendrá como respuesta: *1.68114265456E-2*, que es la humedad del aire en kg de vapor de agua/kg de aire seco.

2.1.1. Ejemplo

Como ejemplo, calculemos la humedad del aire cuando la presión parcial del vapor de agua es 2.76 kPa y la presión total 101.325 kPa.

Aunque el módulo que hemos desarrollado funciona igualmente con otras unidades de presión, es aconsejable trabajar con las unidades para las cuales ha sido elaborado, es decir mm de Hg. Por lo tanto en primer lugar convertiremos los datos a mm de Hg, para ello escribimos en la pila (en el emulador) lo siguiente:

```
{ 2.76_kPa 101.325_kPa } 1_mmHg CONVERT
```

Con lo que obtenemos:

```
{ '20.7017024426_mmHg' '760_mmHg' }
```

Ahora eliminamos las unidades (con *UVAL*), deshacemos la lista (con *LIST→*) y eliminamos el número de elementos (con *DROP*).

```
UVAL LIST→ DROP
```

De esa manera quedan en la pila los dos valores en mm de Hg, ahora simplemente escribimos *Hpa* (o pulsamos la tecla respectiva), con lo que obtenemos el resultado: *.017417773843* que es la humedad del aire en kg de vapor de agua / kg de aire seco.

2.1.2. Preguntas y ejercicios

1. ¿Cómo se define la humedad del aire?
2. Calcule la humedad del aire si la presión parcial del vapor de agua es igual a 0.04 atm y la presión total es 0.79 atm.
3. Calcule la humedad del aire si la presión parcial del vapor de agua es igual a 0.387 lb/plg² (psi) y la presión total es 10.44 lb/plg² (psi).
4. Calcule la humedad del aire si la presión parcial del vapor de agua es igual a 16.07 plg de agua y una presión total de 348.25 plg de agua.

2.2. Humedad de saturación (H_s)

La humedad de saturación es la humedad del aire cuando el vapor de agua se encuentra en equilibrio con el agua líquida a la misma temperatura. La humedad de saturación ocurre cuando la presión parcial del vapor de agua en el aire iguala a la presión de vapor que ejerce el agua líquida a la misma temperatura. Cuando el aire está saturado, cualquier exceso de vapor de agua se condensa.

Si en la ecuación de humedad reemplazamos la presión parcial (P_A) por la presión de vapor (P_{vA}), la humedad resultante es la humedad de saturación.

$$H_s = \frac{18.02}{28.97} \left(\frac{P_{vA}}{P - P_{vA}} \right) \quad (2)$$

En consecuencia, si podemos calcular la presión de vapor la humedad de saturación se determina con el programa *Hpa* mandándole la presión de vapor y la presión total de la mezcla.

2.2.1. Cálculo de la presión de vapor

De las muchas correlaciones existentes para el cálculo de la presión de vapor, estudiaremos las ecuaciones de *Antoine*, *Harlacher* y *Thek-Stiel*.

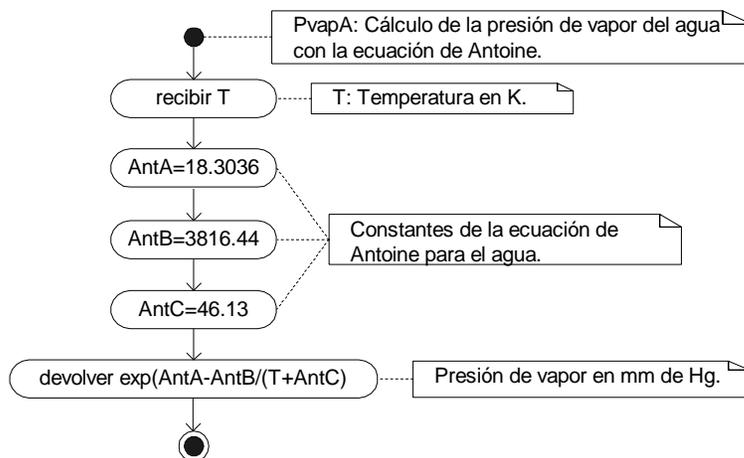
2.2.1.1. Ecuación de Antoine

Una de las correlaciones más sencillas es la ecuación de *Antoine*, que en el caso del agua puede ser empleada con seguridad para temperaturas comprendidas entre 11°C y 168°C aproximadamente:

$$\ln(P_v) = A_{NTA} - \frac{A_{NTB}}{T + A_{NTC}} \quad (3)$$

Donde P_v es la presión de vapor en mm de Hg, T es la temperatura en *Kelvins* y A_{NTA} , A_{NTB} y A_{NTC} son las constantes que de la ecuación. Para el caso del agua estas constantes son: $A_{NTA}=18.3036$, $A_{NTB}=3816.44$ y $A_{NTC}=-46.13$.

El diagrama de actividades para resolver esta ecuación es el siguiente:



El código respectivos, que debe ser añadido al archivo *Humedad.s*, es el siguiente:

```

xNAME pvapA ( Presión de vapor del agua con la ecuación de Antoine )
:: ( Dato en la pila: T = temperatura en K )
CK1NOLASTWD ( resultado en mm de Hg)
    
```

```

CK&DISPATCH1
ONE
:: ( T=4; AntA=3; AntB=2; AntC=1 )
  % 18.3036 % 3816.44 % -46.13
  ' NULLLAM FOUR NDUPN DOBIND
  3GETLAM 2GETLAM 4GETLAM 1GETLAM %+ %/ %- %EXP
  ABND ( resultado en mm Hg)
;
;

```

Una vez escrito el código, guarde el programa y compile el proyecto. Entonces haga la prueba calculando la presión de vapor a 150°F:

```
150_°F 1_K CONVERT UVAL pvapA
```

Con lo que deberá obtener el resultado:

```
192.348983626
```

Que es la presión del vapor de agua a 150 °F, en mm de Hg.

El programa puede ser modificado de manera que permita calcular la presión de vapor de cualquier compuesto, para ello es suficiente recibir las constantes de la ecuación, en lugar de escribirlas como constantes.

2.2.1.2. Ecuación de Harlacher

Otra relación que suele ser más exacta que la ecuación de Antoine es la ecuación de Harlacher:

$$\ln(P_v) = H_{ARA} + \frac{H_{ARB}}{T} + H_{ARC} * \ln(T) + H_{ARD} * \frac{P_v}{T^2} \quad (4)$$

Donde P_v es la presión de vapor en mm de Hg, T es la temperatura en Kelvins y H_{ARA} , H_{ARB} , H_{ARC} y H_{ARD} son las constantes de la ecuación. Para el caso del agua, estas constantes son: $H_{ARA}=55.336$, $H_{ARB}=-6869.50$, $H_{ARC}=-5.115$ y $H_{ARD}=1.05$.

Puesto que la presión de vapor aparece a ambos lados de la ecuación y no puede ser despejada, esta ecuación debe ser resuelta de forma iterativa. De los muchos métodos iterativos disponibles, emplearemos en esta materia los métodos de *Regula Falsi*, *Secante* y *Newton-Raphson*. Se puede emplear también el programa *ROOT*, que viene con la calculadora *HP*, en nuestro caso, sin embargo, no emplearemos este programa porque trabaja con variables globales y porque además suele ser más lenta que los métodos antes mencionados.

Como el fundamento teórico de los métodos antes mencionados ha sido ya estudiado en materias previas, presentamos directamente el código elaborado en *System RPN*. Para implementar este código, cree en *Debug4x* (en el proyecto *PRQ205*), un archivo fuente con el nombre *Metodos1* (*File -> New Source -> Metodos1*) y en él escriba el siguiente código:

```

xNAME Refa ( Método de Regula - Falsi )
:: ( Parámetros: función "f", límites del segmento: "x1", "x2", )
  CK3&Dispatch
  # 811
  :: ( f=12; x1=11; x2=10; err=9; pre=8; max=7; f1=6; f2=5; f3=4;
    fp=3; x3=2; xp=1 )
  ' ID err @ ?SKIP % 1E-10 ( si no existe "err" se emplea 1E-10 )
  ' ID pre @ ?SKIP % 1E-9 ( si no existe "pre" se emplea 1E-9 )
  ' ID max @ ?SKIP % 50 ( si no existe "max" se emplea 50 )
  %0 %0 %0 %0 %0 %0
  ' NULLLAM TWELVE NDUPN DOBIND

```

```

::
11GETLAM 12GETLAM EVAL 6PUTLAM ( f1=f[x1] )
6GETLAM %ABS 9GETLAM %< ( |f1|<err )
IT :: 11GETLAM RDROP ; ( devolver x1 )
10GETLAM 12GETLAM EVAL 5PUTLAM ( f2=f[x2] )
5GETLAM %ABS 9GETLAM %< ( |f2|<err )
IT :: 10GETLAM RDROP ; ( devolver x2 )
11GETLAM 1PUTLAM ( xp=x1 )
6GETLAM 3PUTLAM ( fp=f1 )
BEGIN
    5GETLAM 11GETLAM %* 6GETLAM 10GETLAM %* %- 5GETLAM 6GETLAM %- %/
    2PUTLAM ( [x3=f2x1-f1x2]/[f2-f1] )
    "x:" 2GETLAM a%>$ &$ DispCoord1 ( mostrar x3 )
    2GETLAM 12GETLAM EVAL 4PUTLAM ( f3=f[x3] )
    4GETLAM %ABS 9GETLAM %< ( |f3|<err )
    IT :: 2GETLAM 2RDROP ; ( devolver x3 )
    1GETLAM 2GETLAM %/ %1- %ABS 8GETLAM %< ( |xp/x3-1|<pre )
    IT :: 2GETLAM 2RDROP ; ( devolver x3 )
    7GETLAM %1- 7PUTLAM ( max=max-1 )
    7GETLAM %0= ( max==0 )
    IT :: ABND #A01 DO#EXIT ; ( valor inicial erróneo )
    3GETLAM 4GETLAM %* %0> 4GETLAM 2GETLAM ( f1*f3<0, f3, x3 )
    6GETLAM 4GETLAM %* %0< ( fp*f3>0 )
    ITE
    :: 10PUTLAM 5PUTLAM ( x2=x3, f2=f3 )
    IT :: 6GETLAM %2 %/ 6PUTLAM ; ( f1=f1/2 )
    ;
    :: 11PUTLAM 6PUTLAM ( x1=x3, f1=f3 )
    IT :: 5GETLAM %2 %/ 5PUTLAM ; ( f2=f2/2 )
    ;
    2GETLAM 1PUTLAM ( xp=x3 )
    4GETLAM 3PUTLAM ( fp=f3 )
    AGAIN
;
ABND
;
;

xNAME Seca ( Método de la Secante )
:: ( Parámetros: función "f", Valores asumidos: "x1", "x2", )
CK3&Dispatch
# 811
:: ( f=10; x1=9; x2=8; err=7; pre=6; max=5; f1=4; f2=3; f3=2; x3=1 )
' ID err @ ?SKIP % 1E-10 ( si no existe "err" se emplea 1E-10 )
' ID pre @ ?SKIP % 1E-9 ( si no existe "pre" se emplea 1E-9 )
' ID max @ ?SKIP % 50 ( si no existe "max" se emplea 50 )
%0 %0 %0 %0
' NULLLAM TEN NDUPN DOBIND
::
9GETLAM 10GETLAM EVAL 4PUTLAM ( f1=f[x1] )
4GETLAM %ABS 7GETLAM %< ( |f1|<err )
IT :: 9GETLAM RDROP ; ( devolver x1 )
8GETLAM 10GETLAM EVAL 3PUTLAM ( f2=f[x2] )
3GETLAM %ABS 7GETLAM %< ( |f2|<err )
IT :: 8GETLAM RDROP ; ( devolver x2 )
BEGIN
    3GETLAM 9GETLAM %* 4GETLAM 8GETLAM %* %- 3GETLAM 4GETLAM %- %/
    1PUTLAM ( x3=[f2x1-f1x2]/[f2-f1] )

```

```

"x:" 1GETLAM a%>$ &$ DispCoord1 ( mostrar x3 )
1GETLAM 10GETLAM EVAL 2PUTLAM ( f3=f[x3] )
2GETLAM %ABS 7GETLAM %< ( |f3|<err )
IT :: 1GETLAM 2RDROP ; ( devolver x3 )
8GETLAM 1GETLAM %/ %1- %ABS 6GETLAM %< ( |x2/x3-1|<pre )
IT :: 1GETLAM 2RDROP ; ( devolver x3 )
5GETLAM %1- 5PUTLAM ( max=max-1 )
5GETLAM %0= ( max==0 )
IT :: ABND #A01 DO#EXIT ; ( valor inicial erróneo )
8GETLAM 9PUTLAM ( x1=x2 )
3GETLAM 4PUTLAM ( f1=f2 )
1GETLAM 8PUTLAM ( x2=x3 )
2GETLAM 3PUTLAM ( f2=f3 )
AGAIN
;
ABND
;
;

xNAME Newt ( Método de Newton - Raphson )
:: ( parámetros f: función, forma fx=0, x1: valor asumido )
CK2&Dispatch
# 81
:: ( f=9; x1=8; err=7; pre=6; max=5; h=4; df=3; x2=2; f1=1 )
' ID err @ ?SKIP % 1E-10 ( si no existe "err" se emplea 1E-10 )
' ID pre @ ?SKIP % 1E-9 ( si no existe "pre" se emplea 1E-9 )
' ID max @ ?SKIP % 50 ( si no existe "max" se emplea 50 )
%0 %0 %0 %0
' NULLLAM NINE NDUPN DOBIND
::
BEGIN
8GETLAM 9GETLAM EVAL 1PUTLAM ( f1=f[x1] )
1GETLAM %ABS 7GETLAM %< ( |f1|<err )
IT :: 8GETLAM 2RDROP ; ( devolver x1 )
8GETLAM % .0001 %* 4PUTLAM ( h=x1*0.0001 )
8GETLAM 4GETLAM %+ 9GETLAM EVAL ( f[x1+h] )
8GETLAM 4GETLAM %- 9GETLAM EVAL ( f[x1-h] )
%- %2 4GETLAM %* %/ 3PUTLAM ( df=[f[x1+h]-f[x1-h]]/[2h] )
8GETLAM 1GETLAM 3GETLAM %/ %- 2PUTLAM ( x2=x1-f1/df )
"x2:" 2GETLAM a%>$ &$ DispCoord1 ( mostrar x2 )
8GETLAM 2GETLAM %/ %1- %ABS 6GETLAM %< ( |x1/x2-1|<pre )
IT :: 2GETLAM ABND 2RDROP ; ( devolver x2 )
5GETLAM %1- 5PUTLAM ( max=max-1 )
5GETLAM %0= ( max==0 )
IT :: ABND #A01 DO#EXIT ; ( valor inicial erróneo )
2GETLAM 8PUTLAM ( x1=x2 )
AGAIN
;
ABND
;
;

```

Como puede observar, todos estos métodos trabajan por defecto con un error absoluto (*err*) igual a 1E-10, una precisión (*pre*) igual a 1E-9 y un máximo de iteraciones (*iter*) igual a 50. Para cambiar cualquiera de estos valores se deben almacenar los nuevos valores en las variables globales respectivas.

El método de *Newton - Raphson: Newt*, requiere que estén en la pila la función a resolver (igualada a cero) y un valor inicial supuesto; el método la *Secante: Seca*, requiere la función a resolver (igualada a cero) y dos valores iniciales asumidos (generalmente dos valores próximos entres si) y el método de *Regula - Falsi: Refa*, requiere la función a resolver (igualada a cero) y el valor inicial y final del intervalo de búsqueda.

Una vez escrito el código guarde los cambios y compile el proyecto. Posteriormente haga la prueba escribiendo programas (en *User*) para resolver la ecuación cúbica por los tres métodos:

$$f(x) = x^3 + 2x^2 + 3x + 4 = 0$$

En los programas de prueba que elaboraremos controlaremos el tiempo que requiere el método para encontrar la solución, para ello simplemente se ejecuta el programa con *TEVAL* en lugar de *EVAL*. Dado que la *HP48* no cuenta con este comando (*TEVAL*), los que trabajan con esta modelo deberán añadir el siguiente código al final de los métodos (en el archivo *Metodos1*):

```
xNAME TEVAL
:: ( f=2 : función a evaluar; t=1 : Tiempo inicial del sistem )
  CK1NOLASTWD
  CK&DISPATCH0
  EIGHT ( El dato tiene que ser un subprograma )
  ::
    TOD ( Tiempo antes de evaluar la función )
    ' NULLLAM TWO NDUPN DOBIND
    2GETLAM EVAL ( Se evalúa la función )
    TOD 1GETLAM %HMS- ( Tiempo final - Tiempo inicial )
    %HMS> % 3600 %* ( Convierte el resultado en segundos )
    %4 RNDXY "s" >TAG ( Redondea al cuarto decimal y arma la etiqueta )
    ABND
  ;
;
```

Ahora resolveremos la ecuación cúbica por el método de *Newton - Raphson*, empleando un valor inicial igual a 1.1:

```
«
« → X
« X 3 ^ 2 X SQ * + 3 X * + 4 +
»
» 1.1 « Newt » TEVAL
»
```

Donde el nombre del programa *Newt* se coloca en una secuencia para evitar que se evalúe automáticamente. Haciendo correr el programa con *EVAL* se obtiene la solución y el tiempo empleado en encontrarla.

-1.65062919144

s: 1.4271

Para los otros otros programas de prueba son prácticamente iguales, sólo que cambian los valores iniciales y el nombre del método. Así para el método de *la secante* tenemos:

```
«
« → X
« X 3 ^ 2 X SQ * + 3 X * + 4 +
»
» 1.1 1.2 « Seca » TEVAL
»
```

Resultando: -1.65062919144; s: .8943. Y para el método de *Regula - Falsi*:

```

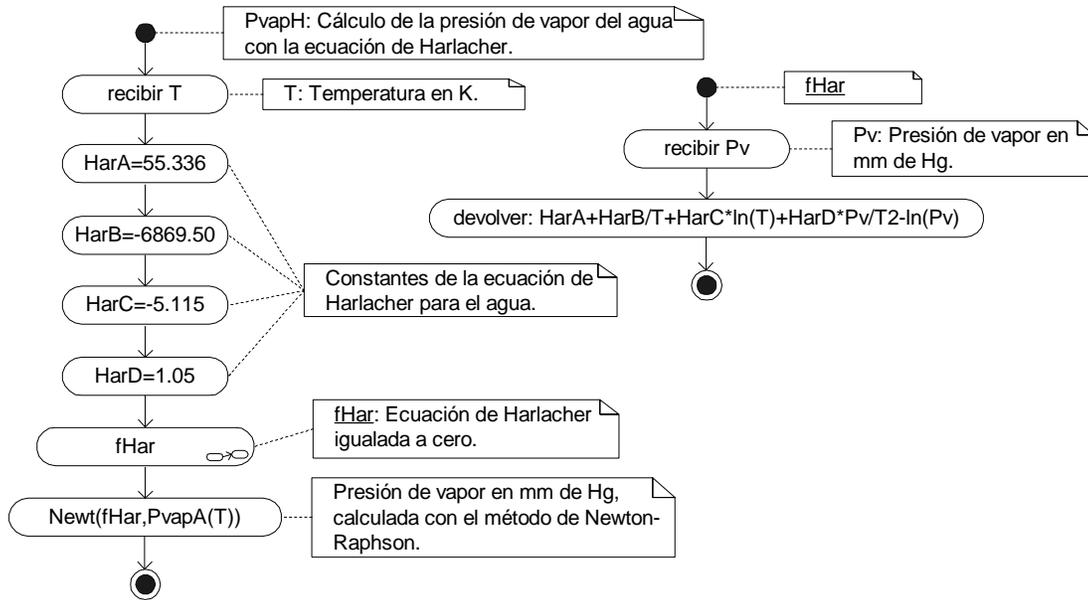
«
« → X
  « X 3 ^ 2 X SQ * + 3 X * + 4 +
  »
» -5 5 « Refa » TEVAL
»

```

Resultando: -1.65062919148; s: 1.0684.

Ahora estamos en condiciones de resolver la ecuación de *Harlacher*. El valor inicial para comenzar el proceso iterativo, será calculado con la ecuación de *Antoine* y se empleará el método de *Newton-Raphson*.

El diagrama de actividades para resolver la ecuación de *Harlacher* (ecuación 4) es el siguiente:



El código respectivo, que debe ser escrito en el archivos *Humedad.s*, es el siguiente:

```

xNAME pvapH ( Presión de vapor del agua con la ecuación de Harlacher)
:: ( Dato en la pila: temperatura en K)
CK1NOLASTWD ( resultado en mm de Hg)
CK&DISPATCH1
ONE
:: ( T=6; HarA=5; HarB=4; HarC=3; HarD=2 )
% 55.336 % -6869.50 % -5.115 % 1.05
{ LAM T LAM HARA LAM HARB LAM HARC LAM HARD }
BIND
'
:: ( Ecuación de Harlacher igualada a cero )
1LAMBIND ( Pvap=1 )
LAM HARA LAM HARB LAM T %/ %+ LAM HARC LAM T %LN %* %+
LAM HARD 1GETLAM LAM T %2 %^ %/ %* %+ 1GETLAM %LN %-
ABND
;
LAM T xpvapA ( valor inicial calculado con Antoine )
xNewt ( resolución de la ecuación con el método de Newton - Raphson)
ABND

```

;
;

Como se puede observar, en este programa se han empleado variables locales con nombre porque las variables se emplean en un subprograma. Observe que se llama a la ecuación de Antoine con *xpvapA* (con una *x* por delante) pues se trata de un nombre *User* (*xNAME*), igualmente y por la misma razón, al método de *Newton* se llama con *xNewt*.

Para probar el programa calcule la presión de vapor del agua a 150°F:

```
150_°F 1_K CONVERT UVAL pvapH
```

Con lo que obtendrá el resultado:

```
192.59491247
```

Que es la presión de vapor del agua a 150°F.

Al igual que la ecuación de Antoine, este programa puede ser modificado de manera que calcule la presión de vapor de cualquier compuesto, para ello se deben recibir las constantes de la ecuación en lugar de escribirlas como constantes.

2.2.1.3. Ecuación de Thek - Stiel

Otra ecuación adecuada para predecir presiones de vapor es la ecuación de Thek-Stiel:

$$\ln(P_{vr}) = A \left(1.14893 - \frac{1}{T_r} - 0.11719T_r - 0.03174T_r^2 - 0.375\ln(T_r) \right) + (1.042\alpha_c - 0.46284A) \left[\frac{T_r^{5.2691+2.0753A-3.1738h} - 1}{5.2691+2.0753A-3.1738h} + 0.040 \left(\frac{1}{T_r} - 1 \right) \right] \quad (5)$$

$$A = \frac{\Delta H_{vb}}{RT_c (1 - T_{br})^{0.375}} \quad (6)$$

$$h = T_{br} \frac{\ln(P_c)}{1 - T_{br}} \quad (7)$$

Donde P_{vr} es la presión de vapor reducida: P_v/P_c ; P_v es la presión de vapor en atm; P_c es la presión crítica en atm; T_r es la temperatura reducida: T/T_c ; T es la temperatura del líquido en Kelvin, T_c es la temperatura crítica en Kelvin, T_{br} es la temperatura normal de ebullición reducida: T_b/T_c ; T_b es la temperatura normal de ebullición en Kelvin; ΔH_{vb} es la entalpía de vaporización a la temperatura normal de ebullición en cal/gmol; R es la constante universal de los gases: 1.987 cal/gmol.

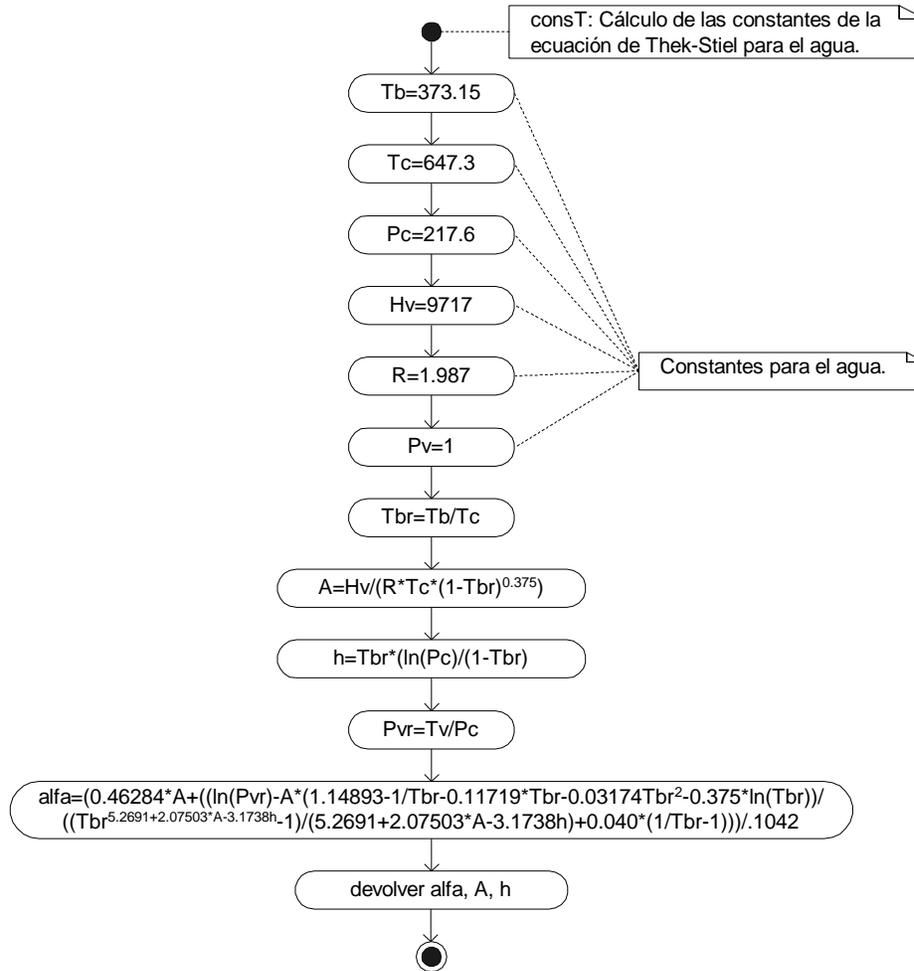
En esta ecuación se debe calcular previamente el valor de α_c :

$$\alpha_c = \frac{0.46284A + \frac{\ln(P_{vr}) - A \left(1.14893 - \frac{1}{T_r} - 0.11719T_r - 0.03174T_r^2 - 0.375\ln(T_r) \right)}{\left[\frac{T_r^{5.2691+2.0753A-3.1738h} - 1}{5.2691+2.0753A-3.1738h} + 0.040 \left(\frac{1}{T_r} - 1 \right) \right]}}{1.042} \quad (8)$$

Para ello se requiere conocer una presión de vapor a una determinada temperatura. Generalmente se emplea la temperatura normal de ebullición (T_b), para la cual se sabe que la presión de vapor es igual a 1 atm (760 mmHg).

Los valores de las constantes para el agua son: $T_c=647.3\text{ K}$; $P_c=217.6\text{ atm}$; $T_b=373.15\text{ K}$ y $\Delta H_{vb}=9717\text{ cal/gmol}$.

Como α_c es una constante para cada compuesto, sólo es necesario calcular su valor una vez, lo mismo es válido para las constantes A y h . El diagrama de actividades para el cálculo de estas constantes es el siguiente:



El programa respectivo, que debe ser escrito en el archivo *Humedad.S*, es el siguiente:

```

xNAME const ( cálculo de las constantes de Thek-Stiel para el agua )
:: ( Tb=10=373.15; Tc=9=647.3; Pc=8=217.6; Hv=7=9717; R=6=1.987; Pv=5=1;
    Pvr=4; A=3; h=2; Tbr=1 )
% 374.15 % 647.3 % 217.6 % 9717 % 1.987 %1 %0 %0 %0 %0
' NULLLAM TEN NDUPN DOBIND
10GETLAM 9GETLAM %/ 1PUTLAM ( Tbr=Tb/Tc)
7GETLAM 6GETLAM 9GETLAM %* %/ %1 1GETLAM %- % 0.375 %^ %/ 3PUTLAM ( A )
1GETLAM 8GETLAM %LN %* %1 1GETLAM %- %/ 2PUTLAM ( h)
5GETLAM 8GETLAM %/ 4PUTLAM ( Pvr=Pv/Pc)
% 1.042 %1/ % 0.46284 3GETLAM %* 4GETLAM %LN 3GETLAM % 1.14893
1GETLAM %1/ %- % 0.11719 1GETLAM %* %- % 0.03174 1GETLAM %2 %^ %* %-
% 0.375 1GETLAM %LN %* %- %* %-
% 5.2691 % 2.0753 3GETLAM %* %+ % 3.1738 2GETLAM %* %- DUP
1GETLAM SWAP %^ %1- SWAP %/ % 0.040 1GETLAM %1/ %1- %* %+ %/ %+ %* ( alfa)
3GETLAM ( A)
2GETLAM ( h)
    
```

ABND

;

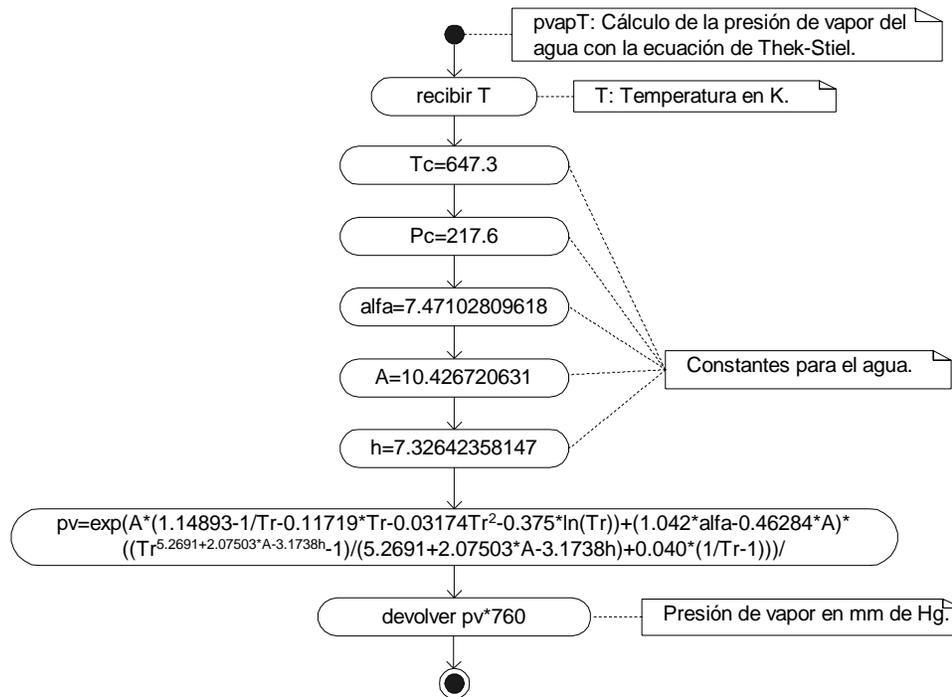
Como las constantes deben ser calculadas una sola vez, el cálculo podía ser llevado a cabo manualmente, sin embargo, este programa puede ser fácilmente modificado, de manera que permita calcular las constantes de cualquier otro compuesto. Para ello simplemente se reciben las constantes del compuesto en lugar de escribirlas directamente en el programa. Haciendo correr el programa se obtiene:

7.47102809618

10.426720631

7.32642358147

Que son los valores de las constantes α_c (7.47455945375), A (10.426720631) y h (7.32642358147) respectivamente. Con estas constantes podemos elaborar ahora el programa para calcular la presión de vapor con la ecuación de Thek-Stiel. El diagrama de actividades para resolver la ecuación de Thek-Stiel (ecuación 5) es el siguiente:



El código, que debe ser escrito en el archivo *Humedad.S*, es el siguiente:

```

xNAME pvapT ( Presión de vapor del agua con la ecuación de Thek-Stiel )
:: ( Dato en la pila : T = Temperatura en K )
CK1NOLASTWD ( resultado en mm de Hg)
CK&DISPATCH1
ONE
:: ( T=7; Tc=6=647.3; Pc=5=217.6; alfa=4=7.54190914447;
    A =3=10.4417356394; h=2=7.37528676787; Tr=1 )
% 647.3 % 217.6 % 7.47455945375 % 10.4274338374 % 7.32874191096 %0
' NULLLAM SEVEN NDUPN DOBIND
7GETLAM 6GETLAM %/ 1PUTLAM ( Tr=T/Tc)
3GETLAM % 1.14893 1GETLAM %1/ %- % 0.11719 1GETLAM %* %- % 0.03174
1GETLAM %2 %^ %* %- % 0.375 1GETLAM %LN %* %- %*
% 1.042 4GETLAM %* % 0.46284 3GETLAM %* %-
    
```

```

% 5.2691 % 2.0753 3GETLAM %* %+ % 3.1738 2GETLAM %* %- DUP
1GETLAM SWAP %^ %1- SWAP %/ % 0.040 1GETLAM %1/ %1- %* %+ %*
%+ %EXP 5GETLAM %* % 760 %*
ABND

```

```

;
;

```

Para probar el programa calcule la presión de vapor del agua a 150°F:

```
150_°F 1_K CONVERT UVAL pvapT
```

Con lo que obtendrá el resultado:

```
189.744461416
```

Que es la presión de vapor del agua a 150°F en mm de Hg.

Al igual que las ecuaciones de *Antoine* y *Harlacher*, este programa puede ser modificado de manera que calcule la presión de vapor de cualquier compuesto, para ello simplemente se deben recibir las constantes de la ecuación en lugar de escribirlas en el programa.

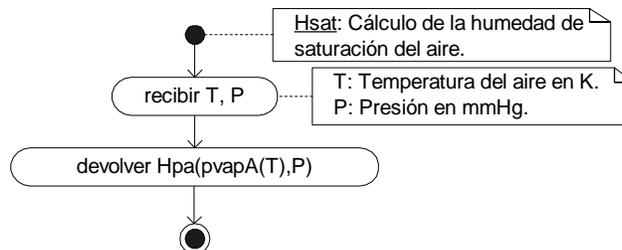
2.2.2. Ejercicios

En los siguientes ejercicios calcule la presión de vapor del agua, en mm de Hg) empleando las ecuaciones de *Antoine*, *Harlacher* y *Thek Stiel*.

5. Calcule la presión de vapor del agua a 350 K.
6. Calcule la presión de vapor del agua a 635 R.
7. Calcule la presión de vapor del agua a 99 °F.
8. Calcule la presión de vapor del agua a 77 °C.
9. Calcule la presión que debería lograrse con una bomba de vacío para que el agua hierva a 50°C.

2.2.3. Programa para el cálculo de la humedad de saturación

Ahora que contamos con programas para el cálculo de la presión de vapor, podemos elaborar un pequeño programa para calcular la humedad de saturación. En el caso del agua y para las temperaturas con las que normalmente se trabaja con el aire en la industria, la ecuación de *Antoine* es lo suficientemente precisa, razón por la cual emplearemos esta ecuación para el cálculo de las presiones de vapor. Sin embargo, para otras mezclas o temperaturas elevadas debemos emplear la ecuación de *Harlacher* o la ecuación de *Thek-Stiel*. El diagrama de actividades para calcular la humedad de saturación es el siguiente:



Como se puede observar, la lógica es tan sencilla que puede ser programado sin la ayuda de variables globales:

```
xNAME Hsat ( Humedad de saturación )
:: ( Datos en la pila: T = temperatura del aire en K;
    P = presión total en mm de Hg)
CK2NOLASTWD
CK&DISPATCH1
# 11
::
    SWAP xpvapA SWAP xHpa
;
;
```

En este código, que debe ser escrito igualmente en el archivo *Humedad.s*, el comando *SWAP* intercambia la posición de los dos últimos datos de la pila, de esa manera el valor de la temperatura es colocado en la posición número 1, entonces se calcula la presión de vapor con la ecuación de Antoine (llamando a *pvapA*). La presión de vapor devuelta por *pvapA* se calcula la humedad llamando a *Hpa*.

Para probar el programa calcule la humedad del aire a una temperatura de 30°C y una presión de 540 mm de Hg:

```
30_°C 1_K CONVERT UVAL 540 Hsat
```

Con lo que obtendrá:

```
3.87296128869E-2
```

Que es la humedad de saturación del aire en kg de vapor de agua por kg de aire seco.

2.2.4. Ejercicios

- 10. Calcule la humedad de saturación del aire a 120 °F y 70 Kpa de presión.
- 11. Calcule la humedad de saturación del aire a 600 R y 20 Psi de presión-
- 12. Calcule la humedad de saturación del aire a 35 °C y 1.2 kg/cm² de presión.
- 13. Calcule la humedad de saturación del aire a 300 K y 0.89 bares de presión.

2.3. Porcentaje de humedad (H_p)

El porcentaje de humedad es la razón entre la humedad y la humedad de saturación multiplicado por 100:

$$H_p = \frac{H}{H_s} * 100 \tag{9}$$

Puesto que ya contamos con los programas para calcular la humedad y la humedad de saturación: *Hpa* y *Hsat*, el porcentaje de humedad puede ser calculado directamente empleando la ecuación 8.

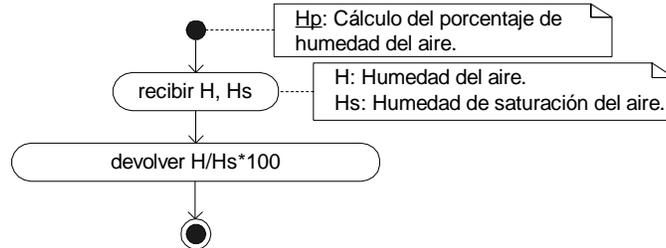
Aunque esta ecuación es muy sencilla elaboraremos un programa para la misma. El diagrama de actividades se presenta en la siguiente página y el código elaborado en base al mismo (que debe ser escrito en el archivo *Humedad.S*) es el siguiente:

```
xNAME Hp ( Porcentaje de humedad )
:: ( Datos en la pila: H = Humedad; Hs = Humedad de saturación)
CK2NOLASTWD
```

```

CK&DISPATCH1
# 11
::
  %/ %100 %*
;
;

```



Una vez más, debido a lo sencillo de la lógica, en el código no se emplean variables.

Para probar el programa calcule el porcentaje de humedad cuando la humedad del aire es 0.0205 y la humedad de saturación 0.0405 (kg de vap. de H₂O/kg de aire seco):

0.0205 0.0405 Hp

Con lo que se obtiene:

50.6172839506

Que es el porcentaje de humedad del aire.

2.3.1. Ejercicios

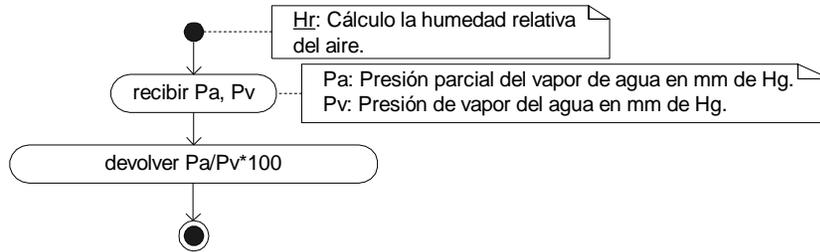
14. Calcule el porcentaje de humedad del aire si la humedad es 0.0145 y la humedad de saturación 0.0345.
15. Calcule el porcentaje de humedad del aire si la presión parcial del vapor de agua es igual a 20.2 mm de Hg, la presión total 600 mm de Hg y se encuentra a una temperatura de 27°C.
16. Calcule el porcentaje de humedad del aire si la presión parcial del vapor de agua es 0.017 atm, la presión total 0.75 atm y se encuentra a una temperatura de 551 R.
17. Calcule el porcentaje de humedad del aire si la presión parcial del vapor de agua es 0.0227 bares, la presión total 0.084 MPa y la temperatura 104 °F.

2.4. Humedad relativa

La humedad relativa, o porcentaje de humedad relativa, es la razón entre la presión parcial del vapor de agua y la presión de vapor del agua (a la misma temperatura) multiplicado por 100:

$$H_r = \frac{P_A}{P_v} * 100 \quad (10)$$

Una vez más se trata de una ecuación muy sencilla. El diagrama de actividades para resolverla se muestra en la siguiente página:



El código respectivo es:

```

xNAME Hr ( Porcentaje de humedad )
:: ( Datos en la pila: Pa = Presión parcial del vapor de agua en mm Hg;
    Pv = Presión de vapor del agua en mm de Hg )
CK2NOLASTWD
CK&DISPATCH1
# 11
::
    %/ %100 %*
;
;

```

Que es el mismo código del porcentaje de humedad, pero que tiene otro nombre.

Para probar el programa calcule la humedad relativa si la presión parcial del vapor de agua es 25.4 mm de Hg y la presión de vapor 55.2 mm de Hg:

25.4 55.2 Hr

Con lo que se obtiene:

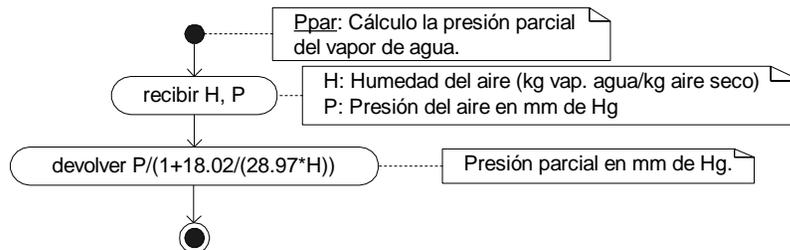
46.0144927536

Que es el porcentaje de humedad relativa (o simplemente la humedad relativa) del aire.

La principal dificultad en el cálculo de la humedad relativa es la determinación de la presión parcial del vapor de agua, pues usualmente no es un dato. Por lo general se tienen la humedad del aire y no la presión parcial, por lo que resulta útil contar con un programa que calcule la presión parcial en base a la humedad y la presión total. Para ello simplemente se despeja la presión parcial de la ecuación de humedad (ecuación 1):

$$P_A = \frac{P}{1 + \frac{18.02}{28.97 * H}} \tag{11}$$

El diagrama de actividades para resolver esta ecuación es el siguiente:



Y el código elaborado en base a este algoritmo es:

```

xNAME Ppar ( Presión parcial del vapor de agua )
:: ( Datos en la pila: H = Humedad del aire;

```

```

P presión del aire en mm de Hg)
CK2NOLASTWD
CK&DISPATCH1
# 11
:: ( H=2; P=1 )
' NULLLAM TWO NDUPN DOBIND
1GETLAM %1 % 18.02 % 28.97 2GETLAM %* %/ %+ %/
;
;

```

Para probar el programa calcule la presión parcial del vapor de agua cuando la humedad del aire es 0.0305 (kg de vap. de agua/kg de aire seco) y la presión total es 540 mm de Hg:

0.0305 540 Ppar

Con lo que se obtiene:

25.2404980326

Que es la presión parcial del vapor de agua en mm de Hg.

2.4.1. Preguntas y ejercicios

18. Calcule la humedad relativa del aire cuya presión parcial es 50.33 mm de Hg y cuya presión de vapor es 70.21 mm de Hg.
19. Calcule la humedad relativa del aire que tiene una humedad de 0.0135 (kg vap. agua/kg aire seco), una presión de 620 mm de Hg y una temperatura de 30°C.
20. Calcule la humedad relativa del aire que tiene una humedad de 0.0243 (kg vap. Agua/kg aire seco), una presión de 0.993 bar y una temperatura de 559.67 R.
21. Calcule la humedad relativa del aire que tiene una humedad de 0.0458 (kg vap. Agua/kg aire seco), una presión de 78660.2 Pa y una temperatura de 122°F.
22. Calcule la humedad relativa del aire que tiene una humedad de 0.0122 (lb vap. agua/lb aire seco), una presión de 13.92 lbf/plg² y una temperatura de 313.15 K.

2.5. Temperatura de rocío

La temperatura de rocío es la temperatura a la cual se forma la primera gota de condensado. Como el aire es una mezcla de gases no condensables y vapor de agua, la temperatura de rocío ocurre cuando la presión parcial del vapor de agua iguala a la presión de vapor del agua a la misma temperatura, es decir cuando se cumple que:

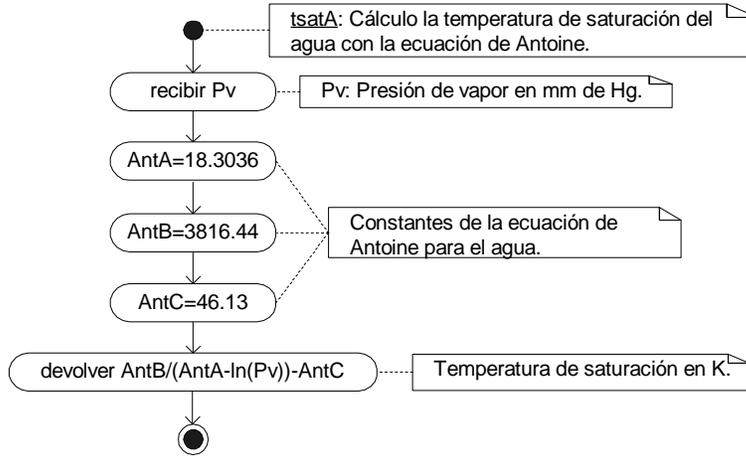
$$P_A = P_v \quad (12)$$

Entonces para calcular la temperatura de rocío podemos resolver una de las ecuaciones de presión de vapor, pero en lugar de calcular la presión de vapor (que ahora es conocida) calcular la temperatura, esta temperatura es la temperatura de rocío.

En el caso de la ecuación de *Antoine*, el cálculo es directo, pues la temperatura puede ser despejada de la ecuación:

$$T = \frac{A_{NTB}}{A_{NTA} - \ln(P_v)} - A_{NTC} \quad (13)$$

El diagrama de actividades para resolver esta ecuación es el siguiente:



Y el código respectivo es:

```

xNAME tsatA ( Temperatura de saturación con la ecuación de Antoine )
:: ( Dato en la pila: Pv = Presión de vapor en mm de Hg )
CK1NOLASTWD
CK&DISPATCH1
ONE
:: ( Pv=4; AntA=3; AntB=2; AntC=1 )
% 18.3036 % 3816.44 % -46.13
' NULLLAM FOUR NDUPN DOBIND
2GETLAM 3GETLAM 4GETLAM %LN %- %/ 1GETLAM %-
ABND ( Temperatura en K)
;
;
  
```

Para probar el programa calcule la temperatura de saturación en °F, cuando la presión de vapor es igual a 192.35 mm de Hg:

```
192.35 tsatA 1_K -UNIT 1_°F CONVERT
```

Con lo que se obtiene:

```
150.00021333_°F
```

Que es la temperatura de saturación a 192.35 mm de Hg (es decir la temperatura a la que herviría el agua a esa presión).

El cálculo de la temperatura de saturación con la ecuación de *Harlacher* es iterativo, pues no se puede despejar la temperatura, por lo tanto debemos recurrir a uno de los métodos iterativos, en este caso emplearemos el método de la secante, tomando como valores iniciales el valor calculado con la ecuación de *Antoine* y 1.01 veces ese valor.

El diagrama de actividades para calcular la temperatura de saturación con la ecuación de *Harlacher* se presenta en la siguiente página y el código elaborado en base al mismo es el siguiente:

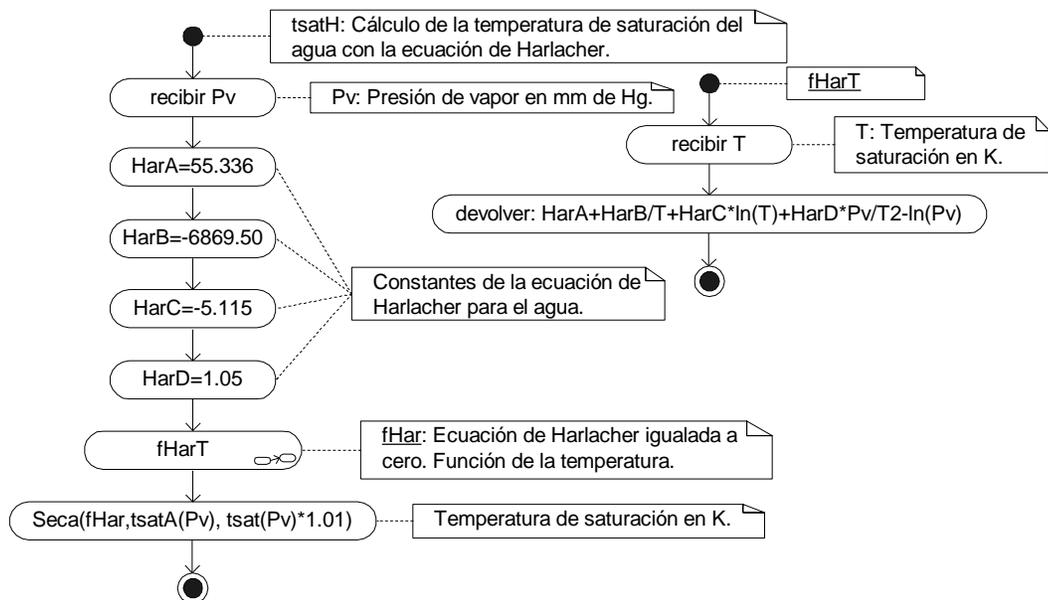
```

xNAME tsatH ( Temperatura de saturación con la ecuación de Harlacher )
:: ( Dato en la pila: Pv = Presión de vapor en mm de Hg )
CK1NOLASTWD
CK&DISPATCH1
ONE
  
```

```

::
% 55.336 % -6869.50 % -5.115 % 1.05
{ LAM Pv LAM HARA LAM HARB LAM HARC LAM HARD }
BIND
'
:: ( Ecuación de Harlacher igualada a cero )
1LAMBIND ( T=Temperatura de sturación = 1 )
LAM HARA LAM HARB 1GETLAM %/ %+ LAM HARC 1GETLAM %LN %* %+
LAM HARD LAM Pv 1GETLAM %2 %^ %/ %* %+ LAM Pv %LN %-
ABND
;
LAM Pv xtsatA DUP % 1.01 %* ( valore iniciales asumidos )
xSeca ( resolución de la ecuación con el método de la Secante )
ABND ( Temperatura en K )
;
;

```



Para probar el programa calcule la temperatura de saturación en °F, cuando la presión de vapor es igual a 192.35 mm de Hg:

```
192.35 tsatH 1_K -UNIT 1_°F CONVERT
```

Con lo que se obtiene:

```
149.948932953_°F
```

Que es la temperatura de saturación a 192.35 mm de Hg.

El cálculo de la temperatura de saturación con la ecuación de *Thek-Stiel* es también iterativo, pues no es posible despejar la temperatura. Al igual que en la ecuación de *Harlacher*, emplearemos el método de la *Secante* para resolver la ecuación.

El diagrama de actividades para calcular la temperatura de saturación con la ecuación de *Thek-Stiel* se presenta en la siguiente página y el código elaborado en base al mismo es el siguiente:

```

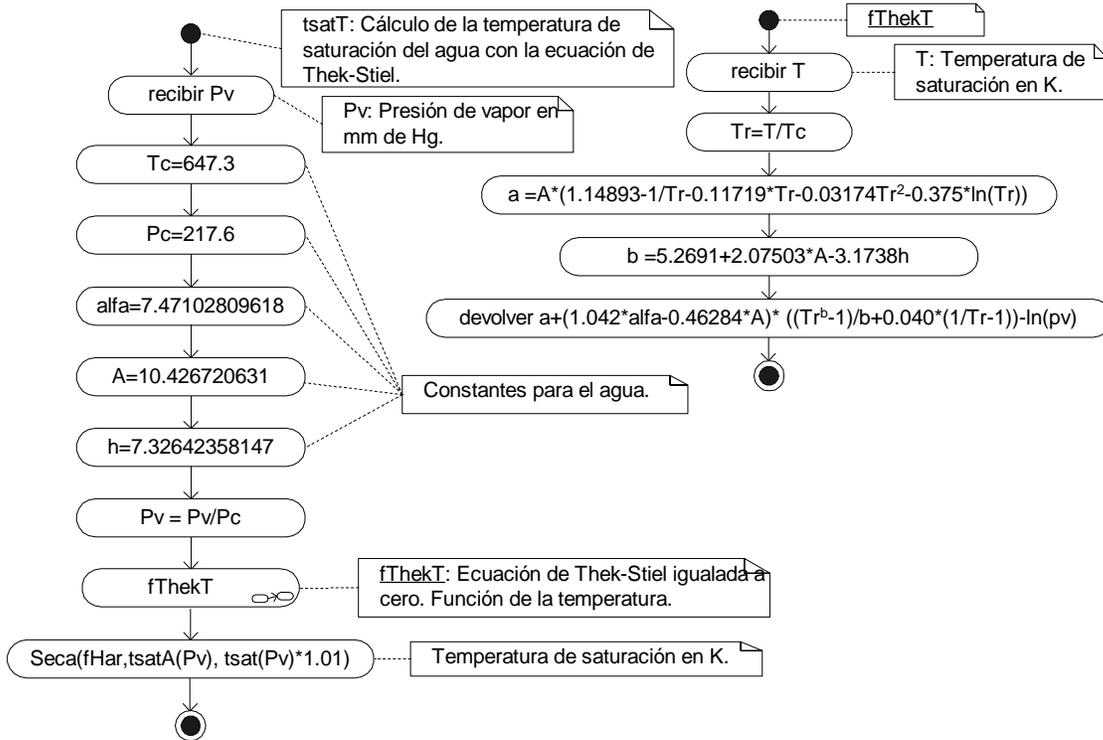
xNAME tsatT ( Temperatura de saturación del agua con Thek-Stiel )
:: ( Dato en la pila: Pv = Presión de vapor en mm de Hg )
CK1NOLASTWD
CK&DISPATCH1

```

```

ONE
::
% 647.3 % 217.6 % 7.47102809618 % 10.426720631 % 7.32642358147
{ LAM Pv LAM Tc LAM Pc LAM alfa LAM A LAM h }
BIND
'
:: ( Thek-Stiel igualada a cero )
1LAMBIND ( T = Tr = 1 )
1GETLAM LAM Tc %/ 1PUTLAM ( Tr=T/Tc)
LAM A % 1.14893 1GETLAM %1/ %- % 0.11719 1GETLAM %* %- % 0.03174
1GETLAM %2 %^ %* %- % 0.375 1GETLAM %LN %* %- %* ( a )
% 5.2691 % 2.0753 LAM A %* %+ % 3.1738 LAM h %* %- ( b )
SWAP % 1.042 LAM alfa %* % 0.46284 LAM A %* %-
1GETLAM 4 PICK %^ %1- 4ROLL %/ % 0.040 1GETLAM %1/ %1- %* %+ %*
%+ LAM Pv %LN %-
ABND
;
LAM Pv xtsatA DUP % 1.01 %* ( Valores iniciales para la Secante )
LAM Pv % 760 %/ LAM Pc %/ ' LAM Pv STO ( Presión de vapor reducida )
xSeca ( Resolución de la función con el método de la Secante )
ABND ( Temperatura calculada en K )
;
;

```



Para probar el programa calcule una vez más la temperatura de saturación en °F, cuando la presión de vapor es igual a 192.35 mm de Hg:

```
192.35 tsatT 1_K →UNIT 1_°F CONVERT
```

Con lo que se obtiene:

```
150.54590644_°F
```

Que es la temperatura de saturación a 192.35 mm de Hg.

2.5.1. Ejercicios

Resuelva los siguientes ejercicios empleando en cada uno de ellos las ecuaciones de Antoine, Harlacher y Thek-Stiel.

23. Calcule la temperatura de rocío del aire que tiene una presión parcial del vapor de agua igual a 20 mm de Hg.
24. Calcule la temperatura de rocío del aire que tiene una presión parcial del vapor de agua es igual a 5.33 kPa.
25. Calcule la temperatura de rocío del aire que tiene una humedad de 0.0124 (kg vap. agua/kg aire seco) y que se encuentra a una presión de 660 mm de Hg.
26. Calcule la temperatura de rocío del aire que tiene una humedad de 0.0342 (lb de vap. agua/lb de aire seco) y que se encuentra a una presión de 0.987 bar.
27. Calcule la temperatura de rocío del aire que tiene una humedad relativa del 60% y que se encuentra a 25°C de temperatura.
28. Calcule la temperatura de rocío del aire que tiene una humedad relativa del 70% y que se encuentra a 180°F de temperatura.

2.6. Volumen húmedo (V_R)

El volumen húmedo es el volumen ocupado por 1 kg de aire seco más el volumen ocupado por el vapor de agua que contiene. Puesto que normalmente se trabaja a presión atmosférica (1 atm o menos), el volumen húmedo puede ser calculado con la ley de los gases ideales:

$$PV = nRT \Rightarrow V = \frac{nRT}{P} \quad (14)$$

El número de moles "n" puede ser calculado con el kg de aire seco y los H kg de vapor de agua que contiene:

$$n = \left(\frac{1 \text{ kg de aire seco}}{28.97 \text{ kg de aire seco}} \right) \frac{1 \text{ kgmol de aire seco}}{28.97 \text{ kg de aire seco}} + \left(\frac{H \text{ kg de vap. de H}_2\text{O}}{18.02 \text{ kg de vap. de H}_2\text{O}} \right) \frac{1 \text{ kgmol de vap. de H}_2\text{O}}{18.02 \text{ kg de vap. de H}_2\text{O}}$$

$$n = \left(\frac{1}{28.97} + \frac{H}{18.02} \right) \text{kgmol} \quad (15)$$

Si la temperatura está en K, la presión en mm de Hg y los moles en kgmol, la constante universal de los gases (R) debe estar en $\text{m}^3 \cdot \text{mmHg} / \text{kgmol} \cdot \text{K}$:

R CONST 1_m^3*mmHg/(kgmol*K) CONVERT

Con lo que se obtiene:

62.3639536146_m^3*mmHg/(kgmol*K)

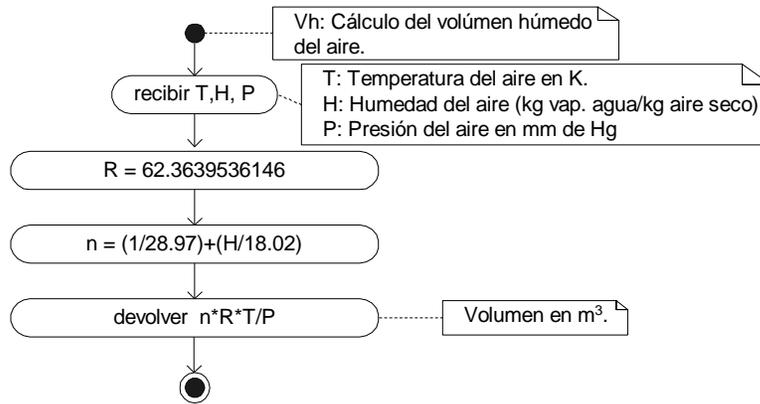
Que es el valor de la constante universal de los gases en las unidades requeridas. Con este valor y la ecuación (15) se puede resolver la ecuación (14). El diagrama de actividades para resolver esta ecuación se presenta en la siguiente página y el código elaborado en base al mismo es:

```
xNAME Vh ( Volumen del aire húmedo )
:: ( Datos en la pila: T: Temperatura en K, H: Humedad,
   P: Presión en mmHg )
```

```

CK3&Dispatch
# 111
:: ( T=4; H=3; P=2; R=1 )
% 62.3639536146
' NULLLAM FOUR NDUPN DOBIND
% 28.97 %1/ 3GETLAM % 18.02 %/ %+ ( n )
1GETLAM %* 4GETLAM %* 2GETLAM %/ ( Volumen en m^3)
ABND
;
;

```



Como ejemplo calculemos el volumen del aire que tiene una temperatura de 30°C, una humedad de 0.42 (kg vap. H₂O/kg aire seco) y una presión de 540 mm de Hg:

```
30_°C 1_K CONVERT UVAL 0.042 540 Vh
```

Con lo que obtenemos

```
1.29010674529
```

Que es el volumen del aire en m³.

2.6.1. Ejercicios

29. Calcule el volumen del aire que se encuentra a 20°C, tiene una humedad de 0.0023 (kg vap. H₂O/kg aire seco) y una presión de 14 lb/plg².
30. Calcule el volumen del aire que se encuentra a 70°F, tiene una humedad de 0.056 (kg vap. H₂O/kg aire seco) y una presión de 0.89 bar.
31. Calcule el volumen del aire que se encuentra a 509.67 R, tiene una presión parcial del vapor de agua igual a 21.43 pulgadas de agua y una presión total de 590 Torricelis.
32. Calcule el volumen del aire que se encuentra a 310 K, tiene una temperatura de rocío de 5 °C y una presión de 0.76 kgf/cm².
33. Calcule el volumen del aire que se encuentra a 14°C, tiene una humedad relativa del 40% y una presión de 0.80 atm.
34. Calcule el volumen del aire que se encuentra a 80°F, tiene un porcentaje de humedad del 60% y una presión de 650 mm de Hg.
35. Calcule el volumen del aire que se encuentra a 50°F, tiene una temperatura de rocío de 37°F y una presión de 0.89 atm.

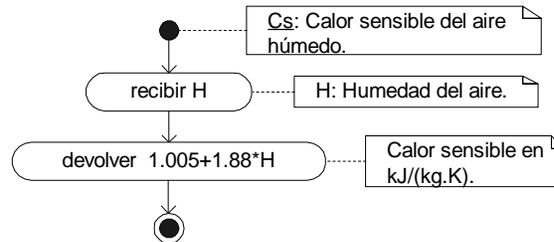
2.7. Calor sensible (C_s)

El calor sensible del aire húmedo es la cantidad de calor necesaria para elevar en un grado Celsius la temperatura de 1 kg de aire seco más el vapor de agua que le acompaña.

Puesto que cuando se trabaja con el aire generalmente se trabaja a temperatura ambiente, no existen variaciones extremas de temperatura, por lo tanto el calor sensible puede ser calculado empleando la capacidad calorífica media del aire seco: $1.005 \text{ kJ}/(\text{kg.K})$ y la del vapor de agua: $1.88 \text{ kJ}/(\text{kg.K})$. Por lo tanto el calor sensible del aire húmedo es:

$$C_s = 1.005 + 1.88H \quad (16)$$

El algoritmo para resolver esta ecuación es el siguiente:



Y el código respectivo:

```

xNAME Cs ( Calor sensible del aire húmedo )
:: ( Dato en la pila: H: Humedad del aire)
  CK1&Dispatch
  ONE
  ::
  % 1.88 %* % 1.005 %+ ( Calor sensible en kJ/kg.K )
  ;
  ;
  
```

Haciendo correr el programa para una humedad igual a 0.0305:

0.0305 Cs

Se obtiene:

1.06234

Que es el calor sensible del aire en $\text{kJ}/(\text{kg.K})$.

2.7.1. Ejercicios

36. Calcule el calor sensible del aire cuya humedad es 0.0145.
37. Calcule el calor sensible del aire que se encuentra a 20°C , tiene una humedad relativa del 70% y una presión de 700 mm de Hg.
38. Calcule el calor sensible del aire que se encuentra a 122°F , tiene un porcentaje de humedad es 50% y una presión de 0.67 atm.
39. Calcule el calor sensible del aire que se encuentra a 551.07°R , tiene una temperatura de rocío es de 17°C y una presión de 0.072 MPa.
40. Calcule el calor sensible del aire que tiene una presión parcial de vapor de agua igual a 0.1934 lbf/plg^2 y una presión total de 0.9 kgf/cm^2 .

2.8. Entalpía (H_y)

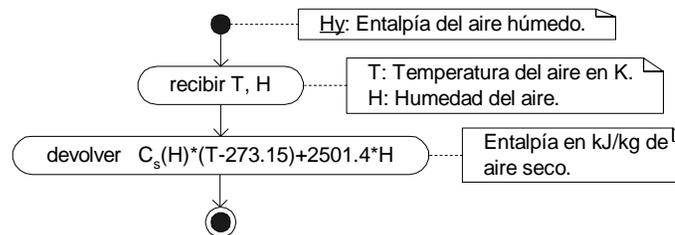
Es la cantidad de calor necesaria para llevar 1 kg de aire seco más el vapor de agua que contiene desde un estado de referencia hasta la temperatura y presión de operación.

Nuestro estado de referencia será 0°C (273.15 K), estando el agua como líquido. Con esta referencia, la entalpía del aire es igual a la temperatura necesaria para llevar 1 kg de aire húmedo desde 0°C hasta la temperatura de operación, más el calor necesario para evaporar el agua que contiene a la temperatura de referencia:

$$H_y = C_s(T - 273.15) + \lambda_0 H \quad (17)$$

Donde λ_0 es la entalpía de vaporización del agua a $0^\circ\text{C} \approx 2501.4\text{ kJ/kg}$.

El algoritmo que permite resolver esta ecuación es el siguiente:



Y el código respectivo es:

```
xNAME Hy ( Entalpía del aire húmedo )
:: ( Datos en la pila: T: Temperatura del aire en K; H: Humedad )
  CK2&Dispatch
  # 11
  ::
  DUP xCs ROT % 273.15 %- %* SWAP % 2501.4 %* %+ ( Entalpía en Kj/kg )
  ;
;
```

Como ejemplo calculemos la entalpía del aire que se encuentra a 35°C y cuya humedad es 0.015 (kg de H_2O /kg de aire seco):

```
35_°C 1_K CONVERT UVAL 0.015 Hy
```

Con lo que obtenemos:

73.683

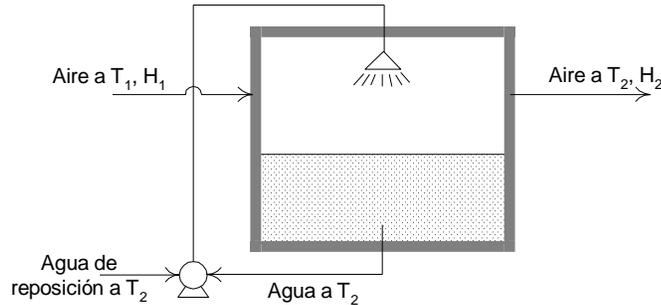
Que es la entalpía de aire en kJ/kg de aire seco.

2.8.1. Ejercicios

41. Calcule la entalpía del aire que se encuentra a 100°F y tiene una humedad de 0.0305 kg de H_2O /kg de aire seco.
42. Calcule la entalpía del aire que se encuentra a 617.67°R , tiene una temperatura de rocío de 86°F y se encuentra a una presión de 12 psi .
43. Calcule la entalpía del aire que se encuentra a 80°F , tiene una humedad relativa del 30% y se encuentra a una presión de 0.71 atm .
44. Calcule la entalpía del aire que se encuentra a 40°C , cuya presión parcial del vapor de agua es igual a 23 mm de Hg y cuya presión total es 0.77 bar .

2.9. Temperatura de saturación adiabática

La temperatura de saturación adiabática se alcanza cuando el equipo mostrado en la figura alcanza el estado estacionario. La temperatura que alcanza el aire a la salida (T_2) se conoce como la temperatura de saturación adiabática.



Si se da el tiempo suficiente, el aire y el agua alcanzan el equilibrio y en consecuencia el aire sale saturado a la temperatura T_2 .

Puesto que el equipo es adiabático, el balance de entalpía en el equipo es el siguiente:

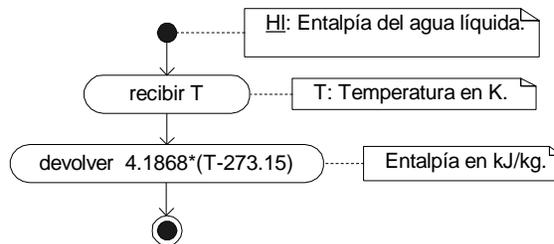
Entalpía de entrada = Entalpía de salida

$$H_{y1} + H_{l2} * (H_2 - H_1) = H_{y2} \tag{18}$$

Donde H_{L2} es la entalpía del agua líquida. Puesto que en el estado de referencia tenemos agua a 273.15 K, la entalpía del agua líquida es la cantidad de calor necesaria para llevar el agua desde 273.15 K hasta la temperatura de operación. Como las temperaturas a las que trabajamos son ambientales, la capacidad calorífica no cambia mucho, por lo que la entalpía puede ser calculada empleando la capacidad calorífica media del agua (C_{pA}):

$$H_l = \bar{C}_{pA} (T - T_0) \tag{19}$$

La capacidad calorífica media del agua es aproximadamente: 4.1868 kJ/(kg.K). El diagrama de flujo para resolver esta sencilla ecuación es:



Y el código respectivo:

```
xNAME Hl ( Entalpía del agua líquida )
:: ( Dato en la pila: T: Temperatura del agua en K )
  CK1&Dispatch
  ONE
  ::
  % 273.15 %- % 4.1868 %*
  ;
;
```

Por ejemplo, la entalpía del agua a 33°C es:

33_°C 1_K CONVERT UVAL H1 => 138.1644

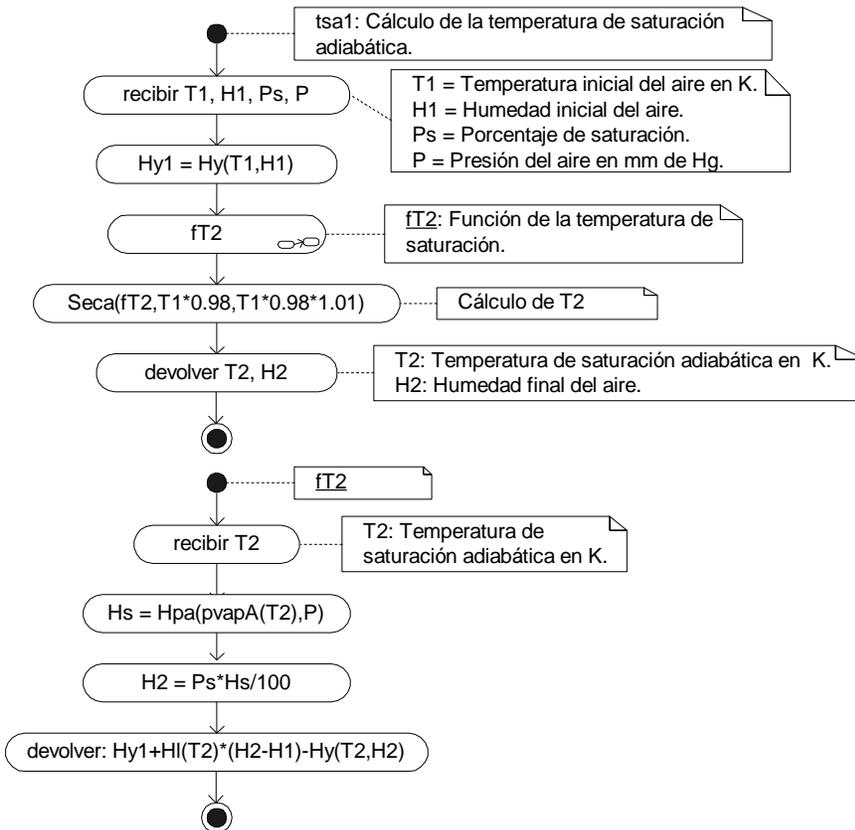
La temperatura de saturación adiabática se calcula para un determinado porcentaje de saturación:

$$P_s = \frac{H_2}{H_{s2}} * 100 \quad (20)$$

Donde H_{s2} es la humedad de saturación a la temperatura T_2 . Si se conoce T_2 y el porcentaje de saturación P_s , se puede calcular la humedad final H_2 y con ella la entalpía final (H_{y2}), en consecuencia la ecuación (18) es función de una sola variable: La temperatura de saturación adiabática (T_2):

$$f(T_2) = H_{y1} + H_{i2} * (H_2 - H_1) - H_{y2} = 0 \quad (21)$$

Puesto que es función de una sola variable, podemos resolverla empleando uno de los métodos que tenemos programados (*Newton*, *Secante* o *Regula-Falsi*). En esta ocasión resolveremos la ecuación empleando el método de la *Secante*. El diagrama de actividades es:



Y el código respectivo:

```
xNAME tsa1 ( Temperatura de saturación adiabática )
:: ( Datos en la pila: T1: Temperatura inicial en K; H1: Humedad inicial;
    Ps: Porcentaje de saturación; P: Presión del aire en mm de Hg)
CK4&Dispatch
# 1111
::
%0 %0
{ LAM T1 LAM H1 LAM Ps LAM P LAM Hy1 LAM H2 }
BIND
LAM T1 LAM H1 xHy ' LAM Hy1 STO
```

```

'
:: ( fT2 )
  1LAMBIND ( T2 = 1)
  1GETLAM xpvapA LAM P xHpa
  LAM Ps %* %100 %/ ' LAM H2 STO
  LAM Hy1 1GETLAM xH1 LAM H2 LAM H1 %- %* %+ 1GETLAM LAM H2 xHy %-
  ABND
;
LAM T1 % .98 %* DUP % 1.01 %* xSeca ( Temperatura de saturación en K )
LAM H2 ( Humedad final del aire )
ABND
;
;

```

Por ejemplo calculemos la temperatura de saturación adiabática y la humedad respectiva, cuando el aire tiene una temperatura inicial de 65°C, una humedad de 0.0041 y una presión de 540 mm de Hg. Se quiere que el aire salga con un porcentaje de saturación del 93%:

```
65_°C 1_K CONVERT UVAL 0.0041 93 540 tsal
```

Con lo que se obtiene:

```
295.167445629
2.18624240089E-2
```

Que es la temperatura de saturación adiabática en K y la humedad final del aire (en kg de vap de H₂O/kg de aire seco).

Alternativamente si se toma como estado de referencia la temperatura final del aire, el balance de entalpía (ecuación 18) toma la siguiente forma:

$$C_{s1} * (T_1 - T_2) + \lambda_2 H_1 + \bar{C}_{pA} (T_2 - T_2)^0 * (H_2 - H_1) = C_{s2} * (T_2 - T_2)^0 + \lambda_2 H_2$$

$$\lambda_2 (H_1 - H_2) = -C_{s1} * (T_1 - T_2)$$

$$\frac{H_1 - H_2}{T_1 - T_2} = -\frac{C_{s1}}{\lambda_2} \quad (22)$$

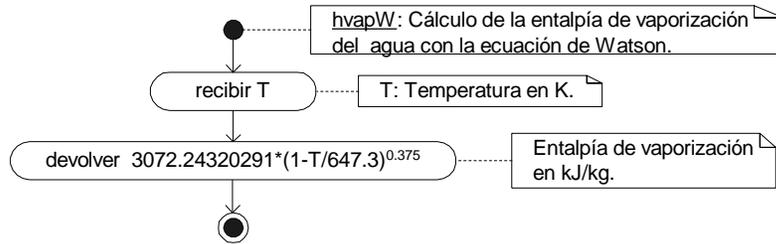
Donde ya no se requiere calcular la entalpía del agua, en contrapartida sin embargo se debe calcular la entalpía de vaporización a la temperatura de saturación adiabática (λ_2). Puesto que λ_2 y H_2 dependen de la temperatura de saturación adiabática (T_2), la ecuación (22) debe ser resuelta iterativamente.

Al resolver la ecuación (22), en cada iteración se debe calcular el valor de λ_2 . De las correlaciones disponibles emplearemos la ecuación de *Watson*:

$$\lambda_2 = \lambda_1 \left(\frac{1 - T_{r2}}{1 - T_{r1}} \right)^{0.375} \quad (23)$$

Donde T_{r2} es la temperatura reducida a la temperatura T_2 (T_2/T_c), λ_1 es el calor latente a la temperatura T_1 y T_{r1} la temperatura reducida respectiva (T_1/T_c). Puesto que conocemos el calor latente de vaporización a 0°C: 2501.4 kJ/kg, emplearemos este valor y su respectiva temperatura reducida $T_{r1} = 273.15/647.3 = 0.421983624285$ ($\Rightarrow \lambda_1/(1 - T_{r1})^{0.375} = 3072.24320291$) para calcular el calor latente de vaporización a la temperatura T_2 .

El algoritmo para resolver esta ecuación (para el caso específico del agua) se muestra en la siguiente página:



Y el código respectivo es:

```

xNAME hvapW ( Entalpía de vaporización del agua: ecuación de Watson)
:: ( Dato en la pila: T: Temperatura del agua en K )
  CK1&Dispatch
  ONE
  ::
  % 647.3 %/ %1- %CHS % 0.375 %^ % 3072.24320291 %*
  ; ( Entalpía de vaporización en kJ/kg )
  ;

```

Como ejemplo calculemos la entalpía de vaporización del agua a 78°F:

```
78_°F 1_K CONVERT UVAL hvapW
```

Con lo que obtenemos:

2435.9095747

Que es el calor latente de vaporización del agua a 78°F en kJ/kg.

Ahora podemos resolver la ecuación (22). Para ello la colocamos primero en la forma $f(T_2)=0$:

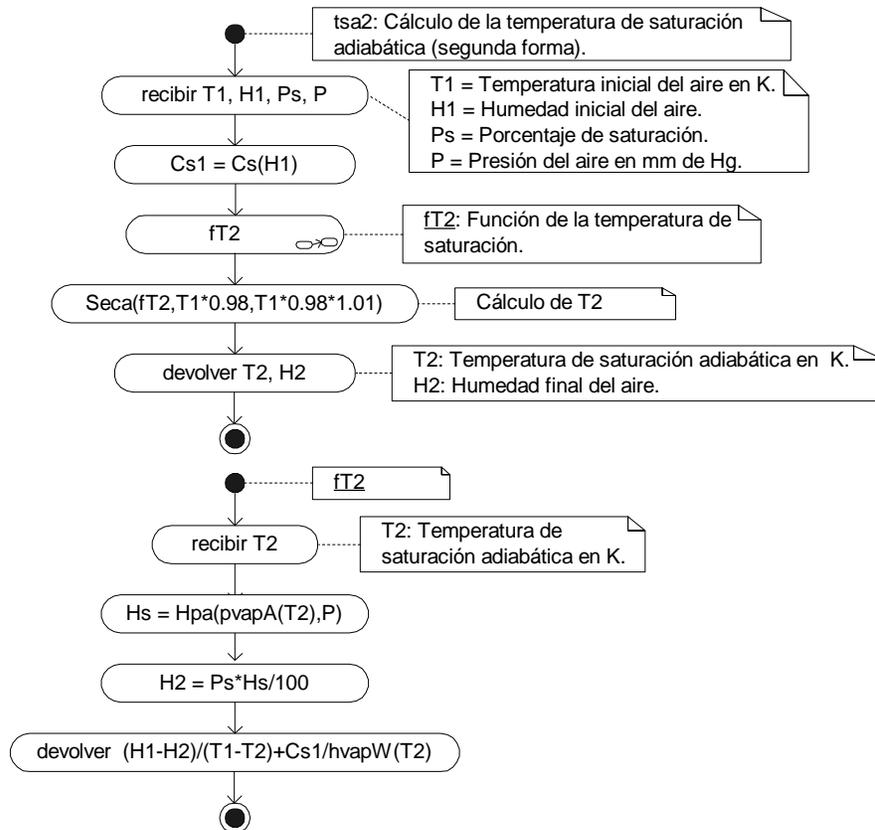
$$f(T_2) = \frac{H_1 - H_2}{T_1 - T_2} + \frac{C_{s1}}{\lambda_2} = 0 \tag{24}$$

El algoritmo que resuelve esta ecuación (con el método de la *Secante*) se muestra en la siguiente página y el código respectivo es el siguiente:

```

xNAME tsa2 ( Temperatura de saturación adiabática )
:: ( Datos en la pila: T1: Temperatura inicial en K; H1: Humedad inicial;
  Ps: Porcentaje de saturación; P: Presión del aire en mm de Hg )
  CK4&Dispatch
  # 1111
  ::
  %0 %0
  { LAM T1 LAM H1 LAM Ps LAM P LAM Cs1 LAM H2 }
  BIND
  LAM H1 xCs ' LAM Cs1 STO
  '
  :: ( ft2 )
  1LAMBIND ( T2 = 1 )
  1GETLAM xpvapA LAM P xHpa
  LAM Ps %* %100 %/ ' LAM H2 STO
  LAM H1 LAM H2 %- LAM T1 1GETLAM %- %/ LAM Cs1 1GETLAM xhvapW %/ %+
  ABND
  ;
  LAM T1 % .98 %* DUP % 1.01 %* xSeca ( Temperatura de saturación en K )
  LAM H2 ( Humedad final del aire )
  ABND
  ;
  ;

```



Como ejemplo volvamos a calcular la temperatura de saturación adiabática y la humedad respectiva, cuando el aire tiene una temperatura inicial de 65°C, una humedad de 0.0041 y una presión de 540 mm de Hg y debe ser saturado en un 93%:

```
65_°C 1_K CONVERT UVAL 0.0041 93 540 tsa2
```

Con lo que se obtiene

295.189569418

2.18933324433E-2

Que como vemos difieren un tanto de los resultados obtenidos con "tsa1". Se puede considerar que los resultados obtenidos con "tsa2" son más exactos, porque la ecuación de Watson es en general más exacta que emplear capacidades caloríficas medias. No obstante, ambos programas devuelven resultados lo suficientemente precisos para la mayoría de los cálculos ingenieriles, donde por lo general las temperaturas se miden con no más de un dígito después del punto. Por otra parte la primera forma (tsa1) es en general más estable que la segunda (tsa2) pues logra convergencia en la mayoría de los casos.

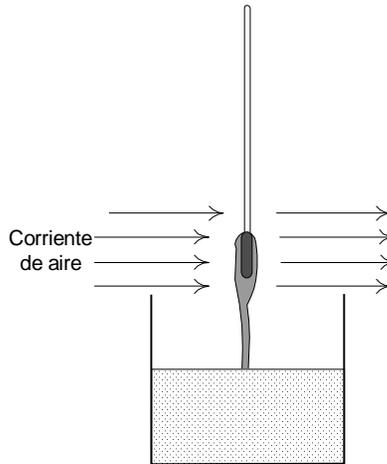
2.9.1. Ejercicios

- 45. Una corriente de aire a 87.8°C con una humedad H=0.030 kg H₂O/kg aire seco y a una presión de 101.325 kPa, se pone en contacto con agua en un saturador adiabático donde se enfría y humidifica hasta un 90% de saturación. Calcule la temperatura y humedad finales del aire. (Emplee tsa1 y tsa2).
- 46. Repita el ejercicio anterior si el aire se satura hasta un 100%.

- 47. Calcule la entalpía de vaporización del agua a 560°R.
- 48. Calcule la temperatura y humedad finales del aire que ingresa a un saturador adiabático a una temperatura de 70°F, tiene una humedad de 0.023 kg H₂O/kg aire seco, una presión de 12.5 psi y sale con un porcentaje de saturación del 80%. (Emplee t_{sa1} y t_{sa2}).
- 49. Calcule la temperatura y humedad finales del aire que ingresa a un saturador adiabático a una temperatura de 599.67°R, tiene una temperatura de rocío de 110°F, una presión de 0.78 bar y sale con un porcentaje de saturación del 90%. (Emplee t_{sa1} y t_{sa2}).
- 50. El aire que ingresa a una cámara de saturación adiabática tiene una temperatura de 32.2°C, una presión de 1.033 kgf/cm² y un porcentaje de humedad del 65%. Dicho aire ingresa a un saturador adiabático y luego es calentado en un intercambiador de calor hasta 23.9°C. El aire final tiene un porcentaje de humedad del 40%. Calcule la humedad inicial así como: la humedad, humedad relativa, temperatura de rocío, calor sensible y entalpía del aire final.

2.10. Temperatura de bulbo húmedo

La temperatura de bulbo húmedo es el método empírico más empleado para determinar la humedad del aire. La temperatura de bulbo húmedo se mide haciendo circular aire sobre un bulbo humedecido con un paño o algodón que se encuentra en contacto con una fuente que provee suministro constante de agua, tal como se muestra en la figura:



A medida que el aire circula la temperatura del bulbo comienza a descender hasta que alcanza un valor estable que corresponde a la temperatura de bulbo húmedo.

El estado estacionario se alcanza cuando la transferencia de calor desde el aire hacia el bulbo iguala al calor absorbido por el agua que se evapora desde la superficie del bulbo. Como la masa de aire es muy grande, comparada con la cantidad de agua que se evapora, el cambio en la humedad es despreciable y por lo tanto se puede asumir que la humedad del aire es constante. Se asume también que el aire que rodea al bulbo está saturado.

El calor absorbido por el agua que se evapora es:

$$q = M_A N_A (\lambda_w + c_{pA} (T - T_w)) A \tag{25}$$

Donde M_A es el peso molecular del agua; λ_w es la temperatura de evaporación a T_w ; T_w es la temperatura de bulbo húmedo; C_{pA} es la capacidad calorífica del agua; T es la temperatura del aire; A es el área de evaporación; N_A es el flujo superficial es vapor y está dada por la siguiente ecuación:

$$N_A = k_y (y_w - y) \quad (26)$$

Donde k_y es el coeficiente de transferencia de masa; y_w es la fracción molar de vapor agua en la superficie del bulbo; y es la fracción molar de vapor de agua en la masa de aire.

Puesto que λ_w es mucho mayor a $C_{pA}(T-T_w)$, este término puede ser despreciado, entonces reemplazando la ecuación (26) en la ecuación (25) se tiene:

$$q = M_A k_y (y_w - y) \lambda_w A \quad (27)$$

Las fracciones molares pueden ser colocadas en función de la humedad empleando los pesos moleculares del agua (M_A) y del aire (M_B):

$$y = \frac{\text{Moles de agua}}{\text{Moles totales de aire}} = \frac{\frac{H}{M_A}}{\frac{1}{M_B} + \frac{H}{M_A}} \quad (28)$$

Pero el como $1/M_B$ es mayor a H/M_A , este término puede ser despreciado, con lo que la ecuación (28) se reduce a:

$$y = \frac{\frac{H}{M_A}}{\frac{1}{M_B}} = \frac{HM_B}{M_A} \quad (29)$$

Reemplazando esta ecuación en la ecuación (27) resulta:

$$q = M_A k_y \left(\frac{H_w M_B}{M_A} - \frac{HM_B}{M_A} \right) \lambda_w A = k_y M_B (H_w - H) \lambda_w A \quad (30)$$

Donde H_w es la humedad del aire en la superficie del bulbo húmedo (que como dijimos está saturado).

Por otra parte, el calor cedido por el aire es:

$$q = h(T - T_w)A \quad (31)$$

Donde h es el coeficiente de transferencia de calor por convección.

Igualando las ecuaciones 30 y 31 resulta:

$$k_y M_B (H_w - H) \lambda_w A = h(T - T_w)A$$

$$\frac{(H_w - H)}{(T - T_w)} = \left(\frac{h}{k_y M_B} \right) \left(\frac{1}{\lambda_w} \right)$$

$$\frac{(H_w - H)}{(T_w - T)} = - \frac{h}{k_y M_B \lambda_w} \quad (32)$$

Comparando la ecuación (32) con la ecuación (22) se observa que si $C_s = h/k_y M_B$, entonces las dos ecuaciones son iguales. Se ha comprobado experimentalmente que en el caso del aire-agua esta relación se cumple bastante

bien, por lo que para el caso específico del aire, el cálculo de la humedad puede ser llevado a cabo con la ecuación (22), donde $T_2=T_w$, el porcentaje de saturación es del 100% y la incógnita es H_1 (que es la humedad del aire):

$$\frac{H_1 - H_2}{T_1 - T_2} = -\frac{C_{s1}}{\lambda_2} = -\frac{1.005 + 1.88H_1}{\lambda_2}$$

$$\frac{H_1 - H_2}{T_1 - T_2} \lambda_2 = -1.005 - 1.88H_1$$

$$H_1 \frac{\lambda_2}{T_1 - T_2} - H_2 \frac{\lambda_2}{T_1 - T_2} = -1.005 - 1.88H_1$$

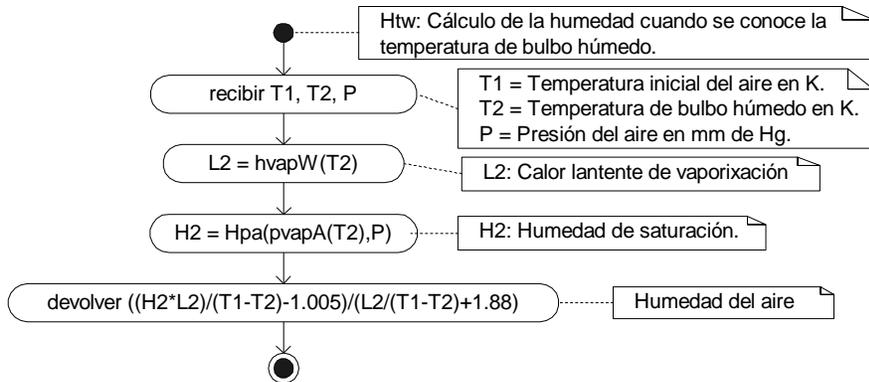
$$H_1 \frac{\lambda_2}{T_1 - T_2} + 1.88H_1 = \frac{H_2 \lambda_2}{T_1 - T_2} - 1.005$$

$$H_1 \left(\frac{\lambda_2}{T_1 - T_2} + 1.88 \right) = \frac{H_2 \lambda_2}{T_1 - T_2} - 1.005$$

$$H_1 = \frac{\frac{H_2 \lambda_2}{T_1 - T_2} - 1.005}{\frac{\lambda_2}{T_1 - T_2} + 1.88} \tag{33}$$

La relación $h/k_y M_B$ se conoce como la *razón psicrométrica* y para mezclas diferentes a las del aire-agua debe ser determinada experimentalmente o, en caso de estar disponibles, calculada con correlaciones empíricas.

El algoritmo que resuelve la ecuación (33) para calcular la humedad (H_1) cuando se cuenta con la temperatura de bulbo húmedo ($T_w=T_2$) es el siguiente:



Y el código respectivo:

```

xNAME Htw ( Cálculo de la humedad cuando se conoce la temperatura de
            bulbo húmedo )
:: ( Datos en la pila: T1: Temperatura del aire; T2: Temperatura de
    bulbo húmedo; P: presión del aire en mm de Hg )
CK3&Dispatch
# 111
:: ( T1=5 ; T2=4 ; P=3 ; L2=2 ; H2=1 )
%0 %0
' NULLLAM FIVE NDUPN DOBIND
4GETLAM xhvapW 2PUTLAM ( L2 = Entalpia de vaporización )
4GETLAM xpvapA 3GETLAM xHpa 1PUTLAM ( H2 = Humedad de saturación )
1GETLAM 2GETLAM %* 5GETLAM 4GETLAM %- %/ % 1.005 %-
    
```

```

2GETLAM 5GETLAM 4GETLAM %- %/ % 1.88 %+ %/
;
;

```

Como ejemplo calculemos la humedad del aire que se encuentra a 60°C, tiene una presión de 760 mm de Hg y una temperatura de bulbo húmedo de 29.5°C:

```
60_°C 1_K CONVERT UVAL 29.5_°C 1_K CONVERT UVAL 760 Htw
```

Con lo que resulta:

```
.013276281958
```

Que es la humedad del aire en kg H₂O/kg aire seco.

El proceso contrario, es decir el cálculo de la temperatura de bulbo húmedo (T_w) cuando se conoce la temperatura, humedad y presión del aire, es simplemente el cálculo de la temperatura de saturación adiabática cuando el porcentaje de saturación es del 100%.

Así por ejemplo si el aire se encuentra a una temperatura de 25°C, tiene una humedad de 0.00356 kg H₂O/Kg aire seco y una presión de 0.83 bares, la temperatura de bulbo húmedo es:

```
25_°C 1_K CONVERT UVAL 0.00356 100 0.83_bar 1_mmHg CONVERT UVAL tsa2
```

Con lo que se obtiene:

```
283.649557676
```

```
9.48764757567E-3
```

Que es la temperatura de bulbo húmedo (10.499557676°C) y la humedad de saturación a esa temperatura.

2.10.1. Ejercicios

51. El aire de un depósito se encuentra a 68°F, una presión de 11.6 psi y una temperatura de bulbo húmedo de 50°F. Calcule la humedad, porcentaje de humedad, humedad relativa, temperatura de rocío, volumen húmedo, calor sensible y entalpía del aire en el depósito.
52. El aire de una habitación se encuentra a 25°C y una presión de 570 mm de Hg. Al disminuir lentamente la temperatura de la habitación los vidrios comienzan a empañarse a 15°C. Calcule la humedad, porcentaje de humedad, humedad relativa, temperatura de rocío, volumen húmedo, calor sensible y entalpía del aire de la habitación.
53. El aire empleado en un proceso industrial se encuentra a 40°C, tiene una humedad de 0.0046 kg H₂O/kg aire seco y una presión de 0.83 atm. Calcule la temperatura de bulbo húmedo, el porcentaje de humedad, la humedad relativa, la temperatura de rocío, volumen húmedo, calor sensible y entalpía de este aire.
54. En un proceso industrial, el aire que se encuentra a una temperatura de 37.8°C, una presión de 0.77 kgf/cm² y una temperatura de bulbo húmedo de 26.7°C, será secado enfriándolo primero a 15.6°C, para condensar vapor de agua y calentándolo luego a 23.9°C. Calcule: a) la humedad, el porcentaje de humedad, la humedad relativa, el volumen húmedo y la entalpía de aire inicial; b) la humedad, el porcentaje de humedad, la humedad relativa, el volumen húmedo y la entalpía del aire final; c) El calor que debe ser removido en el enfriador por kg de aire seco; d) El calor que debe ser añadido en el calentador por kg de aire seco.

55. El aire que ingresa a una cámara de enfriamiento adiabática se encuentra a 570 mm de Hg, 27°C y tiene una temperatura de bulbo húmedo de 19.8°C. Cuando el aire sale de la cámara es calentado hasta 35.2°C, temperatura a la cual la presión parcial del vapor de agua es de 15.4 mm de Hg. Calcule: a) la humedad, porcentaje de humedad, humedad relativa, temperatura de rocío, presión parcial del vapor de agua, volumen húmedo y entalpía del aire inicial; b) La temperatura, humedad, temperatura de rocío, presión parcial, volumen húmedo y entalpía del aire a la salida de la cámara; c) la humedad, temperatura de rocío, temperatura de bulbo húmedo, volumen húmedo y entalpía del aire a la salida del calentador.

3. TORRES DE ENFRIAMIENTO DE AGUA

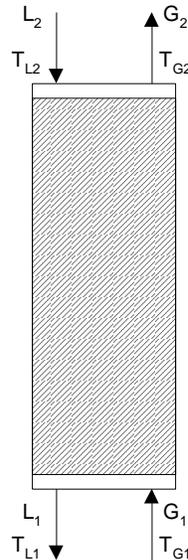
Las torres de enfriamiento se emplean principalmente para enfriar el agua que proviene de los equipos de intercambio de calor.

Como se sabe el agua que se emplea en dichos equipos debe ser tratada (ablandada) para evitar o al menos disminuir la formación de costras. Las torres de enfriamiento permiten reutilizar el agua tratada disminuyendo considerablemente las necesidades de agua fresca, ello no sólo implica un ahorro sino que además disminuye la contaminación térmica que de otra forma se produciría.

En las torres de enfriamiento se aprovecha el fenómeno que se produce cuando un líquido relativamente caliente se pone en contacto con un gas que no está saturado: parte del líquido se evapora y pasa a la fase gaseosa. Como la evaporación es uno de los cambios de estado que más calor consume se logra un enfriamiento importante evaporando cantidades relativamente pequeñas de líquido.

Si bien las torres de enfriamiento se construyen de diferentes formas y empleando diferentes tipos de relleno, prácticamente en todas ellas el agua ingresa por la parte superior y cae por gravedad, mientras que el aire es alimentado en contracorriente por la parte inferior, casi siempre con la ayuda de un equipo de ventilación.

El esquema general de una torre de enfriamiento es el siguiente:

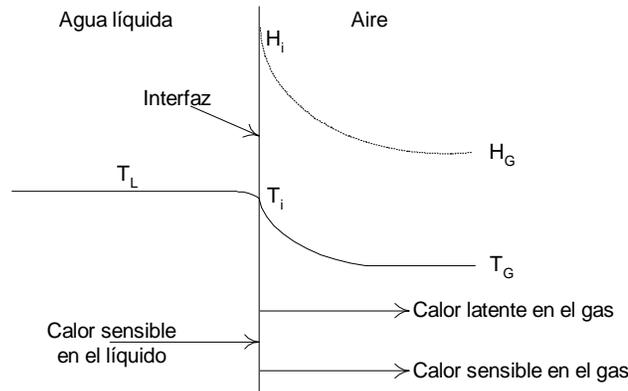


Donde L es el flujo de líquido y G el flujo de gas.

El perfil de temperaturas y concentraciones en la interfaz gas-líquido se presenta en el diagrama de la siguiente página. Como se puede observar, el vapor de agua se difunde hacia la masa de gas gracias a la diferencia de humedades que existen entre la interfaz (H_i) y la masa de aire (H_G). En la interfaz gas-líquido el aire está saturado, por lo tanto la humedad en la interfaz (H_i) es la humedad de saturación a la temperatura T_i . En el líquido no existe diferencia de concentraciones pues se trata de agua pura.

Por otra parte el calor se difunde desde la masa de líquido hacia la masa de gas gracias a las diferencias de temperaturas. En el interior del líquido el calor se transfiere por convección desde la masa de líquido hacia la in-

terfaz, gracias a la diferencia que existe entre la temperatura de la masa de líquido (T_L) y la temperatura en la interfaz (T_i), posteriormente el calor se transfiere hacia la masa de aire gracias a la diferencia que existe entre la temperatura de la interfaz (T_i) y la temperatura de la masa de aire (T_G).



El perfil de temperaturas que se muestra en la figura corresponde a la parte superior de la columna, donde la temperatura del agua es mayor a la temperatura del aire, no obstante en la parte inferior de la columna la temperatura del agua puede ser inferior a la del aire, aunque nunca menor a su temperatura de bulbo húmedo. Cuando esto ocurre el flujo de calor sensible se invierte y el aire cede calor al agua por convección aunque sigue ganando calor por evaporación.

3.1. Altura de una torre de enfriamiento

Tomando como base el esquema de la torre, presentado en la página anterior y realizando el balance de materia y energía en una altura diferencial de la misma, se puede deducir la ecuación que permite calcular la altura de una torre de enfriamiento.

La ecuación resultante es la siguiente^[1]:

$$z = \frac{G}{M_B k_G a P} \int_{H_{y1}}^{H_{y2}} \left(\frac{1}{H_{yi} - H_y} \right) dH_y \quad (1)$$

En la deducción de esta ecuación se asume que el flujo de líquido (L) es constante, lo que realmente no es cierto pues en el proceso se pierde agua como vapor, sin embargo, la masa evaporada es pequeña con relación a la masa total, por lo que esta asunción es válida. Igualmente se asume que la relación psicrométrica es igual a uno, asunción que como vimos en el capítulo anterior, es válida para el caso de la mezcla aire-agua.

En la ecuación (1) "z" es la altura de la torre de enfriamiento en m, "G" es el flujo superficial de aire seco en kg/s.m², "H_y" es la entalpía del aire, "M_B" es el peso molecular del aire (28.97), "P" es la presión en Pa y "k_Ga" es el coeficiente volumétrico de transferencia de masa en kgmol/s.m³.Pa. El subíndice "i" hace referencia a las condiciones en la interfaz gas-líquido, así "H_{yi}" es la entalpía en la interfaz.

La ecuación que relaciona la temperatura en la interfaz "T_i" con la entalpía en la interfaz "H_{yi}" es la siguiente:

$$\frac{-h_L a}{M_B k_G a P} = \frac{H_{yi} - H_y}{T_i - T_L} \quad (2)$$

Donde " h_{La} " es el coeficiente volumétrico de transferencia de calor en $\text{kJ/s.m}^3.\text{K}$ y " T " es la temperatura en K .

La ecuación del balance de energía, deducida tomando como sistema la parte inferior de la torre es:

$$G(H_y - H_{y1}) = LC_L(T_L - T_{L1}) \quad (3)$$

Donde " L " es el flujo superficial de líquido en kg/s.m^2 , " C_L " es la capacidad calorífica media del agua en kJ/kg.K (4.187). El subíndice " L " hace referencia al líquido y el subíndice " 1 " a las condiciones en la parte inferior de la torre.

Por lo general, en el cálculo de la altura de una torre de enfriamiento son datos conocidos el flujo superficial de agua " $L_2 = L_1 = L$ ", la temperatura inicial del agua " T_{L2} ", la temperatura final del agua " T_{L1} ", la presión de operación " P ", el flujo superficial de aire " G ", la temperatura inicial del aire " T_{G1} ", la humedad inicial del aire " H_1 " (o algún otro dato que permita calcularla), el coeficiente volumétrico de transferencia de masa " k_{Ga} " y el coeficiente volumétrico de transferencia de calor " h_{La} ".

Entonces con T_{G1} , H_1 , P y el programa H_y se puede calcular la entalpía inicial del aire H_{y1} .

La entalpía final del aire H_{y2} puede ser calculada con la ecuación del balance de materia (ecuación 3), despejando de la misma H_y :

$$H_y = \frac{LC_L}{G}(T_L - T_{L1}) + H_{y1} \quad (4)$$

Aplicando esta ecuación al punto 2 se tiene:

$$H_{y2} = \frac{LC_L}{G}(T_{L2} - T_{L1}) + H_{y1} \quad (5)$$

3.1.1. Procedimiento numérico para el cálculo de la entalpía en la interfaz

Para calcular la altura de la torre con la ecuación (1), se debe resolver la integral y para ello es necesario calcular los valores de la entalpía en la interfaz gas-líquido (H_{yi}). Dichos valores se calculan resolviendo la ecuación (2), pero como la integral es función de la entalpía (H_y), la temperatura del líquido (T_L) es desconocida, razón por la cual debe ser colocada en función de la entalpía. Para ello se vuelve a recurrir a la ecuación del balance de materia (ecuación 3), pero ahora se despeja la temperatura del líquido:

$$T_L = \frac{G}{LC_L}(H_y - H_{y1}) + T_{L1} \quad (6)$$

Puesto que en la interfaz gas-líquido, el aire está saturado H_i puede ser calculada con la temperatura de la interfaz (T_i), la presión de operación (P) y el programa H_{sat} . En consecuencia la ecuación (2) es función de una sola variable: T_i y como tal puede ser resuelta recurriendo a uno de los métodos iterativos programados en el anterior capítulo.

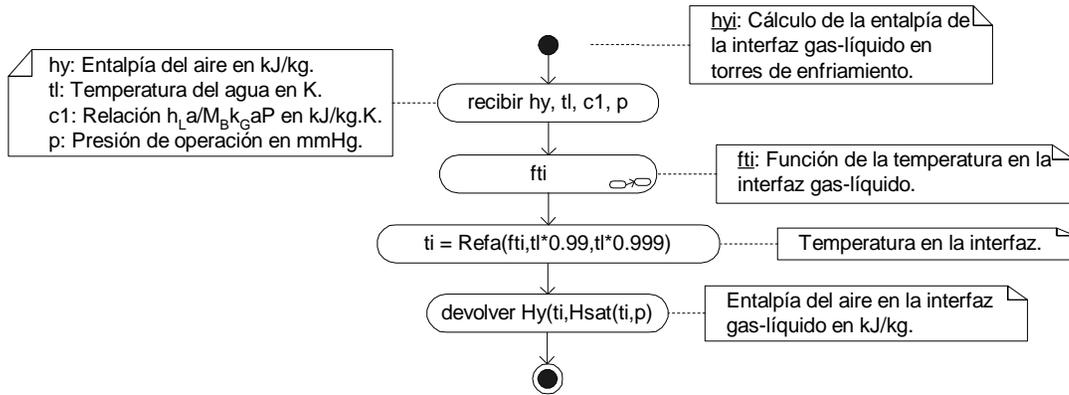
La función a resolver es la siguiente:

$$f(T_i) = \frac{H_{yi} - H_y}{T_i - T_L} + \frac{h_{La}}{M_B k_G a P} = 0 \quad (7)$$

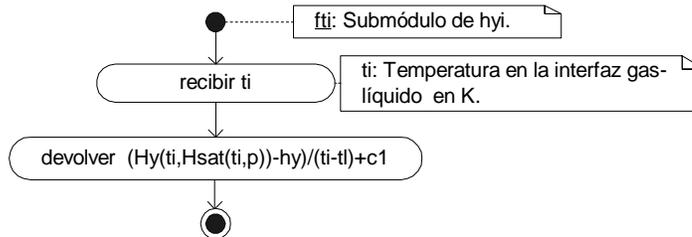
$$H_{yi} = H_{sat}(T_i, P)$$

Donde la relación $h_{La}/M_Bk_{Ga}P$ es constante y como tal no debe ser calculada en el proceso iterativo. Los valores de H_y son valores conocidos pues son generados o fijados en el cálculo de la integral. Los valores de T_L son igualmente conocidos pues se calculan con la ecuación (6) y los respectivos valores de H_y .

El algoritmo que resuelve la ecuación (7) empleando el método de la *Regula Falsi* se presenta en el siguiente diagrama de actividades:



Donde el algoritmo del subprograma *fti* es:



Se ha empleado el método de *Regula Falsi* porque la función es muy sensible al valor inicial asumido y si el mismo no es muy cercano al valor final, no converge con los métodos de *Newton - Raphson* y la *Secante*. Se puede emplear el método de *Regula Falsi* pues se sabe aproximadamente los límites entre los que se encuentra la solución, dado que la temperatura en la interfaz casi nunca es menor al 99% ni mayor al 99.9% de dicha temperatura.

Para escribir el código de los algoritmos de este capítulo cree un nuevo archivo dentro del proyecto *PRQ205*: el archivo *Torres_Enf.S* (*Project -> File -> New Source File -> Torres_Enf.S*).

El código de los anteriores algoritmos (que debe ser escrito en el archivo *Torres_Enf.s*) es el siguiente:

```
xNAME hyi ( cálculo de la entalpía en la interfaz gas - líquido )
:: ( datos en la pila: hy=entalpía del aire; tl= temperatura del agua )
  CK4&Dispatch ( c1= relación hLa/MbKgaP; P= Presión)
  # 1111
  ::
  { LAM hy LAM tl LAM c1 LAM p }
  BIND
  '
  :: ( función de la temperatura en la interfaz : fti )
    DUPDUP LAM p xHsat ( humedad en la interfaz=hi )
    xHy ( entalpía en la interfaz=hyi )
    LAM hy %- SWAP LAM tl %- %/ LAM c1 %+ ( fti=0 )
  ;
```

```

LAM t1 % 0.99 %* LAM t1 % 0.999 %*
xRefa ( cálculo de la temperatura en la interfaz = ti )
DUP LAM p xHsat xHy ( entalpía en la interfaz )
ABND

```

```

;
;

```

3.1.1.1. Ejemplo

Una torre de enfriamiento trabaja con un flujo superficial de aire seco igual a 1356 kg/s.m² e igual flujo superficial de agua. El agua ingresa a 43.3°C y es enfriada hasta 29.4°C. El aire ingresa a 29.4°C y tiene una temperatura de bulbo húmedo igual a 23.9°C, siendo la presión de operación 1 atm. Se ha estimado que la relación $h_{La}/M_B K_{Ga}P$ para la torre es igual a 41.87 kJ/kg.K. Calcule la temperatura de la interfaz para valores de H_y iguales a 71.7, 83.3 94.9 106.5, 118.4 y 129.9 kJ/kg.

Solución

Los datos conocidos son:

```

G = 1356 kg/s.m2
L = 1356 kg/s.m2
TL2 = 43.3 °C 1_K CONVERT = 316.45_K
TL1 = 29.4 °C 1_K CONVERT = 302.55_K
TG1 = 29.4 °C 1_K CONVERT = 302.55_K
TW1 = 23.9 °C 1_K CONVERT = 297.05_K
P = 1_atm 1_mmHg CONVERT = 760_mmHg
hLa/MBKGaP = 41.87 kJ/kg.K
Hy = 71.7, 83.3 94.9 106.5, 118.4, 129.9 kJ/kg.

```

Se sabe además que C_L es igual a 4.187 kJ/kg.

Para cada uno de los valores de entalpía se deben calcular los correspondientes valores de la temperatura del agua (TL) empleando la ecuación (6).

Con la temperatura (T_{G1}), temperatura de bulbo húmedo (T_{W1}) y al presión del aire y el programa H_{tw} se puede calcular la humedad inicial (H_1) y con la temperatura y humedad iniciales y el programa H_y la entalpía inicial del aire (H_{y1}).

El programa que resuelve el problema es el siguiente:

```

« [71.7 83.3 94.9 106.5 118.4 129.9]
1356 1356 4.187 302.55 302.55
297.05 760 41.87 0 0 0
→ hy G L CL TL1 TG1 TW1 P C1 H1 HY1 C2
« TG1 TW1 P Htw 'H1' STO
TG1 H1 Hy 'HY1' STO
G L CL * / 'C2' STO
1 6 FOR i
hy i GET
C2 hy i GET HY1 - * TL1 +
C1 P hyi
NEXT
6 →LIST
{ Hyi1 Hyi2 Hyi3 Hyi4 Hyi5 Hyi6 } →TAG
LIST→ DROP
»
»

```

Haciendo correr el programa se obtienen los siguientes resultados en alrededor de 9 segundos:

Hy1: 94.186491947
Hy2: 108.5514964
Hy3: 124.555091305
Hy4: 142.36307454
Hy5: 162.686022378
Hy6: 184.478492336

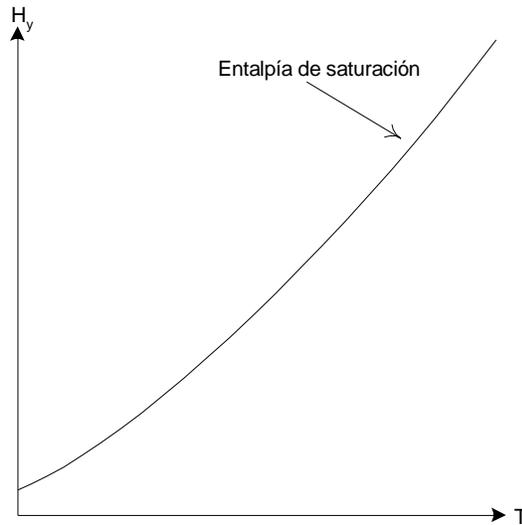
Que son los valores de la entalpía en la interfaz gas-líquido para cada una de las entalpías dadas.

3.1.1.2. Ejercicio

1. En una torre de enfriamiento se enfría agua desde 110°F hasta 85°F empleando aire a una temperatura de 85°F, con una temperatura de bulbo húmedo de 75°F y una presión de 79993 Pa. El flujo superficial del agua es de 2000 lbm/h.pie² y el del aire 1400 lbm/h.pie². La relación h_{LA}/M_{BkGaP} para la torre es 17.71 Btu/lb.°F. Calcule la entalpía en la interfaz gas líquido para entalpías del aire iguales a: 35.1, 39.2, 45.3, 50.2, 54.9, 61.8, 65.6 y 69 Btu/lb.

3.1.2. Procedimiento gráfico para el cálculo de la entalpía en la interfaz

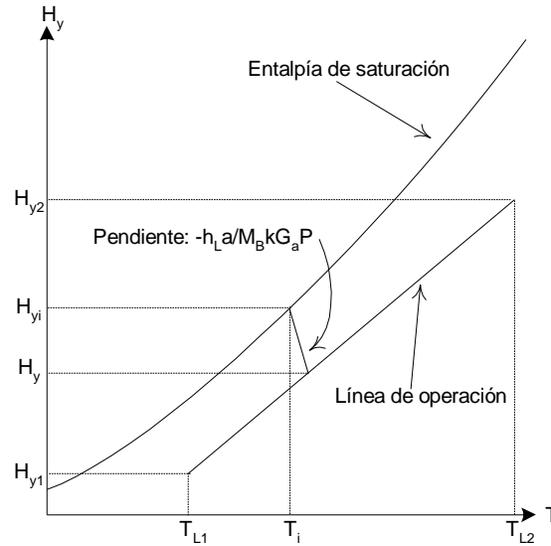
La entalpía en la interfaz puede ser calculada también gráficamente. Para ello se traza primero la gráfica de entalpía de saturación versus la temperatura. Los datos de temperatura y entalpía de saturación pueden ser obtenidos con los programas *Hsat* y *Hy*. la gráfica tiene la apariencia que se muestra en la siguiente figura:



Luego se traza la línea de operación, para ello se traza una línea recta que pasa por los formados por: a) el punto correspondiente a la entalpía inicial del aire H_{y1} y la temperatura final del agua T_{L1} y b) el punto correspondiente a la entalpía final del aire H_{y2} (calculada con la ecuación 5) y la temperatura inicial del agua T_{L2} .

Finalmente se encuentra la entalpía (y temperatura) de la interfaz, para una entalpía H_y dada, trazando una línea recta de pendiente $-h_{LA}/M_{BkGaP}$, comenzando en la intersección de la entalpía H_y y la línea de operación y con-

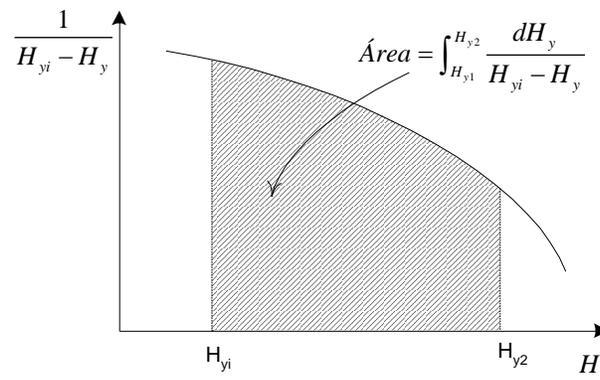
tinuando hasta la curva correspondiente a la entalpía de saturación, tal como se muestra en la siguiente figura:



3.1.3. Cálculo de la altura de la torre de enfriamiento

Una vez calculada la entalpía en la interfaz gas - líquido se puede calcular la altura de la torre con la ecuación (1). Para resolver esta ecuación es necesario realizar a una integración gráfica o numérica.

Gráficamente se traza la curva $1/(H_{yi}-H_y)$ versus la H_y y se mide el área bajo la curva comprendida entre H_{y1} y H_{y2} , tal como se muestra en la siguiente figura:



Numéricamente se puede recurrir a uno de los métodos de integración como el de *Simpson* y el del *Trapezio*. En general el método de *Simpson* es más exacto que el del *Trapezio*, pero existen ocasiones donde la curva se aproxima más a una línea recta o donde existen cambios bruscos de pendiente, donde el método del *Trapezio* resulta más exacto que el método de *Simpson*, por esta razón, presentamos los programas para ambos métodos, sin entrar en detalles pues los mismos ya han sido estudiados en otras materias.

Los programas que se presentan a continuación deben ser añadidos al archivo *Metodos1* del proyecto *PRQ205*.

```
xNAME Trapf ( Integración de funciones por el método del Trapecio)
::
  CK4&Dispatch
```

```

# 8111 ( Datos en la pila: f=función a integrar; a,b: límites )
::      ( n: Número de segmentos )
%0 %0 %0 ( f=7; a=6; b=5; n=4; h=3; s=2; x=1 )
' NULLLAM SEVEN NDUPN DOBIND
5GETLAM 6GETLAM %- 4GETLAM %/ 3PUTLAM ( incremento del segmento)
6GETLAM 1PUTLAM ( valor inicial de x )
4GETLAM COERCE ONE DO
    1GETLAM 3GETLAM %+ 1PUTLAM ( nuevo valor de x )
    2GETLAM 1GETLAM 7GETLAM EVAL %+ 2PUTLAM ( sumatoria )
LOOP
3GETLAM 6GETLAM 7GETLAM EVAL %2 2GETLAM %* %+ 5GETLAM
7GETLAM EVAL %+ %* %2 %/ ( integral )
ABND
;
;

xNAME Simpf ( Integración de funciones por el método de Simpson)
::
CK4&Dispatch
# 8111 ( Datos en la pila: f=función a integrar; a,b: límites )
::      ( n: Número de segmentos )
%0 %0 %0 %0 ( f=8; a=7; b=6; n=5; h=4; si=3; sp=2; x=1 )
' NULLLAM EIGHT NDUPN DOBIND
5GETLAM %2 %MOD %0<> IT :: 5GETLAM %1+ 5PUTLAM ; ( si n es impar: n=n+1)
6GETLAM 7GETLAM %- 5GETLAM %/ 4PUTLAM ( incremento del segmento)
7GETLAM 1PUTLAM ( valor inicial de x )
5GETLAM COERCE #1+ TWO DO
    1GETLAM 4GETLAM %+ 1PUTLAM ( nuevo valor de x )
    1GETLAM 8GETLAM EVAL
    INDEX@ #2 #/ DROP #0<>
    ITE :: 3GETLAM %+ 3PUTLAM ; :: 2GETLAM %+ 2PUTLAM ;
LOOP
4GETLAM 7GETLAM 8GETLAM EVAL %4 2GETLAM %* %+ %2 3GETLAM %* %+
6GETLAM 8GETLAM EVAL %+ %* %3 %/ ( integral )
ABND
;
;

```

Ambos programas reciben como datos la función a integrar (función de una variable), los límites inicial y final de integración y el número de segmentos.

Por ejemplo para integrar la siguiente función:

$$\int_{5.1}^{8.4} (3 - 6x^2 + 1.1x^{3.1}) dx$$

Escribimos el siguiente programa para el método del trapecio:

```

«
« → X
« 3. 6. X SQ * - 1.1 X 3.1 ^ * + »
» 5.1 8.4 50 Trapf
»

```

Al hacer correr este programa con *TEVAL* se obtiene:

```

528.77481474
s: 2.4104

```

Que es la integral y el tiempo empleado en calcularla.

Para el método de *Simpson* se tiene esencialmente el mismo programa:

```

«
« → X
  « 3. 6. X SQ * - 1.1 X 3.1 ^ * + »
» 5.1 8.4 10 Simpf
»
    
```

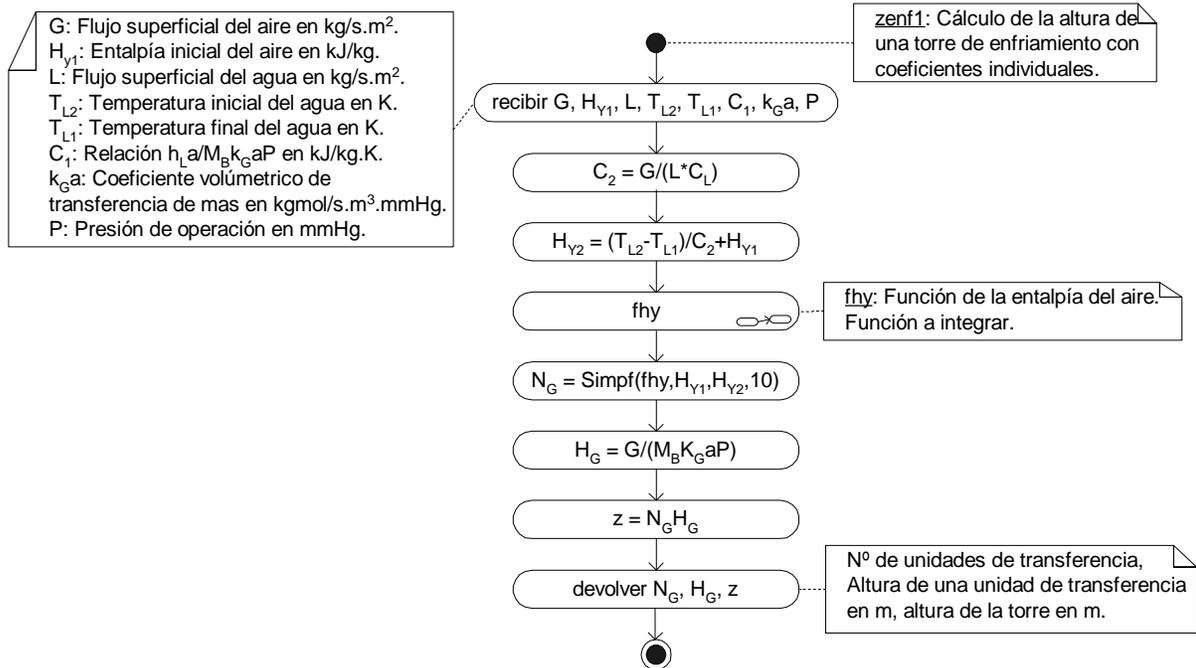
Al hacer correr el programa con *TEVAL* se obtiene:

```

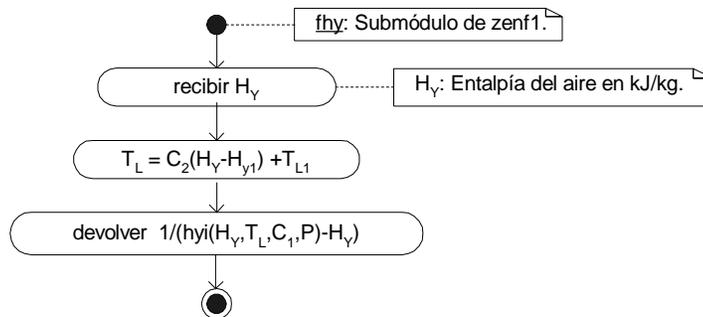
528.719058077
s: .5857
    
```

Que es un resultado más exacto que el obtenido con el método del *Trapezio* a pesar de haber empleado un menor número de segmentos.

Para el cálculo de la altura de la torre de enfriamiento emplearemos el método de *Simpson*, fijando el número de segmentos en 10. El algoritmo para el cálculo de la altura de la torre se presenta en el siguiente diagrama de actividades:



El algoritmo del submódulo "*fhy*" es el siguiente:



Como se puede observar en el algoritmo no sólo se devuelve la altura de la torre, sino también otros dos valores: *N_G* (número de unidades de transferencia) y *H_G* (altura de una unidad de transferencia), los mismos que serán explicados un poco más adelante.

El programa elaborado en base a estos algoritmos y que debe ser escrito en el archivo *Torres_Enf.S*, es el siguiente:

```
xNAME zenf1 ( cálculo de la altura de una torre de enfriamiento )
::      ( empleando coeficientes individuales )
%8 CKNNOLASTWD DROP
CK&DISPATCH1
# 11111
::
6ROLL 7ROLL 8ROLL
CK&DISPATCH1
# 111
::
8UNROLL 7UNROLL 6UNROLL
% 4.187 %0 %0 %0 %0
{ LAM G LAM HY1 LAM L LAM TL2 LAM TL1 LAM C1 LAM kga
  LAM P LAM CL LAM C2 LAM HY2 LAM HY LAM TL }
BIND
LAM G LAM L LAM CL %* %/ ' LAM C2 STO
LAM TL2 LAM TL1 %- LAM C2 %/ LAM HY1 %+ ' LAM HY2 STO
' ( función a integrar )
::
' LAM HY STO
  LAM C2 LAM HY LAM HY1 %- %* LAM TL1 %+ ' LAM TL STO
  LAM HY LAM TL LAM C1 LAM P xhyi LAM HY %- %1/
;
LAM HY1 LAM HY2 %10 xSimpf DUP ( N° de unidades de transferencia )
LAM G % 28.97 LAM kga %* LAM P %* %/ DUP ( Altura de una unidad )
ROT %* ( Altura de la torre )
ABND
;
;
;
```

3.1.3.1. Ejemplo

Una torre de enfriamiento trabaja con un flujo superficial de aire seco igual a 1.356 kg/s.m^2 e igual flujo superficial de agua. El agua ingresa a 43.3°C y debe ser enfriada hasta 29.4°C . El aire ingresa a 29.4°C y tiene una temperatura de bulbo húmedo igual a 23.9°C , siendo la presión de operación 1 atm . Se ha estimado que la relación $h_{1a}/M_B K_{Ga} P$ para la torre es igual a 41.87 kJ/kg.K y que el coeficiente volumétrico de transferencia de masa: k_{Ga} es $1.207 \times 10^{-7} \text{ kgmol/s.m}^3.\text{Pa}$. Calcule la altura que deberá tener la torre para conseguir el enfriamiento requerido.

Solución

Los datos son:

```
G = 1.356 kg/s.m2
L = 1.356 kg/s.m2
TL2 = 43.3 °C 1_K CONVERT = 316.45_K
TL1 = 29.4 °C 1_K CONVERT = 302.55_K
TG1 = 29.4 °C 1_K CONVERT = 302.55_K
Tw1 = 23.9 °C 1_K CONVERT = 297.05_K
P = 1_atm 1_mmHg CONVERT = 760_mmHg
h1a/MBKGaP = 41.87 kJ/kg.K
kGa = 1.207x10-7 kgmol/(s*m3*Pa) 1_kgmol/(s*m3*mmHg) CONVERT
1.60920098684E-5_kgmol/(s*m3*mmHg)
```

Con T_{G1} , T_{W1} y P calculamos la humedad inicial del aire:

$$H_1 = 302.55 \cdot 297.05 \cdot 760 \cdot H_{tw} = 1.62615985795E-2$$

Y con la humedad calculamos la entalpia inicial del aire:

$$H_{y1} = 302.55 \cdot 1.62615985795E-2 \cdot H_y = 71.1225737635 \text{ kJ/kg}$$

Entonces se cuenta con todos los datos para calcular la altura de la torre empleando el programa *zenfl*. El programa que resuelve el problema es e siguiente:

```
« 1.356 71.1225737635 1.356 316.45 302.55 41.87 1.60920098684E-5 760
zenfl
1_m →UNIT "z" →TAG 3 ROLLD
1_m →UNIT "HG" →TAG 3 ROLLD
"NG" →TAG 3 ROLLD
»
```

Haciendo correr el programa (con EVAL) se obtiene:

```
NG: 1.82060133875
HG: '3.82725423085_m'
z: '6.96790417642_m'
```

Por lo tanto la torre de enfriamiento deberá tener una altura de 6.97 m (prácticamente 7 m), para lograr el enfriamiento requerido.

3.1.3.2. Ejercicios

2. Se empleará una torre de enfriamiento para enfriar agua desde 110°F hasta 85°F empleando aire a una temperatura de 85°F , con una temperatura de bulbo húmedo de 75°F y una presión de 79993 Pa. El flujo superficial del agua es de 2000 lbm/h.pie^2 y el del aire 1400 lbm/h.pie^2 . La relación $h_{La}/M_{Bk_{Ga}P}$ para la torre es $17.71 \text{ Btu/lb.}^{\circ}\text{F}$ y el coeficiente volumétrico de transferencia de masa $6.90 \text{ lbmol/h.ft}^3.\text{atm}$. Calcule la altura que deberá tener la torre para conseguir el enfriamiento requerido.
3. Repita el ejercicio 2 empleando un flujo de aire igual a 2000 lbm/h.pie^2 .
4. Repita el ejercicio 2 si el aire ingresa a una temperatura de 95°F y tiene una temperatura de rocío de 50°F .
5. Repita el ejercicio 2 si el flujo de aire es de 1800 lbm/h.pie^2 e ingresa a una temperatura de 90°F con una humedad relativa del 30%.
6. Una industria cuenta con una torre de enfriamiento de 10 m de alto. La industria desea emplear este equipo para enfriar agua desde 40 hasta 25°C . Los ventiladores de la torre permiten alimentar un flujo de aire igual a 1.7 kg/s.m^2 . El aire se encuentra a una temperatura media de 19°C , una humedad relativa del 50% y la presión atmosférica es de 630 mm de Hg. Se sabe que la relación $h_{La}/M_{Bk_{Ga}P}$ para la torres es igual a 37.25 kJ/kg.K y que el coeficiente volumétrico de transferencia de calor k_{Ga} es igual a $1.123 \times 10^{-7} \text{ kgmol/s.m}^3.\text{Pa}$. Calcule el flujo de agua que puede ser enfriado en esta torre.

3.1.4. Cálculo de la altura empleando unidades de transferencia

Frecuentemente la ecuación para el cálculo de la altura de una torre de enfriamiento se expresa en la forma:

$$z = H_G * N_G \quad (8)$$

Donde H_G es la altura de una unidad de transferencia (en metros) y N_G es el número de unidades de transferencia. Esta es en realidad sólo otra forma de escribir la ecuación (1), con la única diferencia de que la misma ha sido reagrupada en dos términos:

$$H_G = \frac{G}{M_B k_G a P} \quad (9)$$

$$N_G = \int_{H_{y1}}^{H_{y2}} \frac{1}{H_{yi} - H_y} dH_y$$

Al ser la misma ecuación el procedimiento de cálculo no cambia, sólo que en este caso se devuelven además estos valores. Esto es algo que ya se ha considerado en el programa "zenfl" elaborado en el anterior acápite, pues el mismo devuelve también el número de unidades de transferencia N_G y la altura de una unidad de transferencia H_G .

3.1.5. Cálculo de la altura empleando coeficientes globales

En la mayoría de los casos sólo se cuenta con el coeficiente global volumétrico de transferencia de masa $K_G a$ (en $\text{kgmol/s.m}^3.\text{mmHg}$) y no así con los coeficientes individuales $k_G a$ y $h_L a$. En estos casos el diseño se lleva a cabo asumiendo que la entalpía en la interfaz (H_{yi}) corresponde al aire saturado a la temperatura (T_L), o lo que es lo mismo que la línea que se traza desde la línea de operación hasta la entalpía de saturación es vertical.

Bajo esta suposición la ecuación para el cálculo de la altura de la torre es:

$$z = \frac{G}{M_B K_G a P} \int_{H_{y1}}^{H_{y2}} \frac{1}{H_y^* - H_y} dH_y \quad (10)$$

Donde H_y^* es la entalpía del aire saturado a la temperatura T_L . El procedimiento de cálculo es esencialmente el mismo que en el caso anterior (cuando se cuenta con los coeficientes individuales), excepto que el cálculo de la entalpía en la interfaz (H_{yi}) es ahora directo (debido a la suposición efectuada).

De manera similar al cálculo con coeficientes individuales, la ecuación (10) puede ser escrita de la siguiente manera:

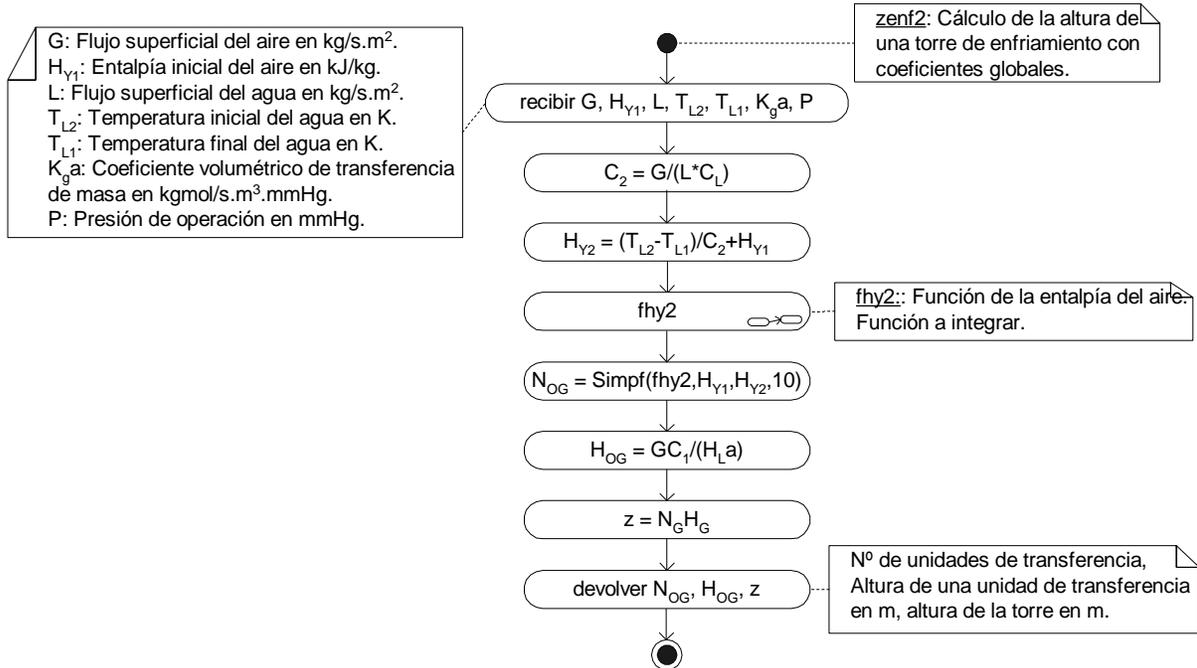
$$z = H_{OG} * N_{OG} \quad (11)$$

Donde H_{OG} es la altura (en m) de una unidad de transferencia basada en el coeficiente global de transferencia de masa y N_{OG} el número de unidades de transferencia. La relación de estos términos con los coeficientes empleados en la ecuación (10) es:

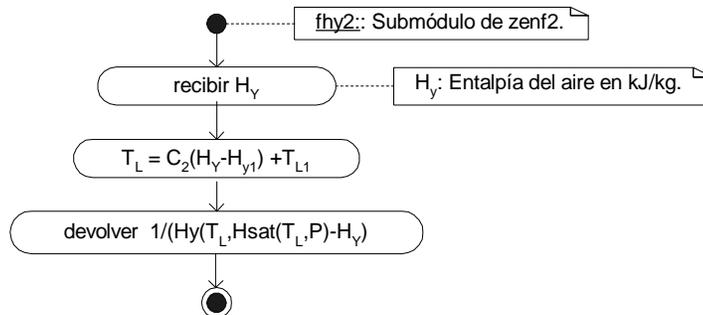
$$H_{OG} = \frac{G}{M_B K_G a P} \quad (12)$$

$$N_{OG} = \int_{H_{y1}}^{H_{y2}} \frac{1}{H_y^* - H_y} dH_y$$

El algoritmo que resuelve la ecuación (10) y por consiguiente la ecuación (12) se presenta en el siguiente diagrama de actividades.



Donde el algoritmo del submódulo *fhy2* es:



Por consiguiente en el programa elaborado para el caso anterior es suficiente modificar el cálculo de la entalpía de la interfaz, no siendo necesario emplear el programa "hyi" pues la humedad de saturación puede ser calculada con *Hsat* y *Hy*.

```

xNAME zenf2 ( cálculo de la altura de una torre de enfriamiento )
::
    ( empleando coeficientes globales )
    %7 CKNNOLASTWD DROP
    CK&DISPATCH1
    # 11111
    ::
    6ROLL 7ROLL
    CK&DISPATCH1
    # 11
    ::
    7UNROLL 6UNROLL
    % 4.187 %0 %0 %0 %0
    { LAM G LAM HY1 LAM L LAM TL2 LAM TL1 LAM Kga
      LAM P LAM CL LAM C2 LAM HY2 LAM HY LAM TL }
    BIND
    
```

```

LAM G LAM L LAM CL %* %/ ' LAM C2 STO
LAM TL2 LAM TL1 %- LAM C2 %/ LAM HY1 %+ ' LAM HY2 STO
' ( función a integrar )
::
' LAM HY STO
LAM C2 LAM HY LAM HY1 %- %* LAM TL1 %+ ' LAM TL STO
LAM TL LAM TL LAM P xHsat xHy LAM HY %- %1/
;
LAM HY1 LAM HY2 %10 xSimpf DUP ( N° de unidades de transferencia )
LAM G % 28.97 LAM Kga %* LAM P %* %/ DUP ( Altura de una unidad )
ROT %* ( Altura de la torre )
ABND
;
;
;

```

3.1.5.1. Ejemplos

3.1.5.1.1. Cálculo de la altura de una torre de enfriamiento

Se empleará una torre de enfriamiento para enfriar 1.427 kg/s.m² de agua desde 45.3°C hasta 30.4°C, alimentando en contracorriente 1.271 kg/s.m² de aire que se encuentra a una temperatura de 27.3°C y cuya temperatura de rocío es de 12°C. La presión de operación es 540 mm de Hg. Se ha estimado que el coeficiente global volumétrico de transferencia de masa: *K_{Ga}* es 3.978x10⁻⁵ kgmol/s.m³.mmHg. Calcule la altura que deberá tener la torre para conseguir el enfriamiento requerido.

Solución

Los datos con los que se cuenta son los siguientes:

```

G = 1.271 kg/s.m2
L = 2.427 kg/s.m2
TL2 = 45.3 °C 1_K CONVERT = 318.45_K
TL1 = 30.4 °C 1_K CONVERT = 303.55_K
TG1 = 27.3 °C 1_K CONVERT = 300.55_K
Tr1 = 12 °C 1_K CONVERT = 285.15_K
P = 540 mmHg
Kga = 1.413x10-5 kgmol/(s*m3*mmHg)

```

Con *T_{r1}* y el programa *pvapA* calculamos la presión parcial:

P_{A1} = 285 *pvapA* = 10.2424551222 mmHg

Entonces con la presión parcial y la presión total se calcula la humedad inicial del aire:

H₁ = 10.2424551222 540 Hpa = 0.012026332599

Ahora con *TG1* y *H1* calculamos la entalpía inicial del aire:

H_{y1} = 300.55 0.012026332599 *Hy* = 58.239168808 kJ/kg

Se tiene entonces con todos los datos para resolver el problema empleando "zenf2". El programa que resuelve el problema es el siguiente:

```

« 1.271 58.239168808 2.427 318.45 303.55 1.413E-5 540
zenf2
l_m →UNIT "z" →TAG 3 ROLLD
l_m →UNIT "HOG" →TAG 3 ROLLD
"NOG" →TAG 3 ROLLD
»

```

Haciendo correr el programa obtenemos:

NOG: 1.41951552706

HOG: '5.74991114781_m'

z: '8.16208815353_m'

Que son el número de unidades de transferencia, la altura de una unidad de transferencia y la altura que deberá tener la torre para cumplir con el enfriamiento requerido.

3.1.5.1.2. Flujo de agua que puede ser enfriado en un equipo existente

Una industria cuenta con una torre de enfriamiento de 15 m de altura que desea emplear para enfriar agua desde 45.3°C hasta 30.4°C. El flujo de aire que puede ser alimentado a la torre es 1.783 kg/s.m². El aire tiene una temperatura de 20°C, una humedad relativa del 50% y la presión de operación es de 0.79 bar. En base a mediciones efectuadas en el equipo se ha encontrado que el coeficiente global volumétrico de transferencia de masa K_G es igual a 2.032×10^{-5} kgmol/s.m³.mmHg. Calcule el flujo de agua que puede ser enfriado en este equipo.

Solución

Los datos con los que se cuenta son los siguientes:

$$z = 15 \text{ m}$$

$$G = 1.783 \text{ kg/s.m}^2$$

$$T_{L2} = 45.3 \text{ } ^\circ\text{C} \text{ 1_K CONVERT} = 318.45 \text{ _K}$$

$$T_{L1} = 30.4 \text{ } ^\circ\text{C} \text{ 1_K CONVERT} = 303.55 \text{ _K}$$

$$T_{G1} = 20.0 \text{ } ^\circ\text{C} \text{ 1_K CONVERT} = 293.15 \text{ _K}$$

$$H_{r1} = 50\%$$

$$P = 0.79 \text{ _bar 1_mmHg CONVERT} = 592.548729336 \text{ _mmHg}$$

$$K_G = 2.032 \times 10^{-5} \text{ _kgmol/(s.m}^3 \text{ .mmHg)}$$

Con la temperatura del aire (T_{G1}) calculamos la presión de vapor

$$P_{v1} = 293.15 \text{ pvapA} = 17.3514645824 \text{ mmHg}$$

Con la humedad relativa calculamos la presión parcial:

$$P_{A1} = 17.3514645824 \cdot 0.5 = 8.6757322912 \text{ mmHg}$$

Con la presión parcial calculamos la humedad inicial del aire

$$H_1 = 8.6757322912 / 592.548729336 \text{ Hpa} = 9.24259755908 \text{E-3}$$

Y con la humedad inicial calculamos la entalpía inicial del aire:

$$H_{y1} = 293.15 \cdot 9.24259755908 \text{E-3} \text{ Hy} = 43.5669552025 \text{ kJ/kg}$$

Como el programa "zenf2" permite calcular la altura de la torre y no el flujo de líquido, el cálculo del flujo es iterativo, siendo el proceso el siguiente: a) se asume un flujo de líquido; b) con el flujo asumido y el programa zenf2 se calcula la altura de la torre; c) se compara la altura calculada con la asumida: si son iguales el proceso concluye, siendo la solución el flujo asumido, caso contrario se repite el proceso desde el inciso (a).

Puesto que todo el proceso depende de una sola variable: el flujo superficial de líquido, siendo la función $f(L) = Z_{calculado} - Z_{conocido} = 0$, la solución puede ser automatizado empleando un método iterativo como el de la Secante o el de Newton Raphson. El programa que resuelve el problema es el siguiente:

«

« → L

```

«
1.783 43.5669552025 L
318.45 303.55 2.032E-5
592.54872936 zenf2 3 ROLLDD
DROP DROP 15 -
»
»
1.5 Newt 1_kg/(s*m^2) →UNIT
"L" →TAG
»

```

Haciendo correr el programa se obtiene

```
L: '8.69424293221_kg/s*m^2'
```

Que es el flujo de agua que puede ser enfriado en el equipo.

3.1.5.1.3. Simulación de condiciones de operación en una torre de enfriamiento existente

Una industria cuenta con una torre de 8 metros de altura que enfría 2.567 kg/s.m² de agua desde 40°C hasta 30°C. En contracorriente ingresan 1.893 kg/s.m² de aire a 25°C, 0.77 atm de presión y un porcentaje de humedad del 60%. La industria quiere saber en cuanto puede incrementar el flujo de agua si a) incrementa el flujo de aire al doble de su valor actual; b) disminuye el porcentaje de humedad del aire al 20%; c) Si disminuye la temperatura del aire a 20°C; d) Si incrementa el flujo de aire en un 50%, disminuye el porcentaje de humedad a 40% y disminuye la temperatura a 22°C.

Solución

Los datos con los que se cuenta son:

```

z = 8 m
G = 1.893 kg/s.m^2
L = 2.567 kg/s.m^2
TL2 = 40.0_°C 1_K CONVERT = 313.15_K
TL1 = 30.0_°C 1_K CONVERT = 303.15_K
TG1 = 25.0_°C 1_K CONVERT = 298.15_K
HP1 = 60%
P = 0.77_atm 1_mmHg CONVERT = 585.2_mmHg

```

Con la temperatura (T_{G1}) y la presión (P) podemos calcular la humedad de saturación:

$$H_{S1} = 298.15 \ 585.2 \ H_{sat} = 2.61106044398E-2$$

Entonces con la humedad de saturación y el porcentaje de humedad se puede calcular la humedad inicial del aire:

$$H_1 = 2.61106044398E-2 \ 0.6 \ * \ = 1.56663626639E-2$$

Finalmente con la temperatura (T_{G1}) y humedad inicial del aire (H_1) podemos calcular la entalpía inicial del aire:

$$H_{Y1} = 298.15 \ 1.56663626639E-2 \ H_y = 65.0491586127 \ \text{kJ/kg}$$

En este problema conocemos la altura de la torre pero no conocemos el coeficiente K_{Ga} . Dicho coeficiente debe ser calculado con las condiciones a las que opera la torre y posteriormente empleado para simular las otras condiciones de operación.

Para calcular el coeficiente K_{Ga} hacemos correr el programa *zenf2* empleando un valor de K_{Ga} cualquiera pues el objetivo es calcular el número de unidades de transferencia N_{OG} y en dicho cálculo no se emplea K_{Ga} , así el

siguiente programa devuelve el valor de N_{OG} empleando un valor de K_{Ga} igual a $2E-5$, pero se puede emplear cualquier otro valor sin que el valor de N_{OG} sea afectado:

```
« 1.893 65.0491586127 2.567 313.15 303.15
  2E-5 585.2
  zenf2
  DROP DROP "NOG" →TAG
»
```

Haciendo correr el programa se obtiene:

N_{OG} : .859613237397

Entonces calculamos H_{OG} con la ecuación (11):

$H_{OG} = 8 \cdot 0.859613237397 / = 9.30651094232 \text{ m}$

Y de la ecuación obtenemos K_{Ga} :

$K_{Ga} = 1.893 \cdot 28.97 \cdot 9.30651094232 \cdot 585.2 \cdot /$
 $1.19980566727E-5 \text{ kgmol/s.m}^3\text{.mmHg.}$

Ahora contamos con todos los datos para simular las otras condiciones de operación.

Para este inciso (a) el flujo de aire es:

$G = 1.893 \cdot 2 \cdot = 3.786 \text{ kg/s.m}^2$

Al igual que el ejemplo 3.1.5.1.2 la resolución del problema es iterativa. El programa que resuelve el problema es el siguiente:

```
«
« → L
  «
  3.786 65.0491586127 L 313.15 303.15
  1.19980566727E-5 585.2
  zenf2
  3 ROLLD DROP DROP 8 -
  »
»
3 Newt 1_kg/(s*m^2) →UNIT
"L" →TAG
»
```

Haciendo correr el programa se obtiene:

L: '2.96080429405_kg/(s*m^2)'

Entonces para un incremento de 100% en el flujo de aire el flujo de agua incrementa sólo en:

$2.96080429405 \cdot 2.567 - 2.567 / 100 \cdot = 15.3410321017\%$

Para el inciso (b) el porcentaje de humedad del aire disminuye a 20%, por lo tanto cambia la entalpía inicial del aire:

$H_{Y1} = 298.15 \cdot 2.61106044398E-2 \cdot 0.2 \cdot H_y = 38.4330528709 \text{ kJ/kg}$

El programa que resuelve el inciso (b) es el siguiente:

```
«
« → L
  «
  1.893 38.4330528709 L 313.15 303.15
  1.19980566727E-5 585.2
```

```

zenf2
3 ROLLD DROP DROP 8 -
»
»
3 Newt 1_kg/(s*m^2) →UNIT
"L" →TAG
»

```

Haciendo correr el programa se obtiene:

```
L: '3.31345573189_kg/(s*m^2)'
```

Entonces para una disminución del 40% en la humedad del aire el flujo de agua incrementa en:

$$3.31345573189 \cdot 2.567 - 2.567 / 100 * = 29.0789143705\%$$

Por lo que se puede concluir que es más eficiente disminuir la humedad del aire en lugar de incrementar su flujo.

Para el inciso (c) disminuye la temperatura del aire, por lo tanto cambia también la humedad inicial del aire:

$$H_{y1} = 293.15 \quad 293.15 \quad 585.2 \quad H_{sat} \quad 0.6 * H_y = 49.0550201036 \text{ kJ/kg}$$

El programa que resuelve el inciso (c) es el siguiente:

```

«
« → L
«
1.893 49.0550201036 L 313.15 303.15
1.19980566727E-5 585.2
zenf2
3 ROLLD DROP DROP 8 -
»
»
3 Newt 1_kg/(s*m^2) →UNIT
"L" →TAG
»

```

Haciendo correr el programa se obtiene:

```
L: '3.01872495897_kg/(s*m^2)'
```

Por lo tanto el flujo de agua incrementa en:

$$3.01872495897 \cdot 2.567 - 2.567 / 100 * = 17.597388351\%$$

Que es un incremento importante para una disminución relativamente pequeña de temperatura, no obstante en la práctica esa disminución de temperatura puede resultar muy costosa.

Finalmente, para el inciso (d) el flujo de aire es:

$$G = 1.893 \cdot 1.5 * = 2.8395 \text{ kg/s.m}^2.$$

La temperatura del aire disminuye a 22°C y el porcentaje de humedad a 40%, por lo que la entalpía inicial del aire cambia:

$$H_{y1} = 295.15 \quad 295.15 \quad 585.2 \quad H_{sat} \quad 0.4 * H_y = 44.0846424352$$

El programa que resuelve el inciso (d) es el siguiente:

```

«
« → L
«
2.8395 44.0846424352 L 313.15 303.15
1.19980566727E-5 585.2

```

```

zenf2
3 ROLLD DROP DROP 8 -
»
»
3 Newt 1_kg/(s*m^2) →UNIT
"L" →TAG
»

```

Haciendo correr el programa se obtiene:

```
L: '3.48479264068_kg/(s*m^2)'
```

Por lo tanto el flujo de agua incrementa en:

```
3.48479264068 2.567 - 2.567 / 100 * = 35.7535115185%
```

Que como era de esperar es un incremento importante.

De todas las alternativas estudiadas probablemente la más viable sea la de disminuir la humedad del aire, pues ello se puede lograr empleando desecantes como zeolitas, silicagel, alúmina activa, cloruro de calcio, sulfato de cobre, etc. En última instancia las consideraciones de tipo económica serán las que determinen el camino a seguir.

Se debe hacer notar que el disponer de programas para el diseño nos permite analizar estas y otras posibilidades en un tiempo reducido y casi sin costo, pues la empresa continúa con su operación normal. Puesto que el coeficiente global volumétrico es calculado con los datos medidos en el equipo, los resultados obtenidos con los programas son bastante confiables.

3.1.5.2. Ejercicios

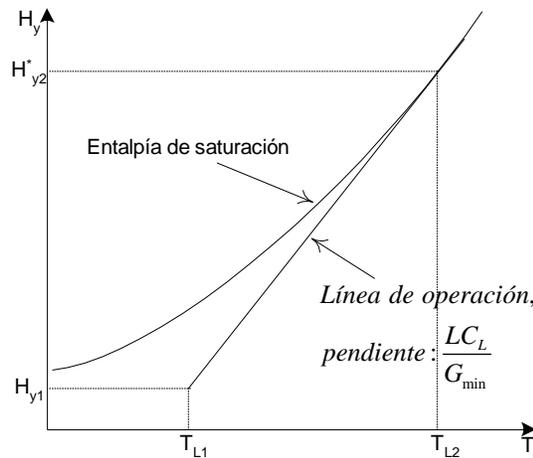
7. Empleando los datos y el resultado obtenido en el ejercicio 2, haga lo siguiente: a) Calcule el coeficiente global volumétrico de transferencia de masa; b) Empleando el resultado del inciso (a) calcule el flujo de agua que podría enfriarse si el flujo de aire es de 2100 lb/h.pie².
8. Considerando que para la torre del anterior ejercicio el coeficiente volumétrico global de transferencia de masa es 6.90 lbmol/h.ft³.atm. a) Calcule la altura que deberá tener la torre para lograr el enfriamiento requerido; b) Calcule la temperatura a la que es enfriada el agua si el aire ingresa a 80°F y tiene una temperatura de bulbo húmedo de 67°F.
9. Una industria cuenta con una torre de enfriamiento de 10 m de altura en la cual se enfría 3.2 kg/s.m² de agua desde 50 hasta 33°C. En contracorriente ingresa 1.7 kg/s.m² de aire a 20°C, con una temperatura de bulbo húmedo de 12°C y una presión de 540 mm de Hg. La industria desea emplear la misma torre para enfriar 4.0 kg/s.m² de agua desde 40 hasta 30°C. Calcule el flujo de aire que deberá alimentarse a la torre para lograr el enfriamiento requerido.
10. En una industria se planifica emplear dos torres de enfriamiento para enfriar 2.345 kg/s.m² de agua desde 60 hasta 25°C. En la primera torre se enfriará el agua desde 60 hasta 35°C y en la segunda desde 35 hasta 25°C. La altura de una unidad de transferencia medida en un equipo experimental operando en condiciones similares a las del equipo industrial es $H_{OG} = 3.674$ m. A cada una de las torres se alimentará 1.5 kg/s.m² de aire atmosférico que se encuentra a 20°C, con un porcentaje de humedad del 70% y una presión de 0.67 atm. Calcule la altura que deberá tener cada una de las torres para lograr el enfriamiento requerido.

11. Una industria cuenta con una torre de enfriamiento de 14.7 m de altura que enfría 4234 lb/h.ft² de agua desde 57 hasta 40.2 °C, empleando 3820 lb/h.ft² de aire atmosférico que se encuentra a una temperatura de 74°F, tiene una humedad relativa del 47% y una presión de 12 psi. a) Empleando estos datos calcule el coeficiente global volumétrico de transferencia de masa de la torre; b) Calcule la temperatura a la que se enfriaría el agua si el aire ingresa con una humedad relativa del 20%; c) Calcule la temperatura a la que se enfriaría el agua si el aire ingresa a una temperatura de 70°F; d) Calcule la temperatura a la que se enfriaría el agua si el flujo de aire incrementa a 4520 lb/h.ft².
12. Tomando los datos y resultados del ejercicio anterior, a) Calcule el flujo de agua que podría ser enfriado si la humedad relativa del aire es del 35%; b) calcule el flujo de aire que debería emplearse si la temperatura del aire es de 60°F y tiene una humedad relativa del 40%.

3.2. Flujo mínimo de aire

En ocasiones no es un dato el flujo de aire seco que ingresa a la torre. En tales casos se suele fijar el flujo de aire seco (G) entre 1.3 y 1.5 veces el flujo mínimo de aire.

El flujo de aire es mínimo cuando la línea de operación es tangente a la curva de equilibrio a la temperatura T_{L2} , es decir cuando la línea de operación casi toca el punto (T_{L2}, H^*_{y2}) , tal como se muestra en la siguiente figura:



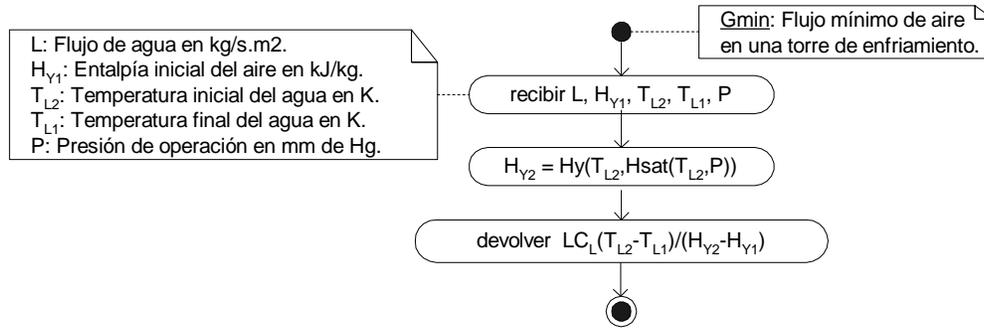
Puesto que se tienen dos puntos conocidos: (T_{L1}, H_{y1}) y (T_{L2}, H^*_{y2}) , la pendiente de la curva puede ser calculada con la ecuación:

$$\frac{LC_L}{G_{\min}} = \frac{H^*_{y2} - H_{y1}}{T_{L2} - T_{L1}}$$

Donde H^*_{y2} es la entalpía del aire saturado a la temperatura T_{L2} . Despejando G_{\min} , se tiene la ecuación que nos permite calcular el flujo mínimo de aire:

$$G_{\min} = \frac{LC_L(T_{L2} - T_{L1})}{H^*_{y2} - H_{y1}} \quad (13)$$

Aunque la lógica para resolver esta ecuación es muy sencilla presentamos a continuación su diagrama de actividades.



El programa elaborado en base al algoritmo es el siguiente:

```

xNAME Gmin ( Flujo mínimo de aire en una torre de enfriamiento )
::
( Datos: )
CK5&Dispatch ( L=6= Flujo de agua )
# 11111 ( HY1=5= Entalpía inicial del aire )
:: ( TL2=4= Temperatura inicial del agua )
%0 ( TL1=3= Temperatura final del agua )
' NULLLAM SIX NDUPN DOBIND ( P=2= Presión de operación )
4GETLAM 4GETLAM 2GETLAM
xHsat xHy 1PUTLAM ( HY2=1= Hy[TL2,Hsat[TL2,P]] )
6GETLAM % 4.187 %* 4GETLAM
3GETLAM %- %* 1GETLAM
5GETLAM %- %/ ( Flujo mínimo de aire )
ABND
;
;
  
```

El cálculo del flujo mínimo de aire es útil no sólo cuando no se cuenta con el valor de G , sino también para verificar que dicho dato sea coherente, es decir que sea mayor a G_{min} , pues de lo contrario la altura de la torre sería infinita.

3.2.1. Ejemplos

3.2.1.1. Cálculo del flujo mínimo

En una torre se quiere enfriar 1.356 kg/s.m² de agua, desde 45.5 hasta 29.7°C empleando aire a 27°C, con una temperatura de bulbo húmedo de 20°C y una presión de 570 mm de Hg. Calcule al flujo mínimo de aire que debe ser alimentado a la torre para lograr el enfriamiento requerido.

Solución

Los datos con los que contamos son:

```

L = 1.356 kg/s.m2
TL2 = 45.5 °C 1_K CONVERT = 318.65_K
TL1 = 29.7 °C 1_K CONVERT = 292.85_K
TG1 = 27.0 °C 1_K CONVERT = 300.15_K
TW1 = 20.0 °C 1_K CONVERT = 293.15_K
P = 570 mmHg
  
```

La entalpía inicial del aire se calcula con T_{G1} , T_{W1} , P y los programas H_{tw} y H_y :

```

HY1 = 300.15 300.15 293.15 570 Htw Hy = 69.4233961631 kJ/kg
  
```

Entonces contamos con todos los datos para calcular el flujo mínimo con el programa G_{min} :

$$G_{min} = 1.356 \cdot 69.4233961631 \cdot 318.65 \cdot 292.85 \cdot 570 \quad G_{min} = .681061302198 \text{ kg/s.m}^2$$

Entonces en la torre deberá emplearse un flujo superior a 0.68 kg/s.m².

3.2.1.2. Altura de una torre empleando 1.5 veces el flujo mínimo de aire

En una industria se enfriará 6200 lb/h.ft² de agua desde 140 hasta 86°F empleando aire a 68°C con una temperatura de bulbo húmedo de 48°F y una presión de 11.4 psi. El coeficiente global volumétrico de transferencia de masa para la torre es 7.2 lbmol/h.ft³.atm. Calcule la altura que deberá tener la torre para lograr el enfriamiento si se trabaja con un flujo de aire igual a 2 veces el flujo mínimo.

Solución

Los datos con los que contamos son:

```
L = 6200_lb/(h*ft^2) 1_kg/(s*m^2) CONVERT = 8.40862537377_kg/(s*m^2)
TL2 = 140_F 1_K CONVERT = 333.15_K
TL1 = 86_F 1_K CONVERT = 303.15_K
TG1 = 68_F 1_K CONVERT = 293.15_K
TW1 = 48_F 1_K CONVERT = 282.038888889_K
Kga = 72.2_lbmol/(h*ft^3*atm) 1_kgmol/(s*m^3*mmHg) CONVERT
      = 4.22709450146E-4_kgmol/(s*m^3*mmHg)
P = 11.4_psi 1_mmHg CONVERT = 589.550231315_mmHg
```

La entalpía inicial del aire se calcula con TG1, TW1, P y los programas Htw y Hy:

$$H_{Y1} = 293.15 \quad 293.15 \quad 282.038888889 \quad 589.550231315 \quad H_{tw} \quad H_y = 31.3439945835 \text{ kJ/kg}$$

Con estos datos podemos calcular el flujo mínimo de aire:

$$G_{min} = 8.40862537377 \cdot 31.3439945835 \cdot 333.15 \cdot 303.15 \cdot 589.550231315 \quad G_{min} = 1.81776649259 \text{ kg/s.m}^2$$

Por lo tanto el flujo de aire a emplear en la torre es:

$$G = 1.81776649259 \cdot 2 = 3.63553298518 \text{ kg/s.m}^2$$

Contamos ahora con todos los datos para calcular la altura de la torre con el programa zenf2. El programa que resuelve el problema es el siguiente:

```
<< 3.63553298518 31.3439945835 8.40862537377
    333.15 303.15 4.22709450146E-4 589.550231315
    zenf2
    1_m →UNIT "z" →TAG 3 ROLLD
    1_m →UNIT "HOG" →TAG 3 ROLLD
    "NOG" →TAG 3 ROLLD
>>
```

Haciendo correr el programa se obtiene:

```
NOG: 2.78719239608
HOG: '.503566490973_m'
z: '1.40353669456_m'
```

Por lo tanto la torre deberá tener un altura de 1.40 metros para lograr el enfriamiento requerido.

3.2.2. Ejercicios

- 13. Una torre de enfriamiento en contracorriente será empleada para enfriar 1.783 kg/s.m² de agua desde 40.2 hasta 26.7°C. El aire ingresa por el

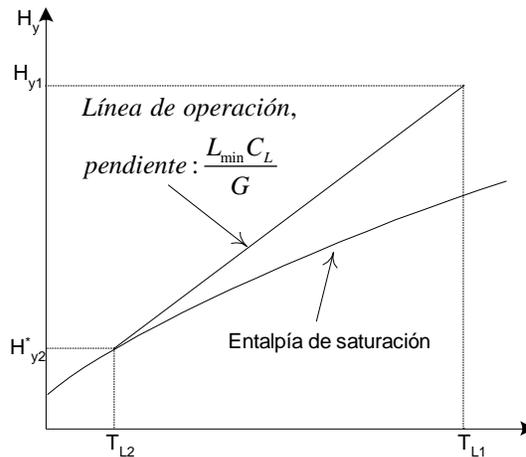
fondo de la columna a 23.9°C con una temperatura de bulbo húmedo de 21.1°C . La altura de una unidad de transferencia para la torre es $H_{OG}=0.533\text{ m}$. Calcule la altura que deberá tener la torre si se emplea un flujo de aire igual a 1.5 veces el flujo mínimo y la torre opera a una presión de 1 atm.

14. En una industria se planifica emplear una torre de enfriamiento con aire en contracorriente para enfriar 2500 lb/h.ft^2 de agua desde 90 hasta 75°F , empleando aire que se encuentra a una temperatura de 70°F , tiene un porcentaje de humedad el 50% y se encuentra a una presión de 12 lbf/plg^2 . El coeficiente global volumétrico de transferencia de masa es $6.45\text{ lbmol/h.ft}^3.\text{atm}$. Calcule la altura que deberá tener la torre para lograr el enfriamiento requerido si se emplea un flujo de aire igual a 1.3 veces el flujo mínimo.

3.3. Flujo mínimo de agua

Cuando la torre se emplea para secar aire en lugar de enfriar agua, suele suceder que se conoce el flujo de gas a secar (deshumidificar) pero no el del agua fría. En esos casos se fija el flujo de agua entre 1.3 y 1.5 veces el flujo mínimo de agua. Puesto que el proceso se invierte (el agua fría gana calor y el aire lo pierde), la línea de operación se encuentra encima de la curva de equilibrio (la curva de la entalpía de saturación).

El flujo mínimo de agua ocurre cuando el aire es saturado y enfriado hasta la temperatura de entrada del agua, situación para la cual se requiere una torre de altura infinita. Por lo tanto el flujo mínimo de líquido se encuentra cuando la línea de operación es tangente a la curva de equilibrio a la temperatura T_{L2} , como se muestra en la siguiente figura:

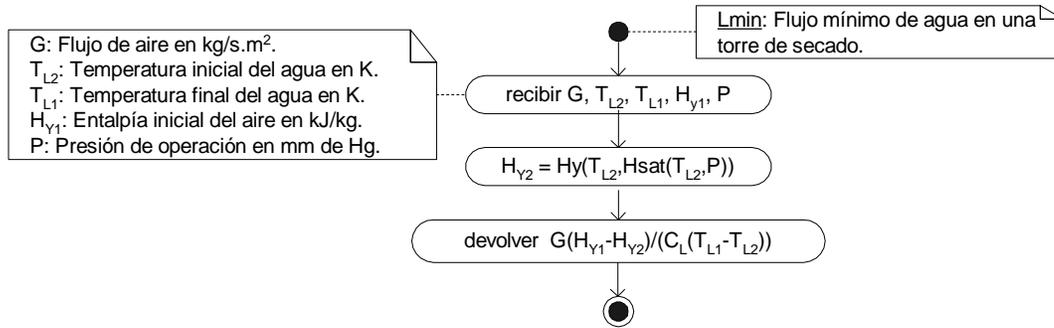


Como se conocen dos puntos (T_{L2}, H_{y2}^*) y (T_{L1}, H_{y1}) , es posible calcular la pendiente de la recta y despejar de ella el flujo mínimo de agua:

$$L_{\min} = \frac{G(H_{y1} - H_{y2}^*)}{C_L(T_{L1} - T_{L2})} \quad (14)$$

Donde H_{y2}^* es la entalpía del aire saturado a la temperatura T_{L2} .

El algoritmo que resuelve la ecuación (14) se presenta en el diagrama de actividades de la siguiente página.



Y el código elaborado en base al mismo es el siguiente:

```
xNAME Lmin ( Flujo mínimo de agua en una torre de humidificación o secado )
::
    CK5&Dispatch
    # 11111
    ::
        %0
        ' NULLLAM SIX NDUPN DOBIND
        4GETLAM 4GETLAM 2GETLAM
        xHsat xHy 1PUTLAM
        6GETLAM 5GETLAM 1GETLAM %-
        %* % 4.187 3GETLAM 4GETLAM
        %- %* %/
        ABND
    ;
;
```

Al igual que sucede con el flujo mínimo del aire, el cálculo del flujo mínimo del agua es útil para verificar que los datos sean coherentes y que por lo tanto el flujo de agua no sea menor o igual al mínimo, pues de ser así la altura de la torre sería infinita.

3.3.1. Ejemplo

3.3.1.1. Flujo mínimo de agua en una torre de secado

En una industria se quiere secar 1.234 kg/s.m² de aire desde 80 hasta 30% de humedad. El aire se encuentra a 40°C y una presión de 640 mm de Hg. Para el secado se empleará agua en contracorriente que ingresará a 5°C y saldrá a 10°C. Calcule el flujo mínimo de agua que puede emplearse.

Solución

Los datos con los que se cuenta son los siguientes:

```
G = 1.234 kg/s.m2
T_L2 = 5 °C 1_K CONVERT = 278.15_K
T_L1 = 10 °C 1_K CONVERT = 283.15_K
T_G1 = 40 °C 1_K CONVERT = 313.15_K
P = 640_mmHg
```

La humedad inicial del aire es:

```
H_y1 = 313.15 313.15 640 Hsat 0.8 * Hy 4 RND = 161.215 kJ/kg
```

Se tiene entonces todos los datos para calcular el flujo mínimo de agua con el programa *Lmin*:

```
L_min = 1.234 278.15 283.15 161.215 640 Lmin 3 RND = .944 kg/s.m2
```

4. TORRES DE ABSORCIÓN POR ETAPAS

En esta operación dos fases: gas y líquido, se ponen en contacto y uno o varios componentes de la fase gaseosa (soluto) son *absorbidos* por la fase líquida. Se recurre a la absorción cuando: a) es necesario recuperar uno o más componentes de una mezcla gaseosa y b) es necesario eliminar uno o más componentes de una mezcla gaseosa.

El primer caso se presenta en la fabricación del amoníaco, donde del reactor sale una mezcla gaseosa compuesta por amoníaco y aire. El amoníaco se recupera de esta mezcla poniéndola en contacto con agua pura en una torre de absorción, como resultado de esta operación el amoníaco es absorbido por el agua y el aire queda libre de amoníaco. Posteriormente la mezcla amoníaco - agua es sometida a un proceso de destilación para obtener amoníaco relativamente puro.

El segundo caso se presenta en la fabricación de ácido sulfúrico, donde como producto intermedio se obtiene una mezcla gaseosa de SO_2 , CO_2 y aire. El SO_2 y el aire deben pasar luego a los reactores catalíticos donde el SO_2 se convierte en SO_3 , sin embargo se debe evitar el paso del CO_2 pues este compuesto ensucia los catalizadores. Con este fin se ponen en contacto la mezcla gaseosa con una mezcla alcalina de agua e hidróxido de calcio (en una torre de absorción). Como resultado de esta operación, el CO_2 es absorbido en la fase líquida y la fase gaseosa queda libre de este compuesto.

Otro ejemplo de absorción que está más a nuestro alcance se presenta en la fabricación de bebidas gaseosas, donde el CO_2 es absorbido en una mezcla líquida (jarabe) a presiones moderadas y temperaturas bajas.

El proceso contrario a la absorción, es decir el paso de un componente de la fase líquida a la fase gaseosa se conoce con el nombre de *desorción*.

4.1. Datos de equilibrio

Para realizar el diseño y/o dimensionamiento de una torre de absorción debemos contar con los datos de equilibrio gas - líquido para el sistema que estemos tratando.

En esta materia el cálculo de los equipos de absorción es simplificado pues sin importar cuan complejas sean las fases gaseosa, líquida y el soluto, siempre se asume que se trata de tres compuestos: uno para la fase líquida, otro para la fase gaseosa y otro para el soluto.

Para los casos más comunes los datos de equilibrio están disponibles en bibliografía. Sin embargo, para llevar a cabo un diseño confiable, estos datos deben ser obtenidos experimentalmente, en condiciones similares a las del equipo industrial. Para ello se colocan cantidades conocidas de las dos fases en un recipiente cerrado a la temperatura a la que se operará en el equipo industrial, entonces se agita repetidamente la mezcla hasta que se alcanza el equilibrio (es decir hasta que no varía la presión).

Una vez alcanzado el equilibrio se efectúa un análisis químico de las dos fases, con lo que se tiene un dato de equilibrio. El experimento se repite empleando diferentes proporciones de gas y líquido llegando a conformar así una tabla o gráfica de equilibrio para una temperatura dada.

4.1.1. La ley de Henry

Cuando las concentraciones son bajas y no se cuenta con datos experimentales de equilibrio, se puede emplear como primera aproximación la ley de

Henry, esta ley señala que la presión parcial de un compuesto "A" en la fase gaseosa es directamente proporcional a su fracción molar en la fase líquida, siendo la constante de proporcionalidad la constante de Henry, es decir:

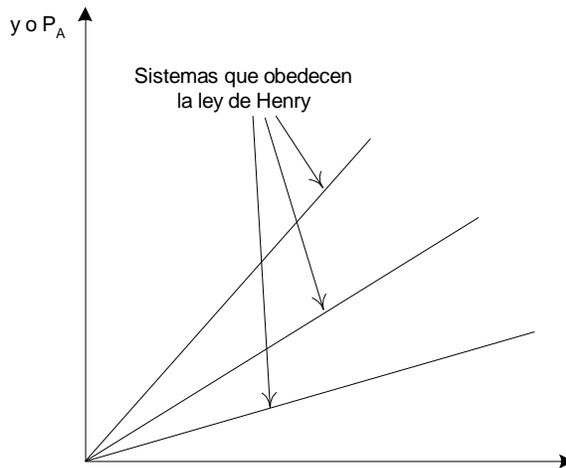
$$P_A = Hx_A \tag{1}$$

Donde "H" (la constante de Henry) está en unidades de presión, normalmente atmósferas. Si ambos lados de la ecuación (1) se dividen entre la presión total (P), se tiene:

$$\frac{P_A}{P} = \frac{H}{P} x_A$$

$$y_A = H' x_A \tag{2}$$

Donde H' es la constante adimensional de Henry. Gráficamente las curvas de equilibrio de los sistemas que siguen la ley de Henry son simplemente líneas rectas que pasan por el origen, tal como se muestra en la siguiente figura:



4.1.1.1. Ejemplo

Calcule los volúmenes de CO₂ por volumen de agua a 293.15 K, en el sistema CO₂ - agua, que está en equilibrio a 2 atm de presión.

Solución:

La constante de Henry para el sistema CO₂ - H₂O a 293.15 K (obtenida del apéndice A de Geankoplis) es 0.142x10⁴ atm, entonces la fracción molar de CO₂ en el agua es:

$$x_A = P_A/H = 2 \text{ } 0.142E4 / = 1.40845070423E-3$$

Por lo tanto existen 0.00141 moles de CO₂ por cada mol de solución. Tomando como base 1 gmol de solución, el volumen ocupado por los moles de CO₂ disueltos en condiciones normales de presión y temperatura es:

$$V_{CO_2} = 1.40845070423E-3 \text{ R CONST } 1_1 \text{ *atm/(K*gmol) CONVERT UVAL * StdT CONST UVAL * StdP CONST UVAL / = 3.15691510931E-2 \text{ litros}$$

Debido a la pequeña cantidad de CO₂ se puede considerar que el volumen del líquido es básicamente el volumen de agua. El volumen del agua es prácticamente 1 litro por kg, entonces el volumen de 1 gmol de agua (18.02 g) es:

$$V_{H_2O} = 18.02 \text{ } 1000 / = 0.01802 \text{ litros}$$

Por lo tanto los volúmenes de CO₂ por volumen de agua son:

$$3.15691510931E-2 \cdot 0.01802 / = 1.7518951772$$

Es decir existen 1.75 volúmenes de CO₂ por volumen de agua.

4.1.1.2. Ejercicios

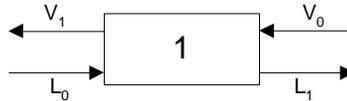
1. Una fábrica de gaseosas requiere que su producto tenga 5 volúmenes de CO₂ por volumen de agua a 4 °C de temperatura. Calcule la presión a la que deberá trabajar el equipo de absorción para lograr este objetivo (la constante de Henry a 4°C es 0.08528x10⁴ atm).
2. Repita el ejercicio anterior si la temperatura es de 20°C.

4.2. Absorción en una etapa de equilibrio

En ocasiones el proceso de absorción se lleva a cabo en un equipo que tiene cierta eficiencia con relación a una etapa de equilibrio. Para dimensionar o diseñar tales equipos se realizan los cálculos en una etapa de equilibrio y se emplea la eficiencia del equipo para convertir los resultados ideales en resultados reales.

En la operación de absorción se asume que nada del líquido inerte (el solvente de la fase líquida) pasa a la fase gaseosa y que nada del gas inerte (el solvente de la fase gaseosa) pasa a la fase líquida. Esto por supuesto no es del todo cierto, sin embargo, las proporciones de líquido inerte que pasan a la fase gaseosa y de gas inerte que pasan a la fase líquida son casi siempre muy pequeñas, razón por la cual esta simplificación es válida en la práctica.

Esquemáticamente una etapa de equilibrio se representa como se muestra en la siguiente figura:



Donde "L" es el flujo de líquido y "V" es el flujo de gas. En esta etapa las corrientes de salida L_1 y V_1 están en equilibrio.

Denominaremos "x" a la fracción de soluto (molar o másica) en la fase líquida y "y" a la fracción de soluto (molar o másica) en la fase gaseosa. Igualmente denominaremos "A" al soluto, "B" al líquido inerte (solvente de la fase líquida) y "C" al gas inerte (solvente de la fase gaseosa).

Realizando el balance general de materia en la etapa de equilibrio resulta:

$$L_0 + V_0 = L_1 + V_1 = M \quad (3)$$

Donde "M" es el flujo total de gas y líquido que ingresa y sale de la etapa. Del balance de materia para el soluto se tiene:

$$L_0 x_0 + V_0 y_0 = L_1 x_1 + V_1 y_1 = A \quad (4)$$

Donde "A" es el flujo total de soluto que ingresa y sale de la etapa.

En este sistema normalmente se conocen los flujos y composiciones de las dos corrientes de entrada y el problema radica en calcular los flujos y composiciones de las corrientes de salida. En consecuencia existen cuatro

incógnitas (L_1 , V_1 , x_1 , y_1) y sólo dos ecuaciones, por lo tanto se requieren dos ecuaciones más para resolver el sistema.

La primera de esas dos ecuaciones resulta de haber asumido que nada de gas inerte pasa a la fase líquida y que igualmente nada de líquido inerte pasa a la fase gaseosa, en consecuencia el flujo de líquido inerte (y por supuesto el de gas inerte) es constante:

$$B = L_0(1 - x_0) = L_1(1 - x_1) = \text{cte.} \quad (5)$$

La cuarta ecuación es la relación de equilibrio. Sin importar que los datos de equilibrio se encuentren en forma tabular, gráfica o como ecuación, nos permite calcular el valor de "y" para un valor conocido de "x":

$$y = f_x(x) \quad (6)$$

Donde f_x es la relación de equilibrio en la forma en que se encuentre. La función de equilibrio contraria, es decir aquella que nos permite calcular el valor de "x" para un valor conocido de "y" es:

$$x = f_y(y) \quad (7)$$

En la mayoría de los casos prácticos, los datos de equilibrio se encuentran en forma tabular o gráfica pues son obtenidos experimentalmente. Si los datos de equilibrio se encuentran en forma gráfica pueden ser llevados a la forma tabular leyendo los datos más significativos de la gráfica y formando así una tabla.

Debido a que con frecuencia los problemas se resuelven de manera iterativa, es útil contar con un programa que empleando los datos tabulados, nos devuelva el valor de la variable dependiente para un determinado valor de la variable independiente. Ha este tipo de programas se conoce con el nombre de *generadores de funciones arbitrarias* y básicamente son programas de interpolación, siendo el método de interpolación más empleado el de interpolación lineal.

Puesto que esta es una situación que se nos presentará con frecuencia en las diferentes operaciones que estudiemos, elaboraremos primero un programa para llevar a cabo la interpolación (o extrapolación) lineal en base a dos puntos conocidos: (x_1, y_1) y (x_2, y_2) .

En la interpolación lineal se emplea la ecuación de la línea recta:

$$y = a + bx \quad (8)$$

Donde "x" es la variable independiente y "y" la variable dependiente. Las constantes "b" y "a" se calculan con:

$$b = \frac{y_1 - y_2}{x_1 - x_2} \quad (9)$$

$$a = y_1 - bx_1 \quad (10)$$

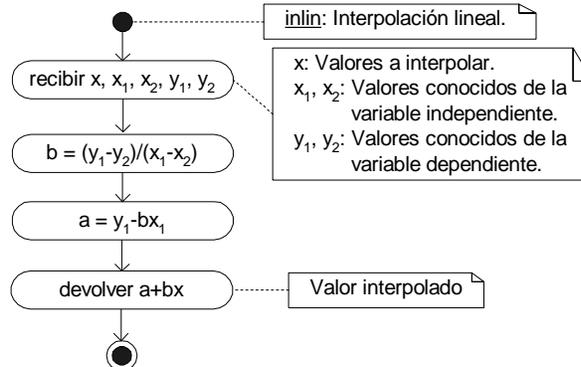
El algoritmo para llevar a cabo la interpolación lineal se presenta en el diagrama de actividades de la siguiente página. El programa elaborado en base a dicho algoritmo (añadido al archivo *Metodos1.s*) es el siguiente:

```
xNAME inlin      ( Interpolación lineal )
::              ( Datos en la pila: x: valor a interpolar; x1,x2: valores )
CK5&Dispatch ( conocidos de la variable dependiente; y1,y2: valores )
# 11111        ( conocidos de la variable dependiente )
::
OVER SWAP %- 4PICK 4ROLL %- %/ ( b )
      DUP 4ROLL %* ROT SWAP %-      ( a )
```

```

3UNROLL %* %+
;
;

```



Por ejemplo dados los siguientes datos:

x	Y
12	14.5
16	19.6

Podemos interpolar el valor de "y" para "x=15":

```
y = 15 12 16 14.5 19.6 inlin = 18.325
```

Igualmente podemos extrapolar el valor de "y" para "x=18":

```
y = 18 12 16 14.5 19.6 inlin = 22.15
```

Por supuesto también podemos interpolar el valor de "x" para un valor conocido de "y", así para "y=17" el valor de "x" es:

```
y = 17 14.5 19.6 12 16 inlin = 13.9607843137
```

Ahora podemos elaborar un programa que busque el segmento de interpolación y una vez ubicado realice la interpolación lineal con el programa *inlin*. El algoritmo de este programa se presenta en el diagrama de actividades de la siguiente página y el código del programa elaborado en base al mismo (y escrito en el archivo *metodos1.s*) es el siguiente:

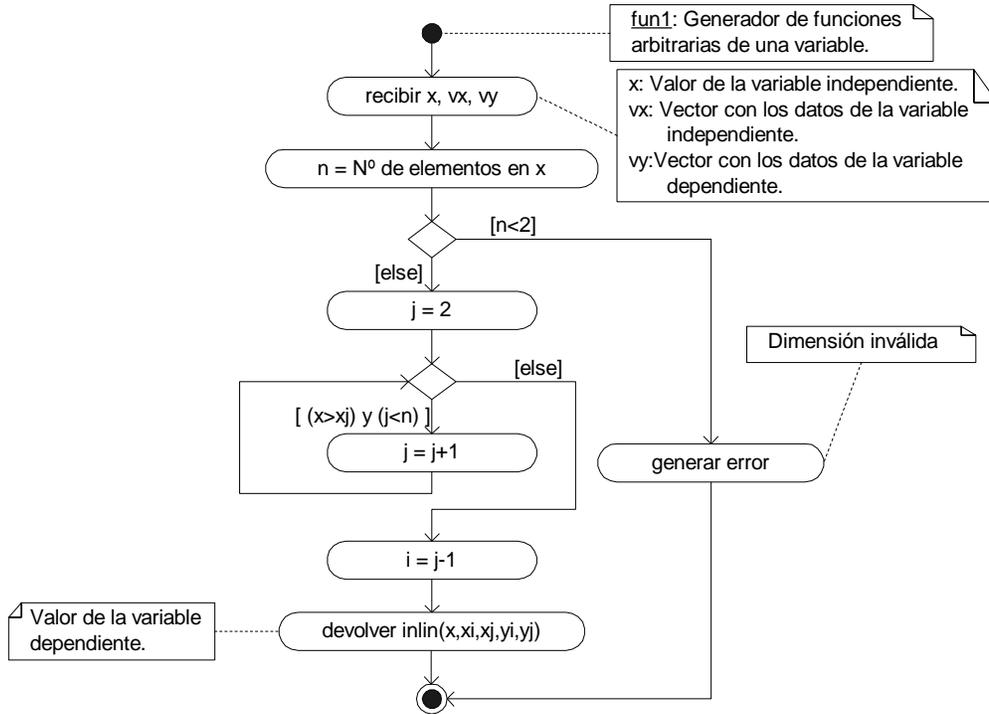
```

xNAME fun1 ( generador de funciones arbitrarias de una variable )
::          ( dato en la pila: x: valor de la variable independiente )
CK3&Dispatch ( vx: vector independiente; vy: vector dependiente)
# 144
::          ( 501 = dimensión inválida )
  SWAP DUP MDIMSDROP ( n )
  DUP #2 #<
  IT :: 2DROP # 501 DO#EXIT ; ( si n<2 => error )
  #2 ( j = 2; pila: x, vy, vx, n, j )
  BEGIN
    DUP 3PICK #< ( j < n )
    6PICK 5ROLL 4PICK PULLREALEL ( x[j] )
    SWAP 6UNROLL %> AND ( x > x[j] )
  WHILE
    #1+ ( j = j+1 )
  REPEAT
    SWAP DROP DUP #1- ( i = j-1; ; pila: x, vy, vx, j, i )
    ROT OVER PULLREALEL ( x[i] )
    SWAP 4PICK PULLREALEL SWAP DROP ( x[j] )
    5ROLL 4ROLL PULLREALEL ( y[i] )
    SWAP 5ROLL PULLREALEL SWAP DROP ( y[j] )

```

```

xinlin
;
;
    
```



Por ejemplo si los datos son los siguientes:

X	Y
1	1.1
5	7.2
7	9.8
9	14.5
12	16.7
16	19.6
20	25.5

La función arbitraria se crea simplemente guardando estos datos en variables, por ejemplo: vx y vy:

```
[ 1 5 7 9 12 16 20 ] 'vx' STO
```

```
[ 1.1 7.2 9.8 14.5 16.7 19.6 25.5 ] 'vy' STO
```

Y llamando al programa "fun1", así por ejemplo los valores de "y" para "x=8, 20, 21, 1.2 y 0.5" son:

$y_{(8)} = 8 \text{ vx vy fun1} = 12.15$

$y_{(20)} = 20 \text{ vx vy fun1} = 25.5$

$y_{(21)} = 21 \text{ vx vy fun1} = 26.975$ (extrapolación)

$y_{(1.2)} = 1.2 \text{ vx vy fun1} = 1.405$

$y_{(0.5)} = 0.5 \text{ vx vy fun1} = 0.3375$ (extrapolación)

Igualmente los valores de "x" para "y=0.5, 10, 17, 23 y 30" son:

$x_{(0.5)} = 0.5 \text{ vy vx fun1} = .606557377049$ (extrapolación)

$x_{(0.5)} = 10 \text{ vy vx fun1} = 7.08510638298$

$x_{(0.5)} = 17$ vy vx fun1 = 12.4137931034
 $x_{(0.5)} = 23$ vy vx fun1 = 18.3050847458
 $x_{(0.5)} = 30$ vy vx fun1 = 23.0508474576 (extrapolación)

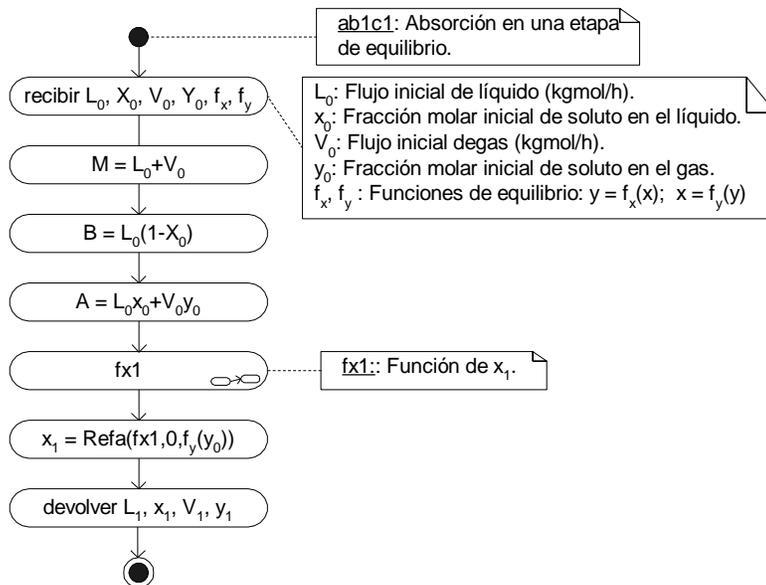
Otra alternativa consiste en ajustar los datos a una ecuación analítica y emplear la ecuación resultante como la función de equilibrio: f_x . El principal inconveniente de esta alternativa radica en encontrar la forma de la ecuación que ajusta correctamente los datos tabulados. Además se debe verificar que el comportamiento de la ecuación resultante sea coherente, más allá del intervalo de los datos, pues no es raro que la ecuación sea empleada para obtener valores fuera del intervalo original (extrapolación).

Con el programa "fun1" estamos ahora en condiciones de resolver el sistema de 4 ecuaciones con 4 incógnitas sin importar que los datos de equilibrio se encuentren en forma tabular, gráfica o analítica.

Entonces para el caso más frecuentemente, donde se conocen los flujos y composiciones de entrada, el sistema de 4 ecuaciones puede ser colocado en función de una sola variable: x_1 , la fracción de soluto en la fase líquida:

$$\begin{aligned}
 f(x_1) &= L_1x_1 + V_1y_1 - A = 0 \\
 y_1 &= f_x(x_1) \\
 L_1 &= \frac{B}{1-x_1} \\
 V_1 &= M - L_1 \\
 M &= L_0 + V_0 = \text{cte.} \\
 B &= L_0(1-x_0) = \text{cte.} \\
 A &= L_0x_0 + V_0y_0 = \text{cte.}
 \end{aligned}
 \tag{11}$$

Que puede ser resuelta con uno de los métodos iterativos como el de Regula - Falsi, empleando como valores límite 0 y la máxima concentración posible de soluto en la fase líquida, la cual corresponde al valor de "x" en equilibrio con el valor inicial de soluto en la fase gaseosa ($f_y(y_2)$). El algoritmo para resolver esta ecuación es el siguiente:



Donde el submódulo "fx1" tiene el siguiente diagrama de actividades:

4.2.1. Ejemplos

4.2.1.1. Absorción de CO_2 en una etapa de equilibrio

Se empleará un equipo de absorción equivalente a una etapa de equilibrio para absorber CO_2 de una mezcla de CO_2 -Aire empleando agua pura a 273.2 K. El gas que ingresa al equipo tiene una presión de 3 atm, una fracción molar de soluto igual a 0.45 y un flujo molar igual a 200 kgmol/h. El flujo molar del agua es de 350 kgmol/h. Asumiendo que el sistema obedece la ley de Henry calcule los flujos y composiciones de las corrientes de salida.

Solución

Los datos con los que se cuenta son los siguientes:

$L_0 = 350$ kgmol/h
 $x_0 = 0$
 $V_0 = 200$ kgmol/h
 $y_0 = 0.45$
 $P = 3$ atm
 $H = 0.0728 \times 10^4$ atm (del apéndice A3 de Geankoplis)

Entonces

$$H' = H/P = 0.0728 \times 10^4 / 3 = 242.666666667$$

Como el sistema obedece a ley de Henry las funciones de equilibrio son:

$$f_x = 242.666666667 * x$$

$$f_y = y / 242.666666667$$

Se tiene entonces todos los datos para resolver el problema empleando el programa *ab1c1*:

```

« 350. 0. 200. .45
« 242.666666667 * »
« 242.666666667 / »
ab1c1
4. →LIST
{ L1 x1 V1 y1 } →TAG
LIST→ DROP
»
  
```

Haciendo correr el programa (con EVAL) se obtiene:

```

L1: 350.647657526
x1: 1.84703223414E-3
V1: 199.352342474
y1: .448213155485
  
```

Que son los flujos y composiciones de salida requeridos.

4.2.1.2. Absorción de metanol en una etapa de equilibrio

Se emplearán 1000 kgmol/h de agua pura en un equipo de absorción equivalente a una etapa de equilibrio, para recuperar metanol de 400 kgmol/h de una mezcla de metanol-aire que se encuentra a una presión de 720 mm de Hg y una temperatura de 45°C. La presión parcial del metanol en esta mezcla es de 85 mm de Hg. Para este sistema se cuenta con los siguientes datos de equilibrio:

X_A	P_A (39.9°C)	P_A (59.4°C)
0.00	0	0
0.05	25.0	50

0.10	46.0	102
0.15	66.5	151

Calcule los flujos y composiciones de salida así como el porcentaje de metanol recuperado.

Solución

Los datos con los que contamos son los siguientes:

$L_0 = 1000$ kgmol/h

$x_0 = 0$

$V_0 = 400$ kgmol/h

$P_{A0} = 85$ mmHg

$P = 720$ mmHg

$\Rightarrow y_0 = 85 \cdot 720 / 5 \text{ RND} = .11806$

Se tienen además los datos de equilibrio aunque no a la temperatura del problema, por lo que deben ser interpolados para 45°C :

```
« [ 0 25 46 66.5 ] [ 0 50 102 151 ]
  → PA1 PA2
  « 1 4 FOR i
    45 39.9 59.4 PA1 i GET PA2 i GET
    inlin 1 RND
  NEXT
  4 →ARRY
  »
»
```

Haciendo correr el programa se obtiene:

[0. 31.5 60.6 88.6]

O dividiendo entre la presión de operación:

$Y_A = [0. 31.5 60.6 88.6] \cdot 720 / 5 \text{ RND} = [0. .04375 .08417 .12306]$

Por lo tanto los datos de equilibrio para la temperatura de operación son:

X_A	P_A (45°C)	Y_A
0.00	0.00	0.00000
0.05	31.5	0.04375
0.10	60.6	0.08417
0.15	88.6	0.12306

Se cuenta entonces con todos los datos para resolver el problema empleando el programa *ablcl*. Para no escribir dos veces los datos de equilibrio es conveniente guardarlos en dos variables:

```
[ 0 0.05 0.10 0.15 ] 'vx' STO
[ 0 .04375 .08417 .12306 ] 'vy' STO
```

Entonces las funciones de equilibrio son:

$f_x = \ll vx \text{ vy fun1 } \gg$

$f_y = \ll vy \text{ vx fun1 } \gg$

Las composiciones y flujos de salida son:

```
« 1000 0 400 0.11806
  « vx vy fun1 »
  « vy vx fun1 »
  ablcl
  4 →LIST 5 RND
  { L1 x1 V1 y1 } →TAG
```

LIST→ DROP

»

Haciendo correr el programa se obtiene:

L1: 1036.12354
 x1: .03486
 V1: 363.87646
 y1: .03051

Entonces el porcentaje de metanol recuperado es:

1036.12354 .03486 * 1000 0.11806 * / 100 * 2 RND = 30.59%

Si bien el caso donde se conocen todos los flujos y composiciones de entrada es el caso más frecuente, en una etapa de equilibrio existen 4 parámetros de entrada y 4 de salida, por lo tanto se pueden presentar alrededor de 24 casos, no resulta práctico entonces analizar y programar cada uno de estos casos, por lo que para los casos menos frecuentes se debe recurrir a las ecuaciones (3) a (7) y resolverlas para el caso en particular.

Analicemos por ejemplo otro caso frecuente: aquel donde se conoce la composición y flujo iniciales de la fase gaseosa (V_0, y_0), la composición inicial de la fase líquida (x_0) y el porcentaje de soluto a absorber (pa).

Con el porcentaje a absorber podemos calcular el flujo final de soluto en la fase gaseosa (V_1y_1):

$$pa = \left(\frac{V_0y_0 - V_1y_1}{V_0y_0} \right) 100 = \left(1 - \frac{V_1y_1}{V_0y_0} \right) 100$$

$$\frac{V_1y_1}{V_0y_0} = 1 - \frac{pa}{100} \quad (12)$$

$$V_1y_1 = \left(1 - \frac{pa}{100} \right) V_0y_0$$

La composición final en la fase gaseosa puede ser calculada tomando en cuenta que el flujo de gas inerte (C), al igual que de líquido inerte (B), es constante:

$$C = V_0(1 - y_0) = V_1(1 - y_1) = cte.$$

$$C = V_1 - V_1y_1 \quad (13)$$

$$V_1 = C + V_1y_1$$

Entonces la fracción final de soluto en la fase gaseosa (y_1) se calcula con:

$$y_1 = \frac{V_1y_1}{V_1} \quad (14)$$

Puesto que los flujos finales están en equilibrio x_1 puede ser calculado con la relación de equilibrio:

$$x_1 = f_y(y_1) \quad (15)$$

Del balance de materia (ecuación 3) se tiene:

$$V_0 - V_1 = L_1 - L_0 = D \quad (16)$$

Del flujo de líquido inerte constante (ecuación 5), se puede despejar el flujo final de líquido (L_1):

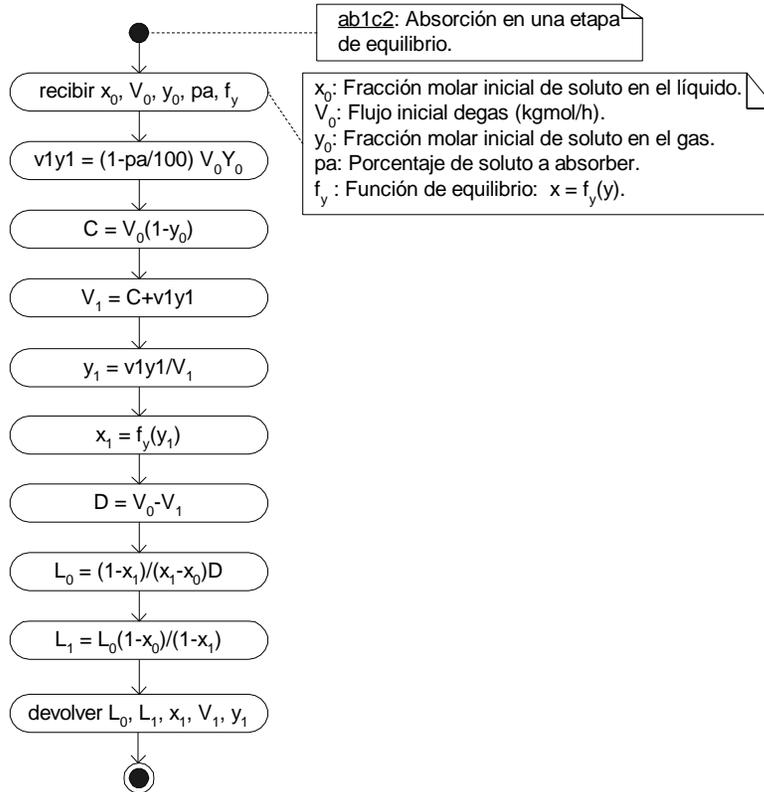
$$L_1 = L_0 \frac{1-x_0}{1-x_1} \tag{17}$$

Y reemplazando esta ecuación en la ecuación (16) obtenemos L_0 :

$$L_0 \frac{1-x_0}{1-x_1} - L_0 = L_0 \left(\frac{1-x_0}{1-x_1} - 1 \right) = L_0 \left(\frac{x_1 - x_0}{1-x_1} \right) = D \tag{18}$$

$$L_0 = \left(\frac{1-x_1}{x_1 - x_0} \right) D$$

Con lo que el sistema queda resuelto y como se puede ver, en este caso, la solución no es iterativa. El diagrama de actividades para automatizar el cálculo de este caso es el siguiente:



Y el programa elaborado en base al mismo es el siguiente:

```

xNAME ab1c2 ( absorción en una etapa de equilibrio: caso 2 )
::
CK5&Dispatch ( Datos: x0= Fracción inicial en el líquido )
# 11118 ( V2,y2= Flujo y fracción inicial en el gas )
:: ( pa= Porcentaje de soluto a absorber )
%0 EIGHT NDUPN DROP ( fy= Función de equilibrio x=fy[y] )
{ LAM x0 LAM V0 LAM y0 LAM pa LAM fy
  LAM v1y1 LAM C LAM V1 LAM y1 LAM x1
  LAM D LAM L0 LAM L1 }
BIND
%1 LAM pa %100 %/ %- LAM V0 LAM y0 %* %* ' LAM v1y1 STO
LAM V0 %1 LAM y0 %- %* ' LAM C STO
LAM C LAM v1y1 %+ ' LAM V1 STO
LAM v1y1 LAM V1 %/ ' LAM y1 STO
LAM y1 LAM fy EVAL ' LAM x1 STO
  
```

```

LAM V0 LAM V1 %- ' LAM D STO
%1 LAM x1 %- LAM x1 LAM x0 %- %/ LAM D %* ' LAM L0 STO
LAM L0 %1 LAM x0 %- %* %1 LAM x1 %- %/ ' LAM L1 STO
LAM L0 LAM L1 LAM x1 LAM V1 LAM y1
ABND

```

```

;
;

```

4.2.2. Ejemplo

Se empleará agua pura en un equipo de absorción equivalente a una etapa de equilibrio, para recuperar el 99% del metanol existente en 400 kgmol/h de una mezcla de metanol-aire que se encuentra a una presión de 720 mm de Hg y una temperatura de 45°C. La presión parcial del metanol en esta mezcla es de 85 mm de Hg. Los datos de equilibrio son los mismos que en el ejemplo 4.2.1.2. Calcule los flujos y composiciones de las corrientes faltantes.

Los datos con los que contamos son los siguientes:

```

x0 = 0
V0 = 400 kgmol/h
PA0 = 85 mmHg
P = 720 mmHg => y0 = 85 720 / 5 RND = .11806
pa = 99%

```

Y los datos de equilibrio:

```

[ 0 0.05 0.10 0.15 ] 'vx' STO
[ 0 .04375 .08417 .12306 ] 'vy' STO

```

Se tiene entonces todos los datos para resolver el problema empleando el programa *ablc2*:

```

« 0 400 0.11806 99 « vy vx fun1 »
ablc2
5 →LIST
{ L0 L1 x1 V1 y1 } →TAG
LIST→ DROP
»

```

Haciendo correr el programa se obtiene:

```

L0: 30553.3770299
L1: 30600.1287899
x1: 1.52782886376E-3
V1: 353.24824
y1: 1.33685025579E-3

```

Como se puede observar el flujo de agua necesario para recuperar el 99% del metanol en una etapa de equilibrio se requiere más de 76 veces del flujo del gas, lo que es muy ineficiente.

4.2.3. Ejercicios

- Se empleará un equipo de absorción equivalente a una etapa de equilibrio para absorber SO₂ de una mezcla SO₂-aire, empleando agua pura a 293.2 K. El gas se encuentra a una presión de 2.51x10⁵ Pa y la presión parcial inicial del SO₂ en la mezcla es 1.76x10⁴ Pa. El flujo inicial del gas es de 6.4 kgmol/h y el del agua 330 kgmol/h. Calcule los flujos y composiciones de las corrientes de salida así como el porcentaje de soluto recuperado. Los datos de equilibrio son los siguientes:

x_A	P_A (mm Hg)
0.0000000	0.0
0.0000562	0.5
0.0001403	1.2
0.0002800	3.2
0.0004220	5.8
0.0005640	8.5
0.0008420	14.1
0.0014030	26.0
0.0019650	39.0
0.0027900	59.0
0.0042000	92.0
0.0069800	161.0
0.0138500	336.0
0.0206000	517.0
0.0273000	698.0

4. Repita el ejercicio 3 asumiendo que el sistema obedece la ley de Henry. La constante de Henry a 293.2 K es 18.09 atm.
5. Si en el sistema del ejercicio 3 se quiere recuperar el 70% del SO₂ disuelto. Calcule los flujos inicial y final de agua así como las fracciones molares en las fases gaseosa y líquida.
6. Si en el sistema del ejercicio 3 la fracción molar final de soluto en la fase gaseosa es 7×10^{-3} . Calcule los flujos inicial y final de agua, las fracciones molares en las fases gaseosa y líquida y el porcentaje de soluto recuperado.
7. Si en el sistema del ejercicio 3 se emplean 2 equipos de absorción, cada uno equivalente a una etapa de equilibrio, donde en el primer equipo se recupera el 30% de soluto, en el segundo el 25% y a ambos se alimenta agua pura a 293.2 K. ¿Cuáles serían los flujos y composiciones en cada uno de los equipos?.
8. Si en el sistema del ejercicio 3 se emplean 3 equipos de absorción, donde en la fase líquida a cada equipo se alimenta 120 kgmol/h de agua pura. En la fase gaseosa, el gas resultante de la primera etapa se alimenta a la segunda y el resultante de la segunda se alimenta a la tercera. Calcule los flujos y composiciones faltantes así como el porcentaje de soluto recuperado.
9. Se empleará un equipo de absorción equivalente a una etapa de equilibrio para absorber el 90% del amoníaco contenido en 65 kgmol/h de una mezcla gaseosa de amoníaco - aire, que contiene 0.12 en fracción molar de amoníaco. La presión del gas es de 1.723×10^5 Pa y para la absorción se emplea agua pura a 293.2 K. Calcule los flujos y composiciones faltantes. Los datos de equilibrio para este sistema son:

x_A	P_A (mm Hg)
0.0000	0.0
0.0208	12.0
0.0258	15.0
0.0309	18.2
0.0405	24.9
0.0503	31.7
0.0737	50.0

0.0960	69.6
0.1370	114.0
0.1750	166.0
0.2100	227.0
0.2410	298.0
0.2970	470.0

10. Se empleará un equipo de absorción equivalente a una etapa de equilibrio para reducir el porcentaje de acetona contenido en 40 kgmol/h, de una mezcla gaseosa de acetona - aire, desde 5% hasta 1%. Para ese fin se empleará agua pura a 293.2 K. Calcule los flujos y composiciones faltantes así como el porcentaje de acetona recuperado. Los datos de equilibrio son los siguientes:

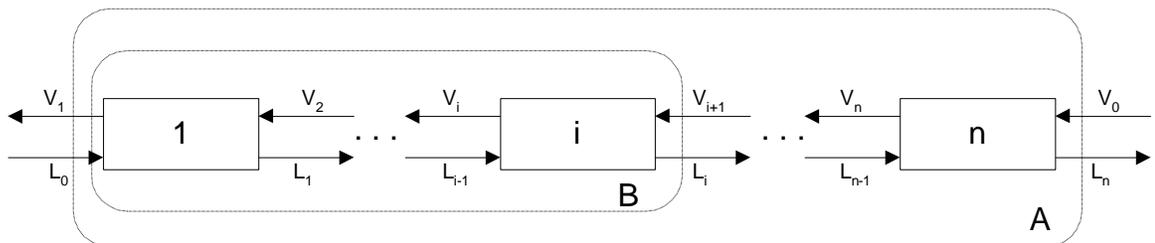
x_A	P_A (mm Hg)
0.0000	0.0
0.0333	30.0
0.0720	62.8
0.1170	85.4
0.1710	103.0

11. Repita el ejercicio anterior asumiendo que el sistema obedece la ley de Henry, siendo la constante de Henry igual a 2.53 atm.

4.3. Torres de absorción con múltiples etapas en contracorriente

En la mayoría de los casos prácticos no resulta económicamente factible emplear una etapa de equilibrio para lograr la separación requerida. Una disposición más eficiente se logra si se emplea un equipo con múltiples etapas donde las fases gaseosa y líquida fluyen en contracorriente.

El esquema general de un equipo de múltiples etapas en contracorriente es el siguiente:



Donde se han identificado dos sistemas: "A" con las "n" etapas del equipo y "B" con las primeras "i" etapas del equipo.

En un equipo con múltiples etapas de equilibrio se introduce una nueva variable: el número de etapas "n", por lo tanto se requiere un dato más para resolver el sistema.

Analicemos primero el sistema "A". En este sistema, dependiendo de los datos conocidos, se presentan múltiples casos siendo el más frecuente aquel donde se conocen los flujos y composiciones de entrada (L_0, x_0, V_0, y_0) y el porcentaje de soluto absorbido en el equipo (p_a).

El balance general de materia en este sistema es:

$$L_0 + V_0 = L_n + V_1 = M \tag{19}$$

Donde "M" es el flujo total de masa que ingresa (y sale) del equipo. El balance de materia para el soluto es:

$$L_0x_0 + V_0y_0 = L_nx_n + V_1y_1 = A \quad (20)$$

Donde "A" es el flujo total de soluto que ingresa y sale del equipo. Al igual que en una etapa de equilibrio, se asumen constantes los flujos de líquido (B) y gas inerte (C), por lo tanto:

$$B = L_0(1-x_0) = L_1(1-x_1) = \dots = L_n(1-x_n) = \text{ctte.} \quad (21)$$

$$C = V_0(1-y_0) = V_1(1-y_1) = \dots = V_n(1-y_n) = \text{ctte.} \quad (22)$$

Por supuesto se debe contar también con los datos de equilibrio:

$$y = f_x(x) \quad (6)$$

$$x = f_y(y) \quad (7)$$

Entonces, para el caso más frecuente, con el porcentaje de soluto absorbido (*pa*) se puede calcular el flujo final de soluto en la fase gaseosa (V_1y_1):

$$V_1y_1 = \left(1 - \frac{pa}{100}\right) V_0y_0 \quad (12)$$

Y con el flujo de gas inerte (C) el flujo final de la fase gaseosa (V_1):

$$V_1 = C + V_1y_1 \quad (13)$$

Entonces la fracción final de soluto en la fase gaseosa (y_1) se calcula con:

$$y_1 = \frac{V_1y_1}{V_1} \quad (14)$$

Del balance general de materia obtenemos el flujo final de la fase líquida (L_n):

$$L_n = M - V_1 \quad (23)$$

Finalmente, del balance de materia para el soluto obtenemos la fracción final de soluto en la fase líquida (x_n):

$$x_n = \frac{A - V_1y_1}{L_n} \quad (24)$$

Entonces la solución del sistema "A" para el caso más frecuente no es iterativa. El algoritmo que automatiza estos cálculos se presenta en el diagrama de actividades de la siguiente página y el código elaborado en base al mismo es el siguiente:

```
xNAME abnc1 ( absorción en múltiples etapas de equilibrio: caso 1 )
::
CK5&Dispatch
# 11111
::
%0 EIGHT NDUPN DROP
{ LAM L0 LAM x0 LAM V0 LAM y0 LAM pa
  LAM M LAM A LAM v1y1 LAM C LAM V1
  LAM y1 LAM Ln LAM xn }
BIND
LAM L0 LAM V0 %+ ' LAM M STO
LAM L0 LAM x0 %* LAM V0 LAM y0 %* %+ ' LAM A STO
%1 LAM pa %100 %/ %- LAM V0 LAM y0 %* %* ' LAM v1y1 STO
```

```

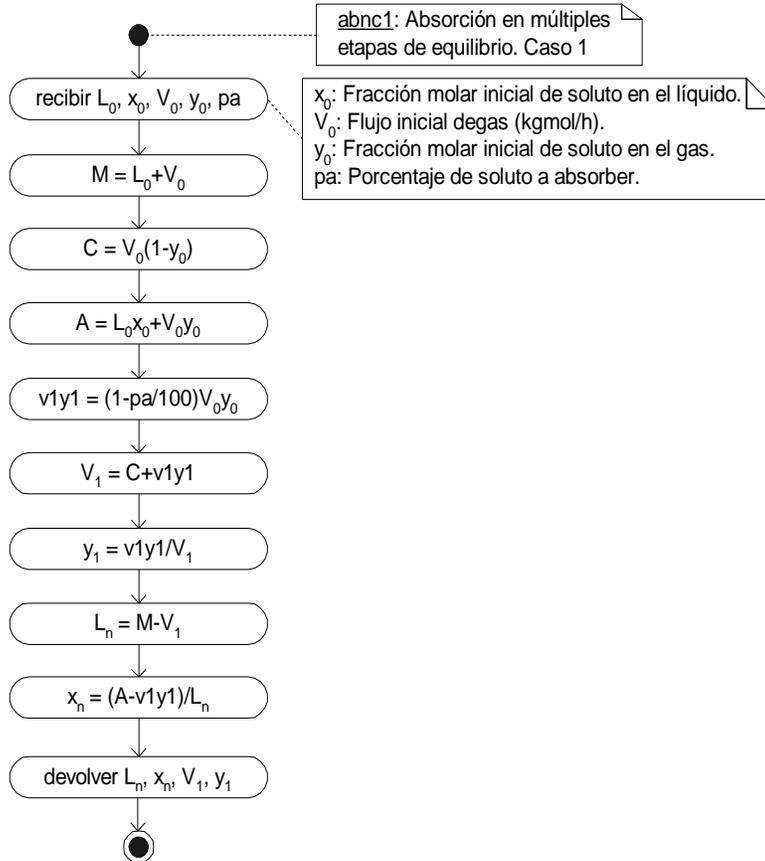
LAM V0 %1 LAM y0 %- %* ' LAM C STO
LAM C LAM v1y1 %+ ' LAM V1 STO
LAM v1y1 LAM V1 %/ ' LAM y1 STO
LAM M LAM V1 %- ' LAM Ln STO
LAM A LAM v1y1 %- LAM Ln %/ ' LAM xn STO
LAM Ln LAM xn LAM V1 LAM y1
ABND

```

```

;
;

```



4.3.1. Ejemplo

Se desea absorber el 90% de la acetona de un gas que contiene 1 mol% de acetona en aire, empleando una torre con múltiples etapas en contracorriente. El flujo inicial del gas es de 30 kgmol/h y el flujo de agua pura 90 kgmol/h. Calcule los flujos y composiciones de las corrientes que salen del equipo.

Solución

Los datos con los que contamos son:

$L_0 = 90$ kgmol/h
 $x_0 = 0$
 $V_0 = 30$ kgmol/h
 $y_0 = 0.01$
 $pa = 90$

Se cuenta entonces con todos los datos para resolver el problema con el programa *abnc1*:

```

« 90 0 0.01 30 90 □ pa x0 y0 V0 L0
  « L0 x0 V0 y0 pa abncl
    4 □LIST
    { Ln xn V1 y1 } □TAG
    LIST□ DROP
  »
»

```

Haciendo correr el programa se obtiene:

```

Ln: 90.27
xn: 2.99102691924E-3
V1: 29.73
y1: 1.00908173562E-3

```

Que son los flujos y composiciones finales pedidos.

4.3.2. Ejercicios

12. Se diseñará una torre de absorción, con múltiples etapas en contracorriente, para absorber SO_2 de una mezcla de SO_2 -aire, empleando 6000 kg/h.m^2 de agua pura a 293 K . Al equipo ingresarán 150 kg/h.m^2 , de gas con un 20% en peso de soluto y se desea que a la salida el porcentaje de soluto sea reducido a 2% en peso. Calcule el porcentaje de soluto absorbido en el equipo, los flujos de salida y el porcentaje final de soluto en la fase líquida.
13. Se empleará una torre de absorción, con múltiples etapas en contracorriente, para absorber el 95% del amoníaco contenido en 65 kgmol/h , de una mezcla de amoníaco-aire, que contiene 20 mol% de amoníaco. Al equipo ingresan 80 kgmol/h de agua pura a 293.2 K . Calcule los flujos y composiciones de las corrientes que salen del equipo.
14. Se empleará una torre de absorción, con múltiples etapas en contracorriente, para absorber el 99% de la acetona contenida en 40 kgmol/h , de una mezcla de acetona-aire que tiene un porcentaje inicial de acetona del 5% (porcentaje molar). Calcule los flujos y composiciones de las corrientes que salen del equipo.

Una vez resuelto el sistema "A", podemos pasar a resolver el sistema "B". Del balance general de materia en este sistema se obtiene:

$$\begin{aligned} L_0 + V_{i+1} &= L_i + V_1 \\ L_0 - V_1 &= L_i - V_{i+1} = D_M \end{aligned} \quad (25)$$

Donde " D_M ", que es el diferencial entre la masa que ingresa y sale de una etapa por un mismo lado, es constante y puede ser calculada pues los valores de L_0 y V_1 se obtienen al resolver el sistema "A". Del balance de materia para el soluto resulta:

$$L_0 x_0 - V_1 y_1 = L_i x_i - V_{i+1} y_{i+1} = D_A \quad (26)$$

Donde " D_A ", que es la diferencia entre la masa de soluto que ingresa y sale de una etapa por un mismo lado, es también una constante y puede ser calculada pues los valores de L_0 , x_0 , V_1 y y_1 se obtienen al resolver el sistema "A". De esta ecuación podemos despejar la fracción de soluto que ingresa a la etapa (y_{i+1}):

$$y_{i+1} = \frac{L_i x_i - D_A}{V_{i+1}} \quad (27)$$

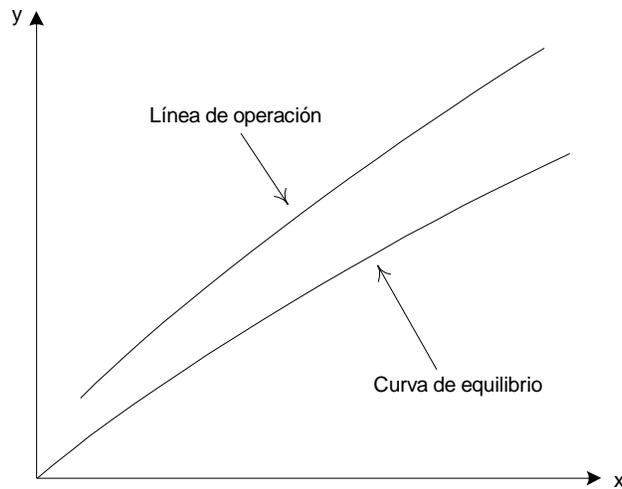
Esta ecuación es conocida como la línea de operación pues con ella se puede trazar una línea en una gráfica x versus y , dándose valores de x_i y calculando los correspondientes valores de y_{i+1} . Para ello se toma en cuenta que el flujo de líquido inerte es constante (ecuación 21):

$$L_i = \frac{B}{(1-x_i)} \quad (28)$$

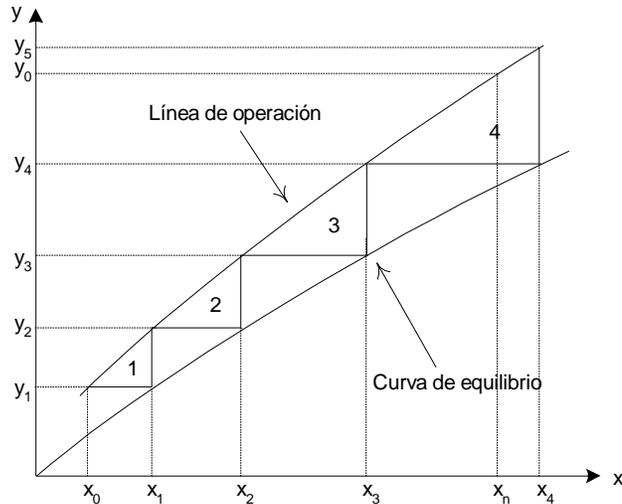
Y que el flujo de gas que entra a la etapa (V_{i+1}) puede ser obtenido del balance de materia (ecuación 25):

$$V_{i+1} = L_i - D_M \quad (29)$$

Con la curva de equilibrio y la línea de operación, trazadas en una gráfica " x " versus " y ", como se muestra en la siguiente figura:



Se puede determinar el número de etapas teóricas de una manera similar al método de *McCabe -Thiele*. Comenzando con la fracción inicial de soluto en la fase líquida " x_0 " se intersecta la línea de operación (ver la siguiente figura), lo que nos da el valor de y_1 , entonces con y_1 se intersecta la curva de equilibrio, lo que nos da x_1 , luego con x_1 se intersecta la línea de operación obteniéndose y_2 y con y_2 se intersecta la curva de equilibrio obteniéndose x_2 . Se continúa de esta manera hasta que el valor de x_i es mayor o igual a la fracción final de soluto en la fase líquida " x_n " (o hasta que el valor de y_i es mayor o igual a la fracción inicial de soluto en la fase gaseosa " y_0 "). Cuando se cumple una de estas condiciones el proceso concluye y entonces se cuenta el número de veces que se ha repetido el proceso, lo que corresponde al número de etapas teóricas.



Así por ejemplo en la figura se requieren casi 4 etapas teóricas para lograr la separación.

Puesto que contamos con las ecuaciones analíticas para la línea de operación y con las funciones para los datos de equilibrio, podemos resolver el problema empleando directamente las ecuaciones en lugar de seguir el procedimiento gráfico.

Para ello comenzamos con la etapa 1, es decir $i=1$. En esta etapa, como resultado del sistema "A", se conoce la fracción final de soluto en la fase gaseosa (y_1) y como está en equilibrio con la fase líquida, podemos calcular el valor de x_1 con la función de equilibrio:

$$x_i = f_y(y_i)$$

Entonces con la ecuación (28) calculamos el valor de L_i , con la ecuación (29) el valor de V_{i+1} y con la ecuación (27) el valor de y_{i+1} (y_2). Ahora podemos repetir el proceso en la segunda etapa $i=2$, pues conocemos el valor de y_2 , de esa manera obtendremos el valor de y_3 y con este valor se puede repetir el proceso para la tercera etapa, continuando de esa manera hasta que el valor de x_i sea mayor o igual a la fracción final de soluto en la fase líquida (x_n) o hasta que el valor de y_{i+1} sea mayor o igual a la fracción inicial de soluto en la fase gaseosa (y_0).

El algoritmo que automatiza el proceso de cálculo es el que se muestra en el diagrama de actividades de la siguiente página, en el mismo el nuevo valor de y_{i+1} se guardan en la variable y_1 y no se conservan los valores de y_{i+1} , pues sólo se utilizan una vez. En V_{i+1} se ha empleado la letra "j" en lugar de "i+1". La fracción útil de la última etapa se obtiene simplemente como la razón entre la separación requerida (x_n-x_{i-1}) y la separación lograda (x_i-x_{i-1}). El valor de x_{i-1} se almacena en la variable x_0 .

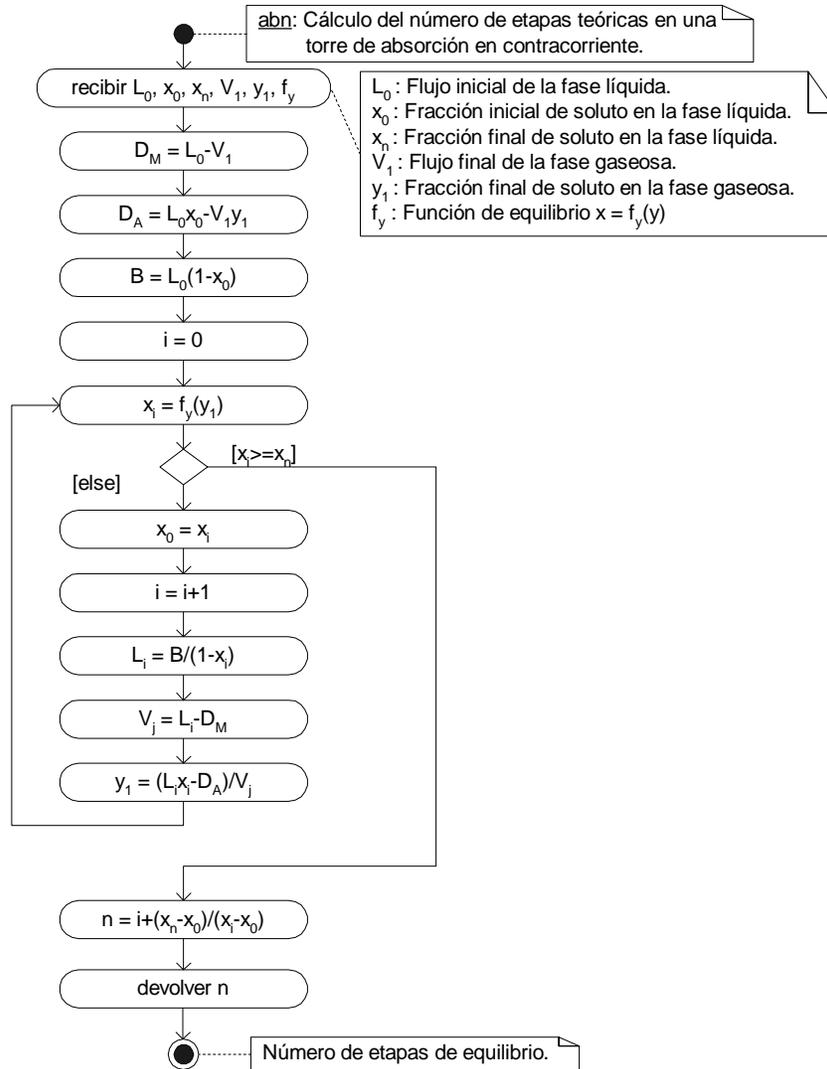
El código elaborado siguiendo el algoritmo es el siguiente:

```
xNAME abn          ( Etapas de equilibrio en una torre )
::                ( de absorción en contracorriente )
%6 CKNNOLASTWD DROP ( Datos: )
CK&DISPATCH1     ( L0=13= flujo inicial de líquido; )
# 11118          ( x0=12= fracción inicial de soluto en el líquido )
::              ( xn=11= fracción final de soluto en el líquido )
6ROLL            ( V1=10= flujo final de gas )
CK&DISPATCH1     ( y1= 9= fracción final de soluto en el gas )
ONE              ( fy= 8= función de equilibrio x=fy[y] )
```

```

::          ( variables locales: )
6UNROLL      ( DM=7; DA=6; B=5; i=4; xi=3; Li=2; Vj=1 )
%0 SEVEN NDUPN DROP
' NULLLAM THIRTEEN NDUPN DOBIND
13GETLAM 10GETLAM %- 7PUTLAM          ( DM = L0-V1 )
13GETLAM 12GETLAM %* 10GETLAM
9GETLAM %* %- 6PUTLAM                ( DA = L0x0-V1y1 )
13GETLAM %1 12GETLAM %- %* 5PUTLAM   ( B = L0[1-x0] )
::
  BEGIN
    9GETLAM 8GETLAM EVAL 3PUTLAM      ( xi = fy[y1] )
    3GETLAM 11GETLAM %>= IT :: 2RDROP ; ( si xi>=xn => salir )
    3GETLAM 12PUTLAM                  ( x0=xi )
    4GETLAM %1+ 4PUTLAM               ( i= i+1 )
    5GETLAM %1 3GETLAM %- %/ 2PUTLAM  ( Li = B/[1-xi] )
    2GETLAM 7GETLAM %- 1PUTLAM       ( Vj = Li-DM )
    2GETLAM 3GETLAM %* 6GETLAM %-
    1GETLAM %/ 9PUTLAM                ( y1= [Lixi-DA]/Vj )
  AGAIN
;
  4GETLAM 11GETLAM 12GETLAM %-
  3GETLAM 12GETLAM %- %/ %+          ( n = i+[xn-x0]/[xi-x0] )
  ABND
;
ZERO
::
  6UNROLL # 202 DO#EXIT              ( 202 = Bad Argument Type )
;
;
;

```



4.3.3. Ejemplo

4.3.3.1. Absorción de acetona

Se desea absorber el 99% de la acetona de un gas que contiene 9 mol% de acetona en aire, empleando una torre con múltiples etapas en contracorriente. El flujo inicial del gas es de 30 kgmol/h y el flujo de agua pura 90 kgmol/h. La presión de operación de la torre es de 1.5×10^5 Pa y la temperatura de 293.2 K. a) Calcule el número de etapas teóricas que deberá tener el equipo para lograr la separación requerida; b) Calcule el número de etapas reales si la eficiencia de cada etapa es del 60%. Los datos de equilibrio son los del ejercicio 10.

Solución

Los datos con los que contamos son:

$$L_0 = 90 \text{ kgmol/h}$$

$$x_0 = 0$$

$$V_0 = 30 \text{ kgmol/h}$$

$$y_0 = 0.09$$

$$pa = 99$$

$$P = 7.2 \times 10^4 \text{ Pa} = 7.2E4_Pa \text{ 1_mmHg CONVERT 2 RND} = 540.04_mmHg$$

Entonces contamos con todos los datos para resolver el sistema "A" empleando el programa *abnc1*:

```
« 90 0 30 0.09 99
  abnc1
  4 →LIST 5 RND
  { Ln xn V1 y1 } →TAG
  LIST→ DROP
»
```

Haciendo correr el programa se obtiene:

```
Ln: 92.673
xn: .02884
V1: 27.327
y1: .00099
```

Que son los flujos y composiciones a la salida del equipo redondeados en el quinto decimal.

Para las funciones de equilibrio guardamos los datos del vector "x" en la variable "vx":

```
[ 0.0000 0.0333 0.0720 0.1170 0.1710 ] 'vx' STO
```

Para el vector "y" dividimos el vector de presiones entre la presión total y guardamos el vector resultante en la variable "vy":

```
[ 0 30 62.8 85.4 103 ] 540.04 / 4 RND 'vy' STO
```

a) Contamos ahora con todos los datos para calcular el número de etapas teóricas empleando el programa *abn*:

```
« 90 0 0.02884 27.327 0.00099
  « vy vx fun1 »
  abn 2 RND
  "n" →TAG
»
```

Haciendo correr el programa se obtiene:

```
n: 5.75
```

Que es el número de etapas teóricas necesarias para lograr la separación requerida.

b) El número de etapas reales se calcula simplemente dividiendo el número de etapas teóricas entre la eficiencia:

Número de etapas reales = $5.75 \cdot 0.6 / 2 \text{ RND} = 9.58$

4.3.3.2. Reutilización de un equipo existente

Una industria cuenta con una torre de absorción con múltiples etapas en contracorriente equivalentes a 12 etapas teóricas. Se quiere emplear este equipo para absorber el SO₂ contenido en una mezcla de SO₂ - aire. La concentración inicial de SO₂ en el gas es del 20 % (molar) y el flujo de aire inerte es de 160 kg/h.m². Para la absorción se empleará agua pura con un flujo inicial igual a 3785 kg/h.m². El equipo opera a una presión de 1x10⁵ Pa. Calcule el porcentaje de SO₂ de azufre que puede ser absorbido en el equipo. Los datos de equilibrio son los del ejercicio 3.

Solución

Los datos con los que se cuenta son los siguientes:

n = 12

```

y0 = 0.2
C = 160 kg/h.m² = 160 28.97 / 2 RND = 5.52 kgmol/h.m²
L0 = 3785 kg/h.m² = 3785 18.02 / 2 RND = 210.04 kgmol/h.m²
x0 = 0
P = 1x105 Pa = 1E5_Pa 1_mmHg CONVERT 2 RND = 750.06_mmHg

```

Se tiene además los datos de equilibrio que por comodidad se guardan en variables globales:

```

[ 0 0.0000562 0.0001403 0.00028 0.000422 0.000564 0.000842 0.001403 0.001965
  0.00279 0.0042 0.00698 0.01385 0.0206 0.0273 ] 'vx' STO
[ 0 0.5 1.2 3.2 5.8 8.5 14.1 26 39 59 92 161 336 517 698 ] 750.06 / 7 RND
'vy' STO

```

Puesto que el flujo de gas inerte es constante, con "C" podemos calcular V_0 :

```
V0 = 5.52 1 0.2 - / 2 RND = 6.9 kg/h.m²
```

En este caso no contamos con el porcentaje absorbido o la fracción final de soluto en la fase gaseosa. Entonces la solución del problema es iterativo: a) se asume un porcentaje de absorción (pa); b) con el valor asumido y el programa "abnc1" se calculan los flujos y composiciones de salida (L_n , x_n , V_1 , y_1); c) con los flujos y composiciones de salida calculados en el inciso anterior y el programa "abn" se calcula el número de etapas teóricas; c) se compara el número etapas calculadas con el número de etapas conocidas, si son iguales el proceso concluye, siendo el porcentaje absorbido el valor asumido (pa), caso contrario el proceso se repite desde el inciso (a).

Puesto que el problema iterativo es función de una sola variable: " pa ", puede ser resuelto empleando uno de los métodos iterativos como el método de la Secante o Regula Falsi. La función a resolver es:

$$f(pa) = N^\circ \text{ de etapas calculadas} - N^\circ \text{ de etapas conocidas} = 0$$

El programa que resuelve esta función, empleando el método de Regula Falsi, buscando la solución entre 1% y 99% es el siguiente:

```

« 12 0.2 160 28.97 / 3785 18.02 / 0
  1E5_Pa 1_mmHg CONVERT UVAL
  [ 0 0.0000562 0.0001403 0.00028 0.000422 0.000564
    0.000842 0.001403 0.001965 0.00279 0.0042
    0.00698 0.01385 0.0206 0.0273 ]
  [ 0 0.5 1.2 3.2 5.8 8.5 14.1 26 39 59 92 161 336
    517 698 ] 0 0 0 0 0 0
  □ n y0 C L0 x0 P vx vy fy V0 Ln xn V1 Y1
  « vy P / 'vy' STO « vy vx fun1 » 'fy' STO
  C 1 y0 - / 'V0' STO
  « 'pa' STO
    L0 x0 V0 y0 pa abnc1 4 □LIST
    { Ln xn V1 y1 } STO
    L0 x0 xn V1 y1 fy abn n -
  »
  1 99 Refa "pa" □TAG
»
»
»

```

Haciendo correr el programa se obtiene:

```
pa: 97.4632670011
```

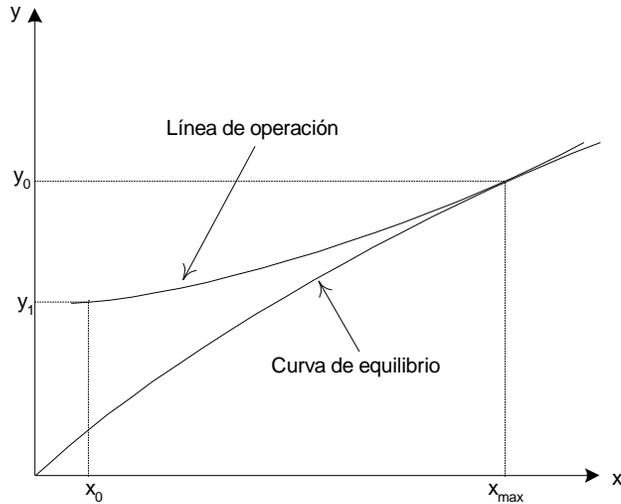
Que es el porcentaje de soluto que puede ser absorbido en el equipo.

4.3.4. Ejercicios

15. Se empleará una torre de absorción, con múltiples etapas en contracorriente, para absorber el 97% del metanol contenido en 400 kgmol/h de una mezcla de metanol-aire que se encuentra a una presión de 1 atm y una temperatura de 55°C. La presión parcial inicial del metanol en esta mezcla es de 120 mm de Hg y para la absorción se emplearán 700 kgmol/h de agua pura. a) Calcule el número de etapas teóricas para lograr la separación requerida. b) Calcule el número de etapas reales si el rendimiento de cada etapa es del 35%. Utilice los datos de equilibrio del ejemplo 4.2.1.2.
16. Se empleará agua pura, en una torre de absorción, con múltiples etapas en contracorriente, para reducir el porcentaje de CO₂ de una mezcla de CO₂ - aire que se encuentra a 10 atm y una temperatura de 0°C. El porcentaje inicial (molar) de CO₂ en la mezcla es del 70% y se quiere reducir el mismo a 20%. El flujo inicial del aire es de 250 kgmol/h y el de agua 16100 kgmol/h. Calcule el número de etapas teóricas que deberá tener el equipo para lograr la absorción requerida. Asuma que el sistema obedece la ley de Henry, siendo la constante de Henry 0.0728x10⁴ atm.
17. Una industria cuenta con una torre de absorción equivalente a 4.3 etapas de equilibrio. Se quiere emplear esta torre para absorber amoníaco, con agua pura, de una mezcla de amoníaco - aire que se encuentra a 1.4x10⁵ Pa y 293.2 K. El flujo inicial de gas es de 80 kgmol/h y tiene un porcentaje (molar) de amoníaco igual al 23%. El flujo inicial de agua pura es de 90 kgmol/h. Calcule el porcentaje de soluto que puede ser absorbido en esta torre. Utilice los datos de equilibrio del ejercicio 9.
18. Una industria cuenta con una torre de absorción equivalente a 5.6 etapas teóricas que piensa emplear para absorber el 99.4% del amoníaco contenido en una mezcla de amoníaco - aire, que se encuentra a 0.89 atm y 293.2 K. El flujo inicial de gas es de 78 kgmol/h y tiene un porcentaje (molar) de amoníaco igual a 25%. Para la absorción se empleará agua pura a 293.2 K. Calcule el flujo de agua que deberá emplearse para lograr la absorción requerida. Utilice los datos de equilibrio del ejercicio 9.

4.4. Flujo mínimo de líquido

El flujo mínimo de líquido que puede ser empleado para lograr una absorción determinada, corresponde a la máxima concentración final de soluto que puede alcanzar la fase líquida (x_{\max}) y ocurre cuando la línea de operación se hace tangente a la curva de equilibrio, tal como se muestra en la siguiente figura:



Para estas condiciones el número de etapas teóricas es infinito.

El flujo mínimo puede ser encontrado por tanteo, haciendo correr los programas "abnc1" y "abn", para diferentes valores de L_0 , hasta que se encuentra el flujo mínimo para el cual todavía es posible calcular el número de etapas.

4.4.1. Ejemplo: Determinación del flujo mínimo por tanteo

Calcule el flujo mínimo de agua pura que puede ser empleada para recuperar el 99% de la acetona de un gas que contiene 9 mol% de acetona en aire. El flujo inicial del gas es de 30 kgmol/h y la presión de operación de la torre es de 7.2×10^4 Pa a una temperatura de 293.2 K. Los datos de equilibrio son los del ejercicio 10.

Solución

Los datos con los que contamos son:

```
x0 = 0
V0 = 30 kgmol/h
y0 = 0.09
pa = 99
P = 7.2x104 Pa = 7.2E4_Pa 1_mmHg CONVERT 2 RND = 540.04_mmHg
```

Se tienen también los datos de equilibrio:

```
[ 0.0000 0.0333 0.0720 0.1170 0.1710 ] 'vx' STO
[ 0 30 62.8 85.4 103 ] 540.04 / 4 RND 'vy' STO
```

La búsqueda del flujo mínimo la llevaremos a cabo con el siguiente programa, donde se recibe el flujo inicial de líquido " L_0 " y con el mismo se calculan las composiciones y flujos finales empleando el programa "abnc1" y con estos resultados se calcula el número de etapas teóricas con el programa "abn":

```
<< 0 30 0.09 99 0 0 0 0
  → L0 x0 V0 y0 pa Ln xn V1 y1
  <<
    L0 x0 V0 y0 pa
    abnc1
    4 →LIST
    { Ln xn V1 y1 } STO
    L0 x0 xn V1 y1
```

```

« vy vx fun1 »
abn 2 RND
"n" →TAG
»
»

```

El cual por comodidad se guarda en la variable "lmin" ('lmin' STO)

Para 90 kgmol/h se obtiene:

```
90 lmin => n:5.75
```

Como es un número finito de etapas vamos bajando de 10 en 10:

```
80 lmin => n:6.74
```

```
70 lmin => n:8.41
```

```
60 lmin => n:12.
```

```
50 lmin => n:26.97
```

```
40 lmin => No se obtiene resultado en más de 1 minuto
```

Para detener el programa, que ha ingresado a un ciclo infinito, debe pulsar la tecla ON.

Entonces como para 40 kgmol/h se requiere un flujo infinito el flujo mínimo debe estar comprendido entre 40 y 50 kgmol/h. Continuamos la búsqueda con 45 kgmol/h:

```
45 lmin => No se obtiene resultado en más de 1 minuto
```

Por lo tanto el flujo mínimo está entre 45 y 50 kgmol/h. Continuamos la búsqueda con 47 kgmol/h:

```
47 lmin => n:60.1
```

Ahora sabemos que el flujo mínimo está entre 45 y 47 kgmol/h. Continuamos la búsqueda con 46 kgmol/h:

```
46 lmin => n:183.82
```

Entonces el flujo mínimo está entre 45 y 46 kgmol/h. Continuamos la búsqueda con 45.5 kgmol/h:

```
45.5 lmin => No se obtiene resultado en más de 1 minuto
```

Por lo tanto el flujo mínimo está entre 45.5 y 46 kgmol/h. Continuamos la búsqueda con 45.7 kgmol/h:

```
45.7 lmin => No se obtiene resultado en más de 1 minuto
```

Ahora probamos con 45.9 kgmol/h:

```
45.9 lmin => n:305.03
```

Por lo tanto el flujo mínimo está comprendido entre 45.7 y 45.9 kgmol/h. Probamos entonces con 45.8 kgmol/h

```
45.8 lmin => No se obtiene resultado en más de 1 minuto
```

En consecuencia el flujo mínimo que se puede emplear para lograr la separación requerida (con un dígito de precisión después del punto) es 45.9 kgmol/h.

El flujo mínimo de líquido puede ser calculado también recurriendo a la ecuación del balance de materia para el soluto (ecuación 20), tomando en cuenta que para el flujo mínimo de líquido la fracción molar final de soluto x_n es máxima x_{max} . La fracción máxima de soluto en la fase líquida se deter-

mina con la función de equilibrio, pues está en equilibrio con la fracción inicial de soluto en la fase gaseosa (y_0):

$$x_{max} = f_y(y_0)$$

El flujo final de soluto V_1y_1 se calcula con la (ecuación 12):

$$V_1y_1 = \left(1 - \frac{pa}{100}\right)V_0y_0$$

Entonces de la ecuación del balance de materia (ecuación 7), se puede despejar el flujo mínimo de líquido inerte:

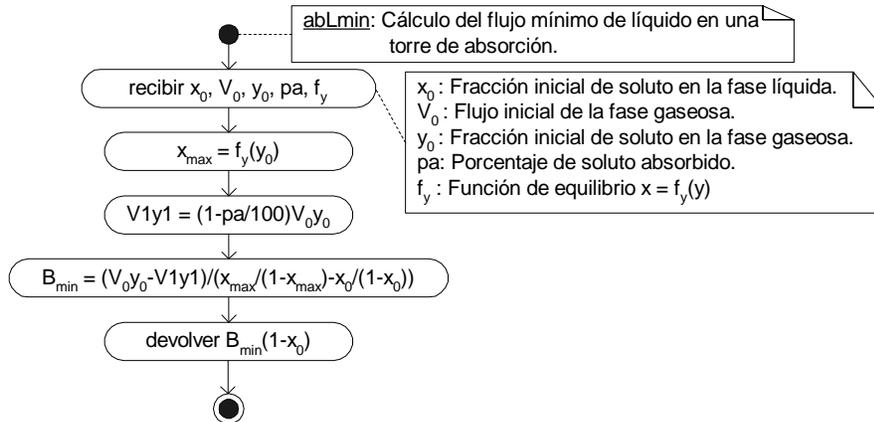
$$L_n x_{max} - L_0 x_0 = B_{min} \frac{x_{max}}{1-x_{max}} - B_{min} \frac{x_0}{1-x_0} = B_{min} \left(\frac{x_{max}}{1-x_{max}} - \frac{x_0}{1-x_0} \right) = V_0 y_0 - V_1 y_1$$

$$B_{min} = \frac{V_0 y_0 - V_1 y_1}{\frac{x_{max}}{1-x_{max}} - \frac{x_0}{1-x_0}} \tag{30}$$

Finalmente el flujo mínimo de líquido se calcula con:

$$L_{min} = B_{min} (1-x_0) \tag{31}$$

El algoritmo que automatiza el cálculo del flujo mínimo de líquido se presenta en el siguiente diagrama de actividades.



El código elaborado en base al algoritmo es el siguiente:

```

xNAME abLmin ( Flujo mínimo de líquido en una torre de absorción )
::
( Datos: )
CK5&Dispatch ( x0=8= Fracción inicial de soluto en el líquido)
# 11118 ( V0=7= Flujo inicial de gas )
::
( y0=6= Fracción inicial de soluto en el gas )
%0 THREE NDUPN DROP ( pa=5= Porcentaje de soluto absorbido )
' NULLLAM EIGHT ( fy=4= Función de equilibrio x=fy[y] )
NDUPN DOBIND ( xmax=3; V1y1=2; Bmin=1 )
6GETLAM 4GETLAM EVAL 3PUTLAM ( xmax= fy[y0] )
%1 5GETLAM %100 %/ %-
7GETLAM 6GETLAM %* %* 2PUTLAM ( V1y1= [1-pa/100]V0y0 )
7GETLAM 6GETLAM %* 2GETLAM %- ( [V0y0-V1y1] )
3GETLAM %1 3GETLAM %- %/ ( [xmax/[1-xmax] ] )
8GETLAM %1 8GETLAM %- %/ %- ( -x0/[1-x0] )
%/ 1PUTLAM ( Bmin )
1GETLAM %1 8GETLAM %- %* ( Lmin )
    
```

ABND

;
;

4.4.2. Ejemplo: Cálculo del flujo mínimo empleando ecuaciones analíticas

Calcule el flujo mínimo de líquido que puede emplearse para las condiciones del ejemplo 4.4.1, empleando las ecuaciones analíticas.

Solución

Tenemos los mismos datos que en el ejemplo 4.4.1, sólo que ahora el problema se resuelve directamente con el programa "abLmin":

```
« 0 99 0.09 30 1.5E5_Pa 1_mmHg CONVERT UVAL
  [ 0.0000 0.0333 0.0720 0.1170 0.1710 ]
  [ 0 30 62.8 85.4 103 ] 0
  □ x0 pa y0 V0 P vx vy fy
  « vy P / 'vy' STO « vy vx fun1 » 'fy' STO
    x0 V0 y0 pa fy abLmin "Lmin" □TAG »
»
```

Haciendo correr el programa se obtiene:

Lmin: 45.7227477652

Que es el flujo mínimo de líquido que puede ser empleado en la torre. Este resultado concuerda bien con el resultado obtenido por simple tanteo en el ejemplo 4.4.1. Como se dijo, no se puede emplear este flujo en un equipo real, porque para el mismo se requiere una torre de altura infinita, por lo tanto, el flujo mínimo real deberá ser siempre mayor al valor calculado.

En ocasiones, cuando el sistema se aleja mucho del comportamiento ideal, puede suceder que, debido a su curvatura de la línea de equilibrio, la línea de operación se haga tangente a la curva de equilibrio antes del punto x_{max}, y_0 . En estos casos no queda otra alternativa que encontrar el flujo mínimo por tanteo, como se hizo en el ejemplo 4.4.1.

4.4.3. Ejercicios

19. Se empleará una torre de absorción en contracorriente para absorber el 90% del metano contenido en 400 kgmol/h de una mezcla de metanol-aire, que se encuentra a una presión de 1 atm y una temperatura de 52°C. La presión parcial inicial de metanol en esta mezcla es de 110 mm de Hg. Calcule el número de etapas teóricas requeridas si se emplea un flujo de agua pura igual a: a) 1.1 veces el flujo mínimo; b) 1.5 veces el flujo mínimo; c) 2 veces el flujo mínimo. Los datos de equilibrio son los del ejemplo 4.2.1.2.
20. Se empleará una torre de absorción en contracorriente para reducir el porcentaje de CO₂ de una mezcla de CO₂ - aire que se encuentra a 1.4 atm y una temperatura de 0°C. El porcentaje inicial (molar) de CO₂ en la mezcla es del 12% y se quiere reducir el mismo a 0.9%. El flujo inicial del aire es de 250 kgmol/h. Calcule el número de etapas teóricas que deberá tener el equipo para lograr la absorción requerida si se emplea in flujo de agua pura igual a: a) 1.08 veces el flujo mínimo; b) 1.2 veces el flujo mínimo; c) 1.6 veces el flujo mínimo. Asuma que el sistema obedece la ley de Henry, siendo la constante de Henry 0.0728x10⁴ atm.

21. Se empleará una torre de absorción en contracorriente para absorber el 99.5% del amoniaco contenido en una mezcla de amoniaco - aire, que se encuentra a 0.9 atm y 293.2 K. El flujo inicial del gas es de 78 kgmol/h y tiene un porcentaje (molar) de amoniaco igual a 25%. Para la absorción se empleará agua pura a 293.2 K. Calcule el número de etapas teóricas que deberá tener el equipo para lograr la absorción requerida si se emplea in flujo de agua pura igual a: a) 1.1 veces el flujo mínimo; b) 1.5 veces el flujo mínimo; c) 4 veces el flujo mínimo. Los datos de equilibrio para este sistema son los del ejercicio 9.

4.5. Ecuaciones simplificadas para el cálculo del número de etapas

Aun cuando las ecuaciones que hemos empleado para el cálculo del número de etapas son simplificadas, pues se consideran constantes los flujos de gas y líquido inerte. Pueden ser simplificadas más aún si se asume que la curva de equilibrio es una línea recta, es decir se asume que el sistema obedece la ley de Henry.

Las ecuaciones resultantes para estas condiciones han sido deducidas por Kremser y para el caso de la absorción son las siguientes:

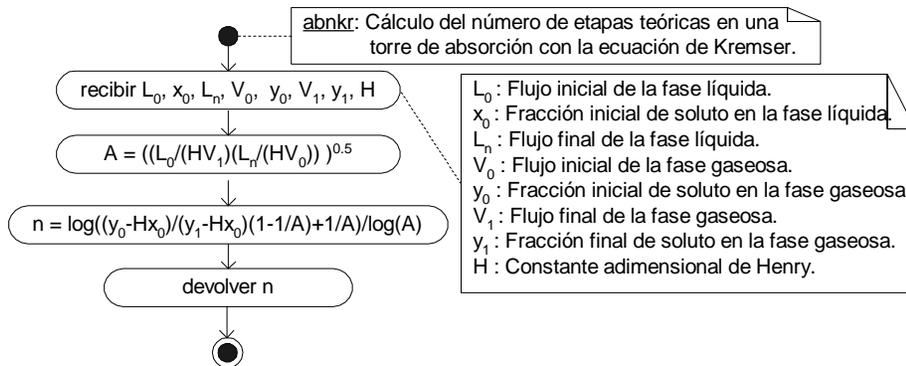
$$n = \frac{\log \left[\frac{y_0 - H'x_0 \left(1 - \frac{1}{A} \right) + \frac{1}{A}}{y_1 - H'x_0 \left(1 - \frac{1}{A} \right) + \frac{1}{A}} \right]}{\log(A)} \tag{32}$$

Donde "H'" es la constante adimensional de la ley de Henry y "A", es el factor de absorción, que está dada por la siguiente expresión:

$$A = \sqrt{\left(\frac{L_0}{H'V_1} \right) \left(\frac{L_n}{H'V_0} \right)} \tag{33}$$

Como se puede observar, para calcular el número de etapas con la ecuación de Kremser, se requiere resolver el sistema "A", pues se emplean los flujos y composiciones a la entrada y salida del equipo. Por lo tanto, salvo el hecho de que no es iterativa, esta ecuación no tiene mayor ventaja con relación al cálculo etapa por etapa (programa "abn").

Automatizaremos la solución de esta ecuación para los casos donde sea conveniente su uso. El algoritmo que resuelve la ecuación de Kremser es el siguiente:



Y el código elaborado en base al mismo es:

```
xNAME abnKr ( Número de etapas de absorción con la ecuación de Kremser )
::          ( datos: L0=9= Flujo inicial de líquido )
```

```

%8 CKNNOLASTWD DROP ( x0=8= Fracción inicial de soluto en el líquido)
CK&DISPATCH1 ( Ln=7= Flujo final de líquido )
# 11111 ( V0=6= Flujo inicial de gas )
:: ( y0=5= Fracción inicial de soluto en el gas )
 8ROLL 8ROLL 8ROLL ( V1=4= Flujo final de gas )
CK&DISPATCH1 ( y1=3= Fracción final de soluto en el gas )
# 111 ( H=2= Constante adimensional de Henry )
::
 8UNROLL 8UNROLL 8UNROLL ( A=1 )
%0
' NULLLAM NINE NDUPN DOBIND
9GETLAM 2GETLAM 4GETLAM %* %/ ( L0/[H*V1] )
7GETLAM 2GETLAM 6GETLAM %* %/ ( Ln/[H*V0] )
%* %SQRT 1PUTLAM ( A=[...*...]^0.5 )
5GETLAM 2GETLAM 8GETLAM %* %- ( y0-H*x0 )
3GETLAM 2GETLAM 8GETLAM %* %- ( y1-H*x0 )
%/ %1 1GETLAM %1/ %- %* ( [.../...][1-1/A] )
1GETLAM %1/ %+ %LOG ( log[...+1/A] )
1GETLAM %LOG %/ ( n= .../log[A] )
ABND
;
# 000
::
 8UNROLL 8UNROLL 8UNROLL
# 202 DO#EXIT
;
;
;

```

4.5.1. Ejemplo

Repita el ejemplo 4.3.3.1 empleando la ecuación de Kremser.

Solución

Los datos del mencionado ejemplo son los siguientes:

```

L0 = 90 kgmol/h
x0 = 0
V0 = 30 kgmol/h
y0 = 0.09
pa = 99
P = 7.2x104 Pa = 7.2E4_Pa 1_mmHg CONVERT 2 RND = 540.04_mmHg

```

Se cuenta además con los resultados el sistema "A", obtenidos con el programa "abnc1":

```

Ln = 92.673 kgmol/h
xn = 0.0288
V1 = 27.327 kgmol/h
y1 = 0.00099

```

Puesto que para aplicar la ecuación de Kremser se supone que el sistema obedece la ley de Henry, ajustamos los 3 primeros datos de equilibrio a una línea recta. Empleamos sólo los tres primeros datos porque sabemos que los sistemas obedecen la ley de Henry sólo cuando las concentraciones son diluidas.

La pendiente de la recta ajustada es la constante de Henry. El programa que realiza el ajuste es el siguiente:

```

« [ 0 0.0333 0.072 ]

```

```

[ 0 30 62.8 ] 540.04 / 4 RND
2 COL→ 'ΣDAT' STO
1 XCOL
2 YCOL
LINFIT
LR
DTAG 4 RND "H" →TAG
SWAP DROP
»

```

Haciendo correr el programa se obtiene:

H: 1.614

Que es la constante de *Henry*, resultante del ajuste, para el sistema acetona-agua.

Ahora se tienen todos los datos para calcular el número de etapas con la ecuación de *Kremser* (programa *abnKr*):

```

« 90 0 92.673 30 0.09 27.327
0.00099 1.614
abnKr
2 RND "n" →TAG
»

```

Haciendo correr el programa se obtiene:

n: 5.6

Que como vemos es ligeramente inferior al número de etapas obtenido en el ejemplo 4.3.3.1 con el programa "abn".

4.5.2. Ejercicios

22. Se empleará una torre de absorción, con múltiples etapas en contracorriente, para absorber el 99% del metanol contenido en una mezcla de metanol-aire que se encuentra a una presión de 1 atm y una temperatura de 49°C. La presión parcial inicial del metanol en esta mezcla es de 130 mm de Hg y para la absorción se emplearán 900 kgmol/h de agua pura. Empleando la ecuación de *Kremser* calcule el número de etapas teóricas para lograr la separación requerida. Utilice los datos de equilibrio del ejemplo 4.2.1.2.
23. Se empleará agua pura, en una torre de absorción, con múltiples etapas en contracorriente, para reducir el porcentaje de CO₂ de una mezcla de CO₂ - aire que se encuentra a 2.1 atm y una temperatura de 0°C. El porcentaje inicial (molar) de CO₂ en la mezcla es del 20% y se quiere reducir el mismo a 0.9%. El flujo inicial del aire es de 270 kgmol/h y el del agua 1.5 veces el flujo mínimo. Empleando la ecuación de *Kremser* calcule el número de etapas teóricas que deberá tener el equipo para lograr la absorción requerida. Asuma que el sistema obedece la ley de Henry, siendo la constante de Henry 0.0728x10⁴ atm.
24. Una industria cuenta con una torre de absorción equivalente a 5.1 etapas de equilibrio. Se quiere emplear esta torre para absorber amoníaco, con agua pura, de una mezcla de amoníaco - aire que se encuentra a 1.6x10⁵ Pa y 293.2 K. El flujo inicial de gas es de 100 kgmol/h y tiene un porcentaje (molar) de amoníaco igual al 25%. El flujo inicial de agua pura es de 120 kgmol/h. Empleando la ecuación de *Kremser* calcule el porcentaje de soluto que puede ser absorbido en esta torre. Utilice los datos de equilibrio del ejercicio 9.

Kremser ha deducido también una ecuación simplificada para el proceso contrario al de la absorción: la desorción. La forma de la ecuación de *Kremser* para la desorción es la siguiente:

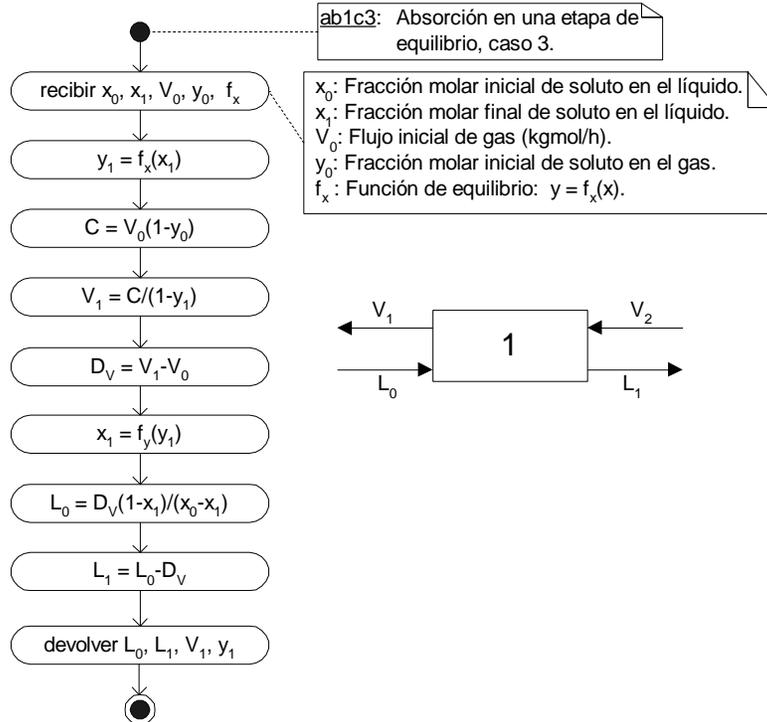
$$n = \frac{\log \left[\frac{x_0 - \frac{y_0}{H}}{x_n - \frac{y_0}{m}} (1-A) + A \right]}{\log \left(\frac{1}{A} \right)} \quad (34)$$

Que como vemos es similar a la ecuación (32), por lo que su solución puede ser automatizada siguiendo una lógica similar a la empleada en el programa "abnKr".

4.5.2.1. Absorción una etapa, caso 3.

Datos conocidos: Composición inicial y final de la fase líquida, flujo y composición inicial de la fase gaseosa.

El diagrama de actividades es el siguiente:



El código en System RPN es:

```

xNAME ab1c3 ( absorción en una etapa de equilibrio, caso 3 )
::
CK5&Dispatch ( Datos: x0,x1= Fracciones inicial y final en el líquido )
# 11118          ( V0,y0= Flujo y fracción inicial en el gas )
::              ( fx= Función de equilibrio y=f[x] )
%0 SIX NDUPN DROP
{ LAM x0 LAM x1 LAM V0 LAM y0 LAM fx
  LAM y1 LAM C LAM V1 LAM DV LAM L0
  LAM L1 }
BIND
LAM x1 LAM fx EVAL ' LAM y1 STO
LAM V0 %1 LAM y0 %- %* ' LAM C STO
LAM C %1 LAM y1 %- %/ ' LAM V1 STO
LAM V1 LAM V0 %- ' LAM DV STO
LAM DV %1 LAM x1 %- %* LAM x0 LAM x1 %- %/ ' LAM L0 STO
LAM L0 LAM DV %- ' LAM L1 STO
LAM L0 LAM L1 LAM V1 LAM y1
ABND
;
;
    
```

Y un programa de prueba, con los datos del ejemplo 4.2.1.1 es:

```

« 0. 1.84703223414E-3 200. .45
« 242.666666667 * »
ab1c3 4. →LIST { L0 L1 V1 y1 }
    
```

→TAG LIST→ DROP

»

Con el cual se obtienen los resultados:

L0: 350.000000391
 L1: 350.647657918
 V1: 199.352342473
 y1: .448213155485

Que prácticamente son los mismos valores del mencionado ejemplo.

Con este programa se puede resolver también el caso 1, así para los datos del ejemplo 4.2.1.1., resolvemos el problema con el siguiente programa:

```
« 350 0 200 0.45 « 242.666666667 * »
« 242.666666667 / » 0 0 0 0
□ L0 x0 V0 y0 fx fy LOC L1 V1 y1
« « 'x1' STO
  x0 x1 V0 y0 fx ab1c3 4 □LIST
  { LOC L1 V1 y1 } STO
  LOC L0 -
  »
  1E-8 y0 fy EVAL Refa DROP
  L1 x1 V1 y1 4 □LIST { L1 x1 V1 y1 } □TAG
  LIST□ DROP
  »
  »
```

Con lo que se obtienen los resultados:

L1: 350.647657375
 x1: 1.84703223415E-3
 V1: 199.352342474
 y1: .448213155488

Que son prácticamente los mismos resultados calculados en el mencionado ejemplo.

Este es un caso que al no ser iterativo permite resolver de manera iterativa prácticamente todos los otros casos siguiendo un procedimiento similar al presentado en el anterior ejemplo.

5. TORRES DE ABSORCIÓN RELLENAS

En el diseño de torres de absorción rellenas la principal dificultad radica en la determinación de los coeficientes de transferencia de masa. Es prácticamente imposible efectuar el cálculo teórico de dichos coeficientes debido al elevado número de variables que deben ser tomados en cuenta, tales como la superficie de contacto, forma, porosidad, distribución del relleno, etc.

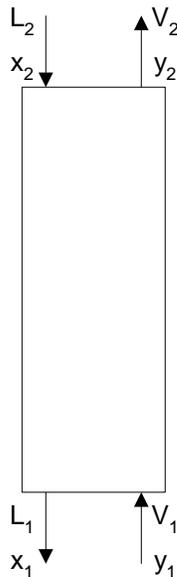
Es por ello que dichos coeficientes se calculan casi exclusivamente con ecuaciones empíricas, sin embargo dichas ecuaciones fácilmente pueden dar lugar a errores de hasta un 25%, dando lugar así a resultados inciertos.

La forma más confiable de calcular dichos coeficientes es mediante pruebas experimentales efectuadas en equipos de absorción, a escala de laboratorio, que operen en condiciones similares de temperatura, flujos superficiales y relación diámetro de relleno a diámetro de la torre que el equipo industrial.

En el experimento se mide la altura real de la torre así como los flujos y composiciones de las corrientes de entrada y salida. Entonces se asumen valores del coeficiente de difusión, calculando con ellos la altura de la columna hasta que la altura calculada sea igual a la altura medida. Cuando esto sucede se tiene el coeficiente de difusión que puede ser empleado para realizar diseños y/o simulaciones. Los coeficientes obtenidos de esta manera son mucho más confiables que los calculados con ecuaciones empíricas y son los únicos que se deben emplear para el dimensionamiento, diseño o simulación real de este tipo de equipos.

5.1. Flujos de entrada y salida en torres rellenas

Para los balances de materia consideraremos el esquema que se muestra en la figura:



Donde como se puede observar, todos los flujos y composiciones que entran o salen del fondo de la columna tienen el subíndice 1, mientras que los flujos y composiciones en la cabeza de la columna tienen el subíndice 2.

El balance de materia para el soluto es:

$$B\left(\frac{x_2}{1-x_2}\right) + C\left(\frac{y_1}{1-y_1}\right) = B\left(\frac{x_1}{1-x_1}\right) + C\left(\frac{y_2}{1-y_2}\right) \quad (1)$$

Donde como en las torres de absorción por etapas, B es el flujo molar de líquido inerte (constante) y C es el flujo molar de gas inerte (constante):

$$B = L_1(1-x_1) = L_2(1-x_2) = \text{ctte.} \quad (2)$$

$$C = V_1(1-y_1) = V_2(1-y_2) = \text{ctte.} \quad (3)$$

Los datos con los que generalmente contamos son los flujos y composiciones de entrada (L_2, x_2, V_1, y_1) y la composición de la fase gaseosa a la salida (y_2) o el porcentaje de soluto absorbido (pa). Por lo tanto sólo queda una incógnita, la misma que puede ser calculada con la ecuación (1).

Una situación un tanto más complicada se presenta cuando el dato adicional con el que contamos es la altura de la torre de relleno. En este caso tenemos que asumir un dato adicional, usualmente y_2 o pa , y con el valor asumido calcular la altura de la torre, entonces se compara la altura calculada con la altura conocida y si son iguales el proceso concluye, caso contrario se asumen otros valores de y_2 (o fa) hasta que la altura calculada sea igual a la altura conocida, de manera similar al procedimiento que se sigue en las torres de absorción por etapas cuando se conoce el número de etapas.

Analicemos el caso más frecuente donde se conocen los flujos y composiciones de entrada y el porcentaje de soluto absorbido. Con los flujos y composiciones de entrada y las ecuaciones (2) y (3) calculamos B y C , siendo el porcentaje absorbido:

$$pa = \left(\frac{V_1 y_1 - V_2 y_2}{V_1 y_1}\right) 100 \quad (4)$$

Como son conocidos todos los flujos de entrada y salida el lado izquierdo de la ecuación (1), es decir el flujo total de soluto, puede ser calculado:

$$A = B\left(\frac{x_2}{1-x_2}\right) + C\left(\frac{y_1}{1-y_1}\right) \quad (5)$$

Puesto que conocemos $V_1 y_1$, podemos despejar de esta ecuación $V_2 y_2$:

$$V_2 y_2 = V_1 y_1 \left(1 - \frac{pa}{100}\right) \quad (6)$$

Ahora con este valor y la ecuación (3) calculamos V_2 :

$$V_2 = C + V_2 y_2 \quad (7)$$

Entonces como ya conocemos V_2 y $V_2 y_2$, podemos calcular y_2 :

$$y_2 = \frac{V_2 y_2}{V_2} \quad (8)$$

Con y_2 calculado, de la ecuación (1) despejamos x_1 :

$$\frac{x_1}{1-x_1} = \frac{A - C\left(\frac{y_2}{1-y_2}\right)}{B} = D$$

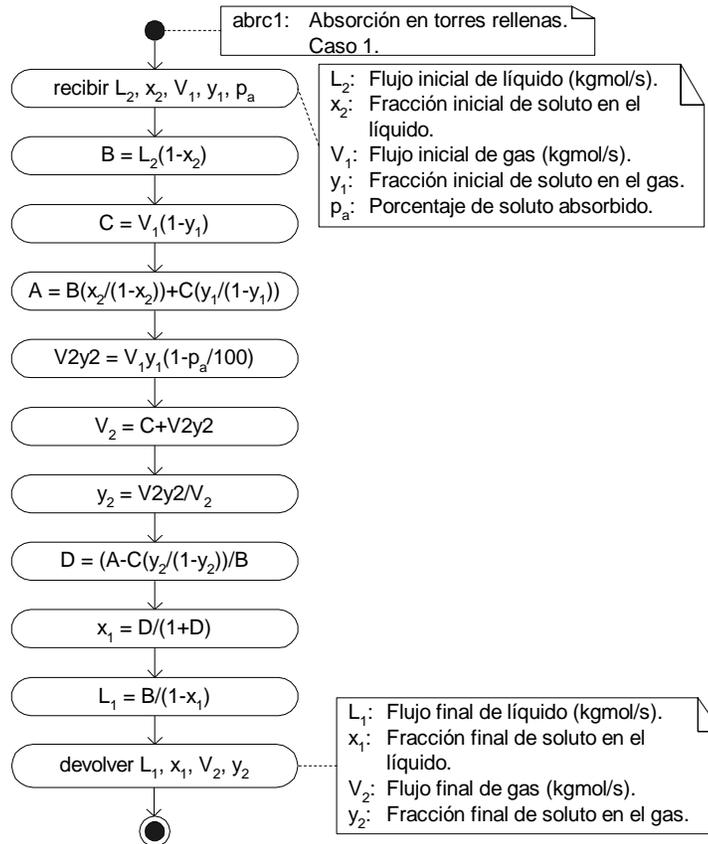
$$x_1 = \frac{D}{1+D} \tag{9}$$

Finalmente con x_1 y la ecuación (2) calculamos L_2 :

$$L_1 = \frac{B}{1-x_1} \tag{10}$$

Con lo que se tienen todos los flujos y composiciones de salida, quedando el sistema resuelto.

Aún cuando el procedimiento de cálculo descrito no es iterativo, es conveniente automatizarlo, pues como se dijo es el caso más frecuente. El diagrama de actividades para este fin es el siguiente:



El código elaborado en base al diagrama es:

```

xNAME abrc1 ( Torres de absorción rellenas, caso 1 )
::
    ( Datos: L2=14= Flujo inicial de líquido )
    CK5&Dispatch      ( x2=13= Fracción inicial de soluto en el líquido )
    # 11111           ( V1=12= Flujo inicial de gas )
    ::
    ( y1=11= Fracción inicial de soluto en el gas )
    %0 NINE NDUPN    ( pa=10= Porcentaje de soluto absorbido )
    DROP ` NULLLAM   ( Locales: A=9; B=8; C=7; D=6; V2y2=5; V2=4 )
    FOURTEEN NDUPN DOBIND ( y2=3; x1=2; L1=1 )
    14GETLAM %1 13GETLAM %- %* 8PUTLAM      ( B=L2*[1-x2] )
    12GETLAM %1 11GETLAM %- %* 7PUTLAM      ( C=V1*[1-y1] )
    8GETLAM 13GETLAM %1 13GETLAM %- %/ %* ( B*[x2/[1-x2] )
    7GETLAM 11GETLAM %1 11GETLAM %- %/ %* ( C*[y1/[1-y1] )
    %+ 9PUTLAM      ( A=...+... )
    12GETLAM 11GETLAM %* ( V1*y1 )
    
```

```

%1 10GETLAM %100 %/ %- %* 5PUTLAM      ( V2y2=...*[1-pa/100] )
7GETLAM 5GETLAM %+ 4PUTLAM              ( V2=C+V2y2 )
5GETLAM 4GETLAM %/ 3PUTLAM              ( y2=V2y2/V2 )
9GETLAM 7GETLAM 3GETLAM %1 3GETLAM
%- %/ %* %- 8GETLAM %/ 6PUTLAM          ( D=[A-C[y2/[1-y2]]/B )
6GETLAM 6GETLAM %1+ %/ 2PUTLAM          ( x1=D/[1+D] )
8GETLAM %1 2GETLAM %- %/ 1PUTLAM        ( L1=B/[1-x1] )
1GETLAM 2GETLAM 4GETLAM 3GETLAM ABND    ( L1 x1 V2 y2 )
;
;

```

5.1.1. Ejemplo

Se absorbe acetona con agua pura en una torre rellena. El aire de entrada contiene 4.0 mol% de acetona y el de salida 0.5 mol%. El flujo de gas es 15.02 kgmol/h. El flujo de entrada de agua pura es 50.5 kgmol de agua/h. Calcule los flujos y composiciones faltantes.

Solución:

Los datos con los que contamos son:

```

L2 = B = 50.5 kgmol/h = 50.5 3600 / 6 RND = 0.014028 kgmol/s
x2 = 0
V1 = 15.02 kgmol/h = 15.02 3600 / 6 RND = 0.004172 kgmol/s
y1 = 0.040
y2 = 0.005

```

Para resolver el problema con el programa *abrc1*, necesitamos el porcentaje absorbido (*pa*), el cual puede ser calculado con la ecuación (4). Antes es necesario calcular el flujo de gas inerte con la ecuación (3):

```

C = V1*(1-y1) = 0.004172 1 0.04 - * 6 RND = 0.004005 kgmol/s

```

Entonces, el flujo final de gas (ecuación 3) es:

```

V2 = C/(1-y2) = 0.004005 1 0.005 - / 6 RND = 0.004025 kgmol/s

```

Por lo tanto el porcentaje absorbido (ecuación 4) es:

```

pa = 0.004172 0.04 * 0.004025 0.005 * - 0.004172 0.04 * / 100 * 6 RND
= 87.940436

```

Ahora se tienen todos los datos para resolver el problema con el programa *abrc1*:

```

« 0.014028 0 0.004172 0.04 87.940436
abrc1
DROP 3 →LIST 6 RND
{ L1 x1 V2 } →TAG
LIST→ DROP
»

```

Haciendo correr el programa se obtiene:

```

L1: .014175
x1: .010353
V2: .004025

```

Que son los flujos y composiciones desconocidas.

5.1.2. Ejercicios

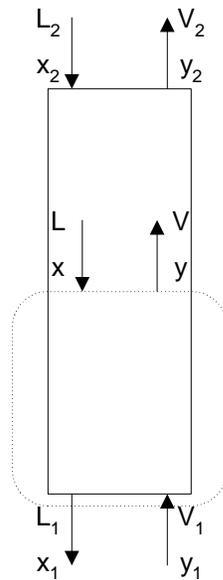
1. Se desea absorber el 70% del metanol contenido en 450 kgmol/h de una mezcla de metanol-aire que se encuentra a una presión de 740 mm de Hg

y una temperatura de 40°C. La presión parcial del metano en esta mezcla es de 100 mm de Hg y para la absorción se emplearán 970 kgmol/h de agua pura. Calcule los flujos y composiciones faltantes.

2. Se empleará una torre de absorción rellena para absorber el 96% del amoniaco contenido en 65 kgmol/h de una mezcla de amoniaco - aire, que contiene 14.3 mol% de amoniaco. La presión del gas es de 1.723×10^5 Pa y para la absorción se emplea agua pura a 293.2 K. Calcule los flujos y composiciones faltantes.
3. Se empleará una torre de absorción rellena para absorber SO_2 de una mezcla gaseosa de SO_2 -Aire que se encuentra a 2.3×10^5 Pa y 30°C. Para la absorción se emplearán 400 kgmol/h de agua pura. El flujo inicial del gas es de 10 kgmol/h y contiene 6 mol% de SO_2 , se desea reducir este porcentaje a 0.4 mol%. Calcule los flujos y composiciones faltantes.

5.2. Línea de operación

Una vez resuelto el sistema general tomamos un segmento de la columna, como se muestra en la figura y efectuamos los balances de materia en el mismo:



El balance de materia para el soluto en este segmento es:

$$B \left(\frac{x}{1-x} \right) + C \left(\frac{y_1}{1-y_1} \right) = B \left(\frac{x_1}{1-x_1} \right) + C \left(\frac{y}{1-y} \right) \quad (11)$$

Al haber resuelto el sistema general conocemos B , C , y_1 , x_1 . Reordenamos entonces la ecuación (11) de manera que los términos conocidos queden en el mismo lado de la ecuación y puedan ser calculados como una constante:

$$B \left(\frac{x}{1-x} \right) - C \left(\frac{y}{1-y} \right) = B \left(\frac{x_1}{1-x_1} \right) - C \left(\frac{y_1}{1-y_1} \right) = D_1 \quad (11.a)$$

Esta ecuación constituye la línea de operación. Con esta ecuación podemos calcular el valor de "x" si conocemos el valor de "y" o viceversa. Así si conocemos el valor de "x", el valor de "y" puede ser despejado de la ecuación (11.a) de la siguiente manera:

$$\frac{y}{1-y} = \frac{B\left(\frac{x}{1-x}\right) - D_1}{C} = D_2$$

$$y = \frac{D_2}{1+D_2} \quad (12)$$

De manera similar, si conocemos el valor de "y" podemos despejar el valor de "x" de la ecuación (11.a) de la siguiente manera:

$$\frac{x}{1-x} = \frac{C\left(\frac{y}{1-y}\right) + D_1}{B} = D_3$$

$$x = \frac{D_3}{1+D_3} \quad (13)$$

Entonces con la ecuación (12) o la ecuación (13) podemos construir la línea de operación en una gráfica "x" versus "y" dándonos diferentes valores de "x" o de "y".

Cuando el problema se resuelve numéricamente estas ecuaciones nos permiten calcular los valores de "x" para valores de "y" conocidos (o viceversa).

5.3. Coeficientes de transferencia de masa

En el cálculo de la altura de una torre rellena uno de los datos con los que se debe contar son los coeficientes de transferencia de masa para las fases gaseosa y líquida.

En general y como ya se dijo, para calcular la altura de una torre de absorción rellena se deben emplear coeficientes medidos experimentalmente y no calculados analíticamente, pues los valores calculados tienen un margen de error inaceptable (superior al 20%), por lo que no deben ser empleados en un caso real.

5.3.1. Coeficientes individuales ideales

Normalmente los coeficientes de transferencia de masa son calculados a partir de los coeficientes individuales ideales, debido a que dichos coeficientes son más fáciles de medir experimentalmente (o calcular analíticamente).

Los coeficientes de transferencia de masa individuales ideales corresponden al proceso de contradifusión equimolar. Proceso ideal que ocurre en soluciones muy diluidas, donde por cada molécula que se difunde en un sentido existe otra que se difunde en el sentido contrario. Denominaremos a estos coeficientes k'_x (para la fase líquida) y k'_y (para la fase gaseosa), entonces se cumple que:

$$N_{Ay} = k'_y (y - y_i) = N_{Ax} = k'_x (x_i - x) \quad (14)$$

Donde N_A es el flujo molar de soluto (en kgmol/s.m²), k'_y y k'_x son los coeficientes individuales ideales para las fases gaseosa y líquida (en kgmol/s.m²), "y" y "x" son las fracciones molares de soluto en las fases gaseosa y líquida y "y_i" y "x_i" son las fracciones molares de soluto en la interfaz gas - líquido para las fases gaseosa y líquida. Se asume que en la interfaz gas - líquido las dos fases están en equilibrio, de manera que conociendo una de ellas es posible calcular la otra, es decir se cumple que:

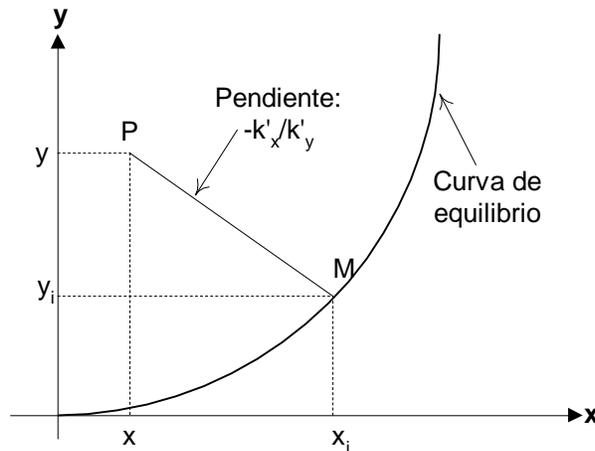
$$y_i = f_x(x_i) \quad (15)$$

o

$$x_i = f_y(y_i) \quad (16)$$

Donde f_x y f_y son las funciones de equilibrio, las mismas que como vimos en el capítulo anterior pueden estar en forma tabular, gráfica o en forma una ecuación empírica.

Con los coeficientes individuales ideales es posible calcular la composición de la interfaz gas-líquido siguiendo el procedimiento gráfico que se describe a continuación y que se muestra en la siguiente figura:



- En una gráfica "x" versus "y" (fracción molar de soluto en la fase líquida versus fracción molar de soluto en la fase gaseosa), se traza la curva de equilibrio empleando los datos de equilibrio o la ecuación de equilibrio.
- Se ubica el punto "P" que corresponde a las composiciones reales de las fases gaseosa y líquida (x, y).
- Partiendo del punto "P" se traza una línea recta con pendiente $-k'_x/k'_y$ hasta intersectar la curva de equilibrio, ubicando así el punto "M".
- En el punto "M" se leen las composiciones de la interfaz gas - líquido (x_i, y_i), que son los valores buscados.

Como veremos luego, los valores de " x_i ", " y_i " son importantes en el cálculo de la altura de la torre.

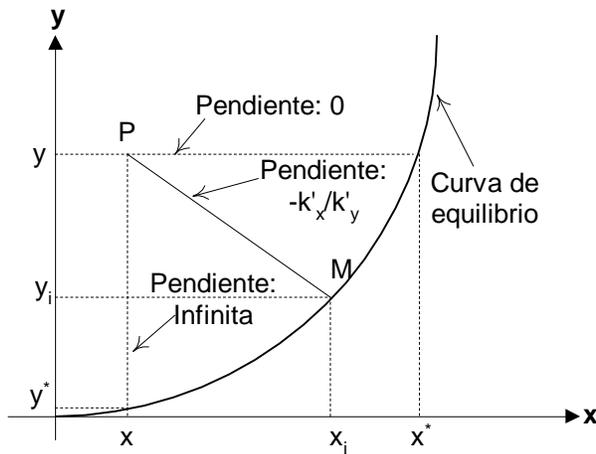
Matemáticamente el problema consiste en encontrar la intersección entre una línea recta (de pendiente $-k'_x/k'_y$) y una curva (la curva de equilibrio). Este problema puede ser resuelto numéricamente igualando la ecuación (14) a cero, empleando una de las ecuaciones de equilibrio (15 o 16) y un método de iterativo como el método de *Newton - Raphson* o *Regula Falsi*.

La función a resolver es:

$$f(x_i) = \frac{k'_x}{k'_y} + \frac{y - y_i}{x - x_i} = 0 \quad (17)$$

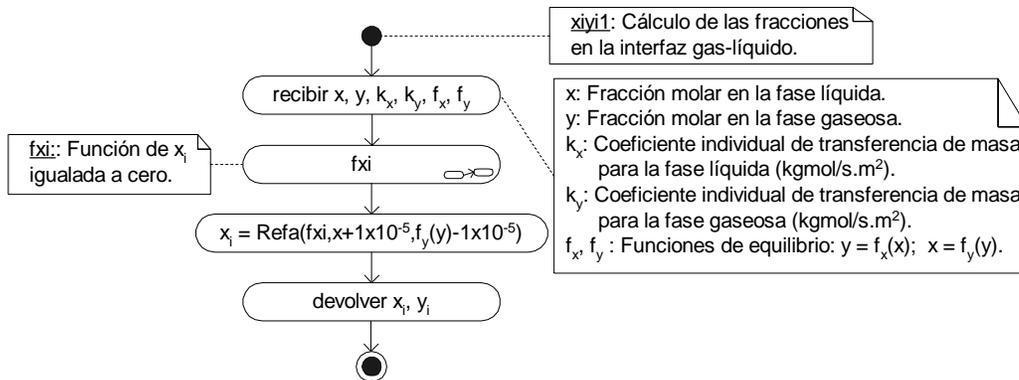
$$y_i = f_x(x_i)$$

En esta función podemos determinar previamente los límites entre los cuales se encuentra la solución (x_i), tomando como base la figura que se muestra a continuación.

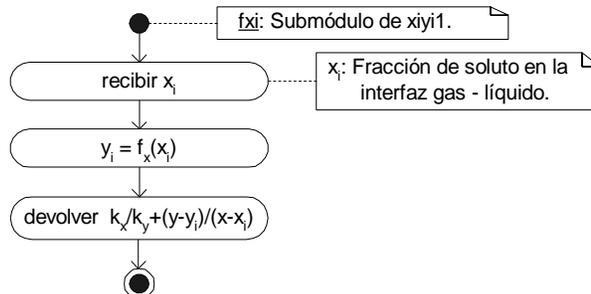


Como se puede ver en esta figura, el valor mínimo se obtiene cuando la pendiente es infinita, situación para la cual el mínimo es igual a "x" (la fracción de soluto en la fase líquida). El máximo se obtiene cuando la pendiente es cero y como se puede observar corresponde al valor de "x" en equilibrio con "y", es decir $x^*=f_y(y)$. Observe también que y^* es el valor de "y" en equilibrio con "x", es decir $y^*=f_x(x)$.

Dado que los límites entre los cuales se encuentra la solución son conocidos (o calculables) es conveniente emplear un método iterativo acotado como el de *Regula - Falsi*, pues la convergencia está prácticamente asegurada. El algoritmo del módulo que resuelve el problema es el siguiente:



Donde el submódulo "fxi" tiene el siguiente algoritmo:



Como se puede observar en este algoritmo no se toman directamente los límites mínimo y máximo, sino que al límite mínimo se le suma 1×10^{-5} y al máximo se le resta 1×10^{-5} para evitar que se produzca una división entre cero.

El programa elaborado en base a estos algoritmos (y que debe ser añadido al archivo Absorción.s) es el siguiente:

```
xNAME xiy1 ( Composición en la interfaz gas - líquido )
::      ( Datos: x= Fracción de soluto en la fase líquida )
%6 CKNNOLASTWD DROP      ( y= Fracción de soluto en la fase gaseosa )
CK&DISPATCH1            ( kx= Coeficiente individual, fase líquida )
# 11188                  ( ky= Coeficiente individual, fase gaseosa )
::      ( fx= Función de equilibrio, y=fx[x] )
6ROLL                    ( fy= Función de equilibrio, x=fy[y] )
CK&DISPATCH1            ( Locales: xi; yi )
ONE
::
  6UNROLL
  %0 %0
  { LAM x LAM y LAM kx LAM ky
    LAM fx LAM fy LAM xi LAM yi }
  BIND
  '                                ( función f[xi] )
  ::
    ' LAM xi STO
    LAM xi LAM fx EVAL ' LAM yi STO
    LAM kx LAM ky %/ LAM y LAM yi %-
    LAM x LAM xi %- %/ %+
  ;
  LAM x % 1E-5 %+           ( límite inferior )
  LAM y LAM fy EVAL % 1E-5 %- ( límite superior )
  xRefa DROP
  LAM xi LAM yi
  ABND
;
ZERO
::
  6UNROLL # 202 DO#EXIT      ( 202 = Bad Argument Type )
;
;
;
```

5.3.1.1. Ejemplo

Un soluto "A" es absorbido de una mezcla gaseosa de "A" y "B", en una torre de paredes mojadas, por el líquido que fluye hacia abajo en forma de película. En un punto de la torre la concentración de soluto en el gas es 0.380 en fracción molar y la concentración de soluto en el líquido es 0.100. La torre opera a 298 K y 1.013×10^5 Pa y los datos de equilibrio son:

x	0	0.05	0.10	0.15	0.20	0.25	0.30	0.35
y	0	0.022	0.052	0.087	0.131	0.187	0.265	0.385

El soluto "A" se difunde a través de "B" en reposo en la fase gaseosa y después, a través de un líquido que no se difunde. Mediante correlaciones para soluciones diluidas en torres de paredes mojadas, se ha determinado que el coeficiente de película de transferencia de masa para "A" en la fase gaseosa es 1.465×10^{-3} kgmol/s.m² (1.08 lbmol/h.pie²) y para la fase líquida 1.967×10^{-3} kgmol/s.m² (1.45 mol/lb/h.pie²). Calcule las concentraciones en la interfaz (y_i y x_i) y el flujo específico (N_A).

Solución:

Los datos con los que contamos son los siguientes:

```
x = 0.1
y = 0.38
k'x = 1.967x10-3 kgmol/s.m2
k'y = 1.465x10-3 kgmol/s.m2
```

Se cuenta también con los datos de equilibrio. Por lo tanto tenemos todos los datos para resolver el problema con el programa *xiyil*.

Como los datos de equilibrio están en forma tabular, los almacenamos por comodidad en variables globales para crear las funciones arbitrarias correspondientes:

```
[ 0 0.05 0.10 0.15 0.20 0.25 0.30 0.35 ] 'vx' STO
[ 0 0.022 0.052 0.087 0.131 0.187 0.265 0.385 ] 'vy' STO
```

Entonces resolvemos el problema con el siguiente programa:

```
« 0.1 0.38 1.967E-3 1.465E-3
« vx vy fun1 »
« vy vx fun1 »
0 0
□ x y kx ky fx fy xi yi
« x y kx ky fx fy
xiyil
'yi' STO 'xi' STO
xi "xi" □TAG
yi "yi" □TAG
kx xi x - * "Nax" □TAG
ky y yi - * "Nay" □TAG
»
»
```

Haciendo correr el programa se obtiene:

```
xi: .246589334222
yi: .183180054329
Nax: 2.88341220415E-4
Nay: 2.88341220408E-4
```

Que son las concentraciones en la interfaz y los flujos específicos de soluto. Estos flujos deberían ser iguales, la pequeña diferencia que existe se debe únicamente a errores de redondeo.

5.3.1.2. Ejercicios

4. Resuelva el ejemplo empleando los mismos datos, pero considerando que las fracciones de soluto son: $x = 0.05$, $y = 0.25$.
5. Resuelva el ejemplo empleando los mismos datos, pero considerando que las fracciones de soluto son: $x = 0.2$; $y = 0.5$.

5.3.2. Coeficientes individuales reales

Si en lugar de los coeficientes individuales ideales se tienen como datos los coeficientes individuales reales (correspondientes al proceso de difusión a través de un gas estancado) el procedimiento de cálculo para determinar la composición de la interfaz gas - líquido, es el mismo que el de los coeficientes individuales ideales, por supuesto que ahora se deben trabajar

con los coeficientes reales, a los cuales denominaremos k_x y k_y , en lugar de los coeficientes individuales (denominados k'_x y k'_y).

5.3.2.1. Ejemplo

Como ejemplo volveremos a resolver el ejemplo 5.3.1.1 pero empleando los coeficientes individuales reales: $k_x = 0.0024$ y $k_y = 0.00207$.

Solución:

El problema se resuelve esencialmente con el mismo programa elaborado para el mencionado ejercicio. La única diferencia es que ahora debemos emplear los coeficientes individuales reales en lugar de los ideales:

```
« 0.1 0.38 0.0024 0.00207
« vx vy fun1 »
« vy vx fun1 »
0 0
→ x y kx ky fx fy xi yi
« x y kx ky fx fy
xiyi1
'yi' STO 'xi' STO
xi "xi" →TAG
yi "yi" →TAG
kx xi x - * "Nax" →TAG
ky y yi - * "Nay" →TAG
»
»
```

Haciendo correr el programa se obtiene:

```
xi: .257018759326
yi: .197949264549
Nax: 3.76845022382E-4
Nay: 3.76845022384E-4
```

Que son las concentraciones en la interfaz y los flujos específicos de soluto.

5.3.2.2. Ejercicios

- Empleando los datos del ejemplo 5.3.1.1 calcule la composición de la interfaz gas - líquido (x_i , y_i) y los flujos específicos de soluto (N_{Ax} y N_{Ay}) para una composición de las fases líquida y gaseosa igual a: $x = 0.05$, $y = 0.25$. Los coeficientes globales reales son: $k_x = 0.0024$ y $k_y = 0.00207$.
- Repita el anterior ejercicio para una composición igual a: $x = 0.15$, $y = 0.43$.

5.3.3. Cálculo de los coeficientes individuales reales a partir de los coeficientes individuales ideales

Como ya se mencionó, la situación más frecuente no es aquella en la cual se conocen los coeficientes individuales reales, sino aquella en la que se conocen los coeficientes individuales ideales y se requiere efectuar el cálculo para una situación real.

Afortunadamente podemos calcular los coeficientes reales a partir de los coeficientes individuales, pues dichos coeficientes están relacionados entre sí por las siguientes expresiones:

$$k_x = \frac{k'_x}{(1-x)_{im}}; \quad k_y = \frac{k'_y}{(1-y)_{im}} \quad (18)$$

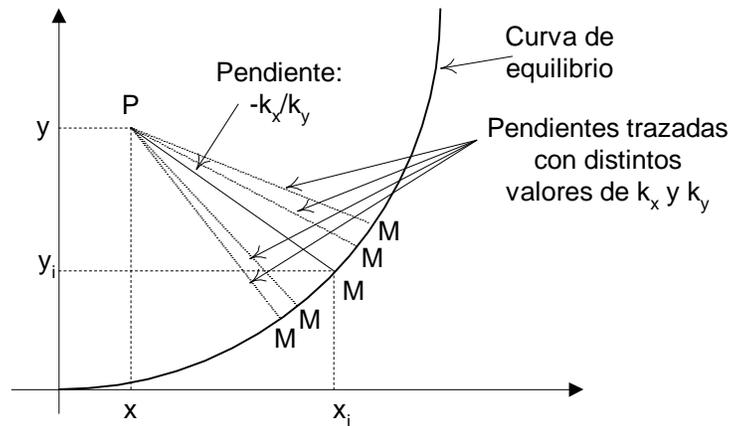
Donde:

$$(1-x)_{im} = \frac{(1-x) - (1-x_i)}{\ln\left(\frac{1-x}{1-x_i}\right)}; \quad (1-y)_{im} = \frac{(1-y_i) - (1-y)}{\ln\left(\frac{1-y_i}{1-y}\right)} \quad (19)$$

Siendo el flujo molar de soluto:

$$N_{Ay} = k_y(y - y_i) = N_{Ax} = k_x(x_i - x) \quad (20)$$

El procedimiento de cálculo, tanto gráfico como numérico es iterativo. Gráficamente la solución se encuentra de acuerdo a los siguientes pasos (un esquema de los mismos se presenta en la figura).



- a) En una gráfica "x" versus "y" (fracción molar de soluto en la fase líquida versus fracción molar de soluto en la fase gaseosa), se traza la curva de equilibrio empleando los datos de equilibrio o la ecuación de equilibrio.
- b) Se ubica el punto "P" que corresponde a la composición de las fases gaseosa y líquida (x,y).
- c) Inicialmente se asume que k_x y k_y son iguales a k'_x y k'_y respectivamente.
- d) Con los valores de k_x y k_y asumidos se traza una línea recta con pendiente -k_x/k_y, la cual parte del punto "P" e intersecta la curva de equilibrio en el punto "M".
- e) En el punto "M" se leen las composiciones de la interfaz gas - líquido (x_i, y_i). Con los valores leídos y las ecuaciones (18) y (19) se calculan nuevos valores de k_x y k_y.
- f) Se comparan los valores calculado con los asumidos. Si son aproximadamente iguales el proceso concluye, habiéndose determinado los valores de k_x, k_y, x_i y y_i. Si no son iguales los valores calculados se convierten en valores asumidos y se repite el proceso desde el inciso (d).

Matemáticamente el problema consiste en encontrar la intersección entre una línea recta y una curva (la curva de equilibrio) de manera tal que los

valores de k_x y k_y empleados en la pendiente de la línea recta, sean los valores calculados con las fracciones x_i y y_i de la intersección.

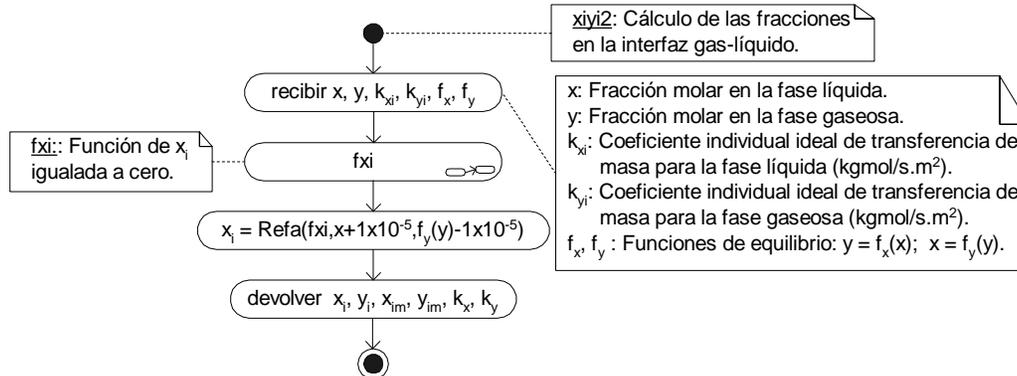
Una vez más se trata de un problema iterativo que puede ser expresado en función de una sola variable (x_i) y resuelto con uno de los métodos existentes para este tipo de ecuaciones, tal como el método de *Regula-Falsi* o la *Secante*.

La función a resolver, obtenida igualando la ecuación (20) a cero es la siguiente:

$$\begin{aligned}
 f(x_i) &= \frac{k_x}{k_y} + \frac{y - y_i}{x - x_i} = 0 \\
 y_i &= f_x(x_i) \\
 k_x &= \frac{k'_x}{(1-x)_{im}} \\
 k_y &= \frac{k'_y}{(1-y)_{im}} \\
 (1-x)_{im} &= \frac{(1-x) - (1-x_i)}{\ln\left(\frac{1-x}{1-x_i}\right)} \\
 (1-y)_{im} &= \frac{(1-y_i) - (1-y)}{\ln\left(\frac{1-y_i}{1-y}\right)}
 \end{aligned}
 \tag{21}$$

Al igual que en el cálculo de la composición en la interfaz gas-líquido (programa *xiyi1*) se resolverá el problema recurriendo al método de *Regula-Falsi*, calculando los límites del segmento de búsqueda de la misma forma.

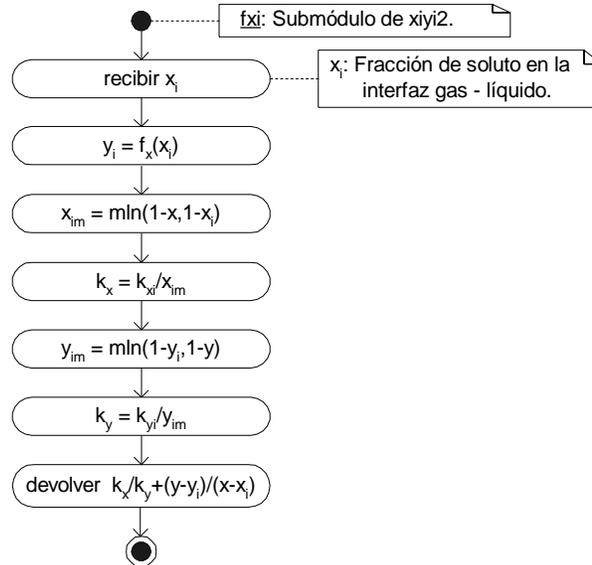
El algoritmo del módulo que resuelve la ecuación (21) es el siguiente:



Donde el algoritmo del submódulo *fxi* se presenta en el diagrama de actividades de la siguiente página. El código elaborado en base a estos algoritmos es el siguiente:

```

xNAME xiyi2 ( Composición en la interfaz gas - líquido )
::      ( Datos: x= Fracción de soluto en la fase líquida )
      %6 CKNNOLASTWD DROP ( y= Fracción de soluto en la fase gaseosa )
      CK&DISPATCH1      ( k_xi= Coeficiente individual ideal, fase líquida )
      # 11188            ( k_yi= Coeficiente individual ideal, fase gaseosa )
      ::                ( f_x= Función de equilibrio, y=f_x[x] )
      6ROLL             ( f_y= Función de equilibrio, x=f_y[y] )
    
```



```

CK&DISPATCH1      ( Locales: xi; yi; xim; yim; kx; ky )
ONE
::
6UNROLL
%0 SIX NDUPN DROP
{ LAM x LAM y LAM kxi LAM kyi
  LAM fx LAM fy LAM xi LAM yi
  LAM xim LAM yim LAM kx LAM ky }
BIND
'
'                                     ( función f[xi] )
::
' LAM xi STO
LAM xi LAM fx EVAL ' LAM yi STO
%1 LAM x %- %1 LAM xi %- mln ' LAM xim STO
LAM kxi LAM xim %/ ' LAM kx STO
%1 LAM y %- %1 LAM yi %- mln ' LAM yim STO
LAM kyi LAM yim %/ ' LAM ky STO
LAM kx LAM ky %/ LAM y LAM yi %-
LAM x LAM xi %- %/ %+
;
LAM x % 1E-5 %+                               ( límite inferior )
LAM y LAM fy EVAL % 1E-5 %-                   ( límite superior )
xRefa DROP
LAM xi LAM yi LAM xim LAM yim LAM kx LAM ky
ABND
;
ZERO
::
6UNROLL # 202 DO#EXIT                          ( 202 = Bad Argument Type )
;
;
;

```

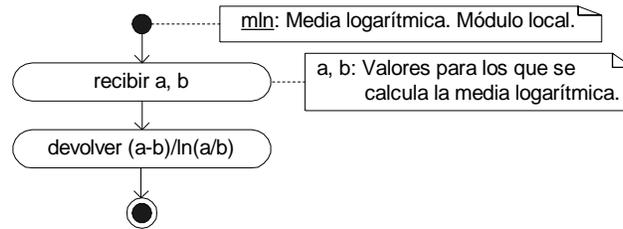
La función *mln* a la que se llama desde el submódulo *fxi* calcula la media logarítmica de dos valores. La lógica de este módulo se presenta en el diagrama de actividades de la siguiente página y el código respectivo es:

```

NULLNAME mln ( Media logarítmica de dos números dados )
:: ( Datos en la pila: a, b )

```

```
2DUP %- UNROT %/ %LN %/
;
```



Como se indica en este diagrama "mLn" es un módulo local, por lo tanto se implementa como un módulo sin nombre: *NULLNAME*. Al ser un módulo sin nombre, sólo puede ser llamado desde otro módulo del proyecto pero no desde el emulador (o calculadora). En este módulo no se comprueba ni el número de datos ni su tipo, por lo que esa tarea (al igual que sucede con la mayoría de las funciones de *System RPN*) debe ser llevada a cabo por el programador.

5.3.3.1. Ejemplo

Como ejemplo volveremos a resolver el ejemplo 5.3.1.1 pero calculando los coeficientes individuales reales, es decir considerando que el proceso es de difusión a través de un gas estancado.

Solución

El problema se resuelve prácticamente de la misma forma que en el mencionado ejemplo, sólo que ahora se llama a "xiyi2" en lugar de "xiyi1". El programa modificado para este ejemplo es el siguiente:

```
<< 0.1 0.38 1.967E-3 1.465E-3
<< vx vy fun1 >>
<< vy vx fun1 >>
0 0 0 0 0 0
□ x y kxi kyi fx fy xi yi xim yim kx ky
<< x y kxi kyi fx fy
xiyi2
6 □LIST DUP
{ xi yi xim yim kx ky } STO
{ xi yi xim yim kx ky } □TAG LIST□ DROP
kx xi x - * "Nax" □TAG
ky y yi - * "Nay" □TAG
>>
>>
```

Antes de hacer correr el programa debe verificar si los datos de equilibrio están guardados en las variables globales "vx" y "vy". Haciendo correr el programa se obtienen los siguientes resultados:

```
xi: .257373689149
yi: .198502955072
xim: .818794080564
yim: .706869323732
kx: 2.40231341028E-3
ky: 2.07251885294E-3
Nax: 3.78060923868E-4
Nay: 3.76156047366E-4
```

Que son las composiciones en la interfaz, las medias logarítmicas, las constantes reales y los flujos de soluto respectivos.

5.3.3.2. Ejercicios

8. Vuelva a resolver el ejemplo 5.3.3.1 pero para una composición igual a: $x=0.25$, $y=0.05$.
9. Se absorberá SO_2 de azufre de una mezcla gaseosa de aire - SO_2 , empleando agua pura. En un punto de la torre la fracción molar de SO_2 en el aire es 0.20 y la fracción molar en el líquido 0.00355. La torre opera a 293 K y 1.013×10^5 Pa, condiciones para las cuales los datos de equilibrio son los siguientes:

x_A	P_A (mm Hg)
0.0000000	0.0
0.0000562	0.5
0.0001403	1.2
0.0002800	3.2
0.0004220	5.8
0.0005640	8.5
0.0008420	14.1
0.0014030	26.0
0.0019650	39.0
0.0027900	59.0
0.0042000	92.0
0.0069800	161.0
0.0138500	336.0
0.0206000	517.0
0.0273000	698.0

El SO_2 se difunde a través del aire en reposo en la fase gaseosa y después, a través del agua que no se difunde. Mediante correlaciones empíricas se ha encontrado que los coeficientes individuales ideales son: $k'_y = 0.03398$ kgmol/s.m² y $k'_x = 0.848$ kgmol/s.m². Calcule las concentraciones en la interfaz gas líquido x_i , y_i , las medias logarítmicas $(1-x)_{im}$, $(1-y)_{im}$, los coeficientes individuales reales k_x , k_y y los flujos específicos de soluto N_{Ax} , N_{Ay} .

5.3.4. Coeficientes globales ideales de transferencia de masa

En la práctica resulta más fácil medir los coeficientes globales de transferencia de masa que los coeficientes individuales, por esta razón en la mayoría de los casos prácticos se dispone o se pueden calcular estos coeficientes y no los coeficientes individuales.

Los coeficientes globales se basan en una de las fases: la fase líquida K'_x o la fase gaseosa K'_y , de manera análoga al procedimiento que se sigue en transferencia de calor, donde los coeficientes globales se basan en una de las superficies: interna o externa.

El coeficiente global de transferencia de masa basado en la fase líquida se define como:

$$N_A = K'_x(x^* - x) \quad (22)$$

Donde x^* es la fracción molar de soluto en equilibrio con "y", es decir:

$$x^* = f_y(y) \quad (23)$$

De manera similar el coeficiente global basado en la fase gaseosa se define como:

$$N_A = K'_y(y - y^*) \quad (24)$$

Donde y^* es la fracción molar de soluto en equilibrio con "x", es decir:

$$y^* = f_x(x) \quad (25)$$

Estos coeficientes se relacionan con los coeficientes individuales ideales a través de las expresiones que se presentan a continuación.

Para el coeficiente basado en la fase líquida la relación es:

$$\frac{1}{K'_x} = \frac{1}{m''k'_y} + \frac{1}{k'_x} \quad (26)$$

$$m'' = \frac{y - y_i}{x^* - x_i}$$

Donde el primer término del lado derecho representa la resistencia de la fase gaseosa y el segundo la resistencia de la fase líquida.

Para el coeficiente basado en la fase gaseosa tenemos:

$$\frac{1}{K'_y} = \frac{1}{k'_y} + \frac{m'}{k'_x} \quad (27)$$

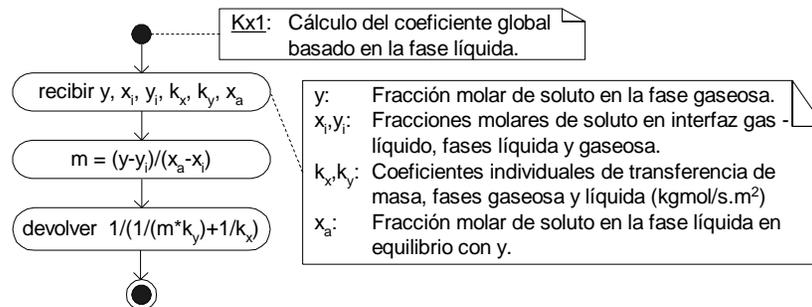
$$m' = \frac{y_i - y^*}{x_i - x}$$

Igual que el caso anterior, el primer término del lado derecho es la resistencia de la fase gaseosa y el segundo la resistencia de la fase líquida.

Por lo tanto si se tienen los coeficientes individuales se pueden calcular los coeficientes globales respectivos. Los valores de x_i y y_i que se requieren en estas ecuaciones pueden ser calculados con el programa "xiyi1" y los valores de x^* y y^* con las ecuaciones (23) y (25).

Aun cuando no es muy frecuente el contar con los coeficientes individuales, pues como se dijo antes es más sencillo medir los coeficientes globales, es conveniente contar con programas que automaticen el cálculo de estas dos ecuaciones.

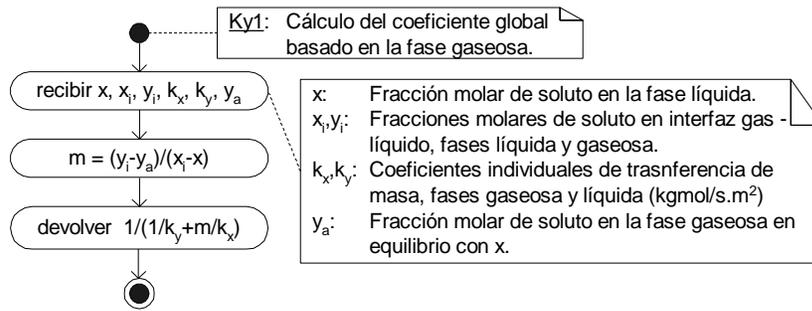
El algoritmo para la primera ecuación es:



El algoritmo para la segunda ecuación se presenta en la siguiente página. El código elaborado en base a estos algoritmos es el siguiente:

```

xNAME Kx1 ( Coeficiente global basado en la fase líquida )
:: ( Datos en la pila: y, xi, yi, kx, ky, ya )
%6 CKNNOLASTWD DROP
CK&DISPATCH1
# 11111
::
  
```



```

6ROLL
CK&DISPATCH1
# 1
::
5ROLL %- SWAP 5ROLL %- %/ ( m=[y-yi]/[xa-xi] )
%* %1/ SWAP %1/ %+ %1/ ( 1/[1/[m*ky]+1/kx] )
;
# 0
::
6UNROLL # 202 DO#EXIT ( 202 = Bad argument Type)
;
;
;

xNAME Ky1 ( Coeficiente global basado en la fase líquida )
:: ( Datos en la pila: x, xi, yi, kx, ky, ya )
%6 CKNNOLASTWD DROP
CK&DISPATCH1
# 11111
::
6ROLL
CK&DISPATCH1
# 1
::
6UNROLL
4ROLL SWAP %- 4ROLL 5ROLL %- %/ ( m=[yi-ya]/[xi-x] )
ROT %/ SWAP %1/ %+ %1/ ( 1/[1/ky+m/kx] )
;
# 0
::
6UNROLL # 202 DO#EXIT ( 202 = Bad argument Type)
;
;
;
;

```

5.3.4.1. Ejemplo

Usando los datos del ejemplo 5.3.1.1, calcule los coeficientes globales basados en la fase líquida (K'_x) y gaseosa (K'_y) y los flujos molares respectivos (N_{Ax} , N_{Ay}).

Solución:

Los datos del mencionado ejemplo son:

x = 0.1
y = 0.38

```

k'x = 1.967x10-3 kgmol/s.m2
k'y = 1.465x10-3 kgmol/s.m2
[ 0 0.05 0.10 0.15 0.20 0.25 0.30 0.35 ] 'vx' STO
[ 0 0.022 0.052 0.087 0.131 0.187 0.265 0.385 ] 'vy' STO

```

Se tiene entonces todos los datos para resolver el problema. El programa elaborado para con ese fin es el siguiente:

```

« 0.1 0.38 1.967E-3 1.465E-3
  « vx vy fun1 »
  « vy vx fun1 »
  0 0 0 0 0 0
  → x y kx ky fx fy xi yi
  xa ya Kx Ky
  « x y kx ky fx fy
    xiyil
    2 →LIST
    { xi yi } STO
    y fy EVAL 'xa' STO
    x fx EVAL 'ya' STO
    y xi yi kx ky xa Kx1 'Kx' STO
    x xi yi kx ky ya Ky1 'Ky' STO
    Kx "Kx" →TAG
    Ky "Ky" →TAG
    Kx xa x - * "Nax" →TAG
    Ky y ya - * "Nay" →TAG
  »
»
»

```

Haciendo correr el programa se obtiene:

```

Kx: 1.16305702352E-3
Ky: 8.79089086615E-4
Nax: 2.88341220414E-4
Nay: 2.8834122041E-4

```

Que son los coeficientes globales ideales y los flujos respectivos.

5.3.4.2. Ejercicio

10. Empleando los datos del ejercicio 9, calcule los coeficientes globales ideales (K'_x , K'_y) y los flujos molares de soluto (N_{Ax} , N_{Ay}).

5.3.5. Coeficientes globales reales de transferencia de masa

Estos coeficientes corresponden al proceso de difusión a través de un gas estancado y al igual que los coeficientes globales ideales están basados en una de las fases.

El flujo de soluto, calculado con el coeficiente global basado en la fase líquida es:

$$N_A = K_x(x^* - x) \quad (28)$$

Y calculado con el coeficiente global basado en la fase gaseosa:

$$N_A = K_y(y - y^*) \quad (29)$$

La relación entre estos coeficientes y los coeficientes individuales reales es la misma que en el caso ideal, sólo que en lugar de los coeficientes ideales se emplean los reales, así para la fase líquida tenemos:

$$\frac{1}{K_x} = \frac{1}{m''k_y} + \frac{1}{k_x} \tag{30}$$

$$m'' = \frac{y - y_i}{x^* - x_i}$$

Y para la fase gaseosa:

$$\frac{1}{K_y} = \frac{1}{k_y} + \frac{m'}{k_x} \tag{31}$$

$$m' = \frac{y_i - y}{x_i - x}$$

Por lo tanto, si se tienen los coeficientes individuales reales (k_y y k_x), los coeficientes globales pueden ser calculados con los programas "Kx1" y "Ky1". Como los coeficientes reales pueden ser calculados a partir de los ideales, los coeficientes reales globales pueden ser calculados también partiendo de los coeficientes individuales ideales, calculando previamente los coeficientes reales con el programa "xiyi2" y empleando los resultados de este programa en los programas "Kx1" y "Ky1".

5.3.5.1. Ejemplo

Empleando los datos del ejemplo 5.3.1.1, calcule los coeficientes globales reales (K_x , K_y) y los flujos respectivos (N_{Ax} , N_{Ay}).

Solución:

Los datos del ejemplo son:

```
x = 0.1
y = 0.38
k'x = 1.967x10-3 kgmol/s.m2
k'y = 1.465x10-3 kgmol/s.m2
[ 0 0.05 0.10 0.15 0.20 0.25 0.30 0.35 ] 'vx' STO
[ 0 0.022 0.052 0.087 0.131 0.187 0.265 0.385 ] 'vy' STO
```

Se tiene entonces todos los datos para resolver el problema. El programa elaborado para con ese fin es el siguiente:

```
« 0.1 0.38 1.967E-3 1.465E-3
« vx vy fun1 »
« vy vx fun1 »
0 0 0 0 0 0
→ x y kx ky fx fy xi yi
xa ya Kx Ky
« x y kx ky fx fy
xiyi2
3 ROLL DROP 3ROLL DROP
4 →LIST
{ xi yi kx ky } STO
y fy EVAL 'xa' STO
x fx EVAL 'ya' STO
y xi yi kx ky xa Kx1 'Kx' STO
x xi yi kx ky ya Ky1 'Ky' STO
Kx "Kx" →TAG
Ky "Ky" →TAG
Kx xa x - * "Nax" →TAG
Ky y ya - * "Nay" →TAG
»
»
```

Haciendo correr el programa se obtiene:

Kx: 1.52125124672E-3
 Ky: 1.14982786051E-3
 Nax: 3.77143538248E-4
 Nay: 3.77143538247E-4

Que son los coeficientes globales reales y los flujos respectivos.

5.3.5.2. Ejercicio

11. Empleando los datos del ejercicio 9, calcule los coeficientes globales reales basados en la fase líquida y gaseosa (K_x y K_y) y los flujos molares respectivos (N_{Ax} , N_{Ay}).

5.3.6. Cálculo de los coeficientes globales reales a partir de los coeficientes globales ideales

Los coeficientes globales reales (K_x y K_y) pueden ser calculados también a partir de los coeficientes globales ideales (K'_x y K'_y) pues están relacionados entre si mediante las siguientes expresiones:

$$K_x = \frac{K'_x}{(1-x)_{*m}}$$

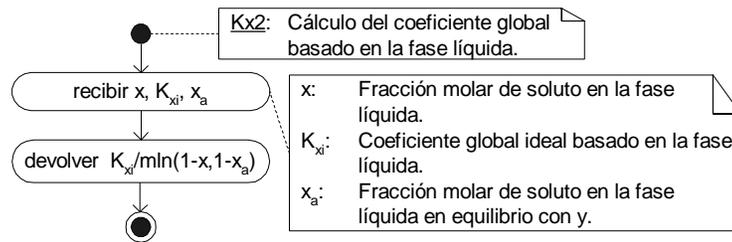
$$(1-x)_{*m} = \frac{(1-x) - (1-x^*)}{\ln\left(\frac{1-x}{1-x^*}\right)} \tag{32}$$

y

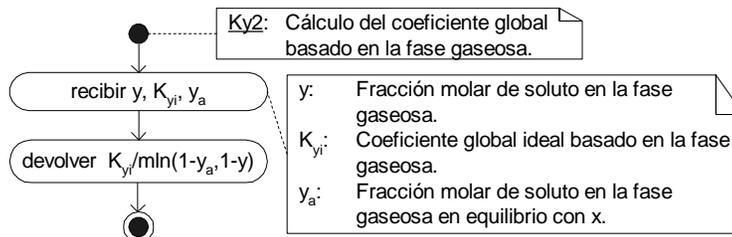
$$K_y = \frac{K'_y}{(1-y)_{*m}}$$

$$(1-y)_{*m} = \frac{(1-y^*) - (1-y)}{\ln\left(\frac{1-y^*}{1-y}\right)} \tag{33}$$

Aunque esta situación no es muy frecuente es conveniente automatizar el cálculo de estas ecuaciones. El algoritmo para la ecuación (32) es:



Y el algoritmo para la ecuación (33):



El código elaborado en base a estos algoritmos es:

```
xNAME Kx2 ( Coeficiente global basado en la fase líquida )
:: ( Datos en la pila: x, Kxi, xa )
  CK3&Dispatch
  # 111
  ::
    %1 4ROLL %- %1 ROT %- mln %/
  ;
;

xNAME Ky2 ( Coeficiente global basado en la fase gaseosa )
:: ( Datos en la pila: x, Kxi, xa )
  CK3&Dispatch
  # 111
  ::
    %1 SWAP %- %1 4ROLL %- mln %/
  ;
;
```

5.3.6.1. Ejemplo

Empleando los datos del ejemplo 5.3.1.1, calcule los coeficientes globales reales y los flujos de soluto específicos.

Solución

Una vez más los datos son:

```
x = 0.1
y = 0.38
k'x = 1.967x10-3 kgmol/s.m2
k'y = 1.465x10-3 kgmol/s.m2
[ 0 0.05 0.10 0.15 0.20 0.25 0.30 0.35 ] 'vx' STO
[ 0 0.022 0.052 0.087 0.131 0.187 0.265 0.385 ] 'vy' STO
```

Empleando los programas *Kx1* y *Ky1* se calcularán los coeficientes individuales globales y con los mismos y los programas *Kx2* y *Ky2* se calcularán los coeficientes globales reales. El programa elaborado con este fin es el siguiente:

```
« 0.1 0.38 1.967E-3 1.465E-3
« vx vy fun1 »
« vy vx fun1 »
0 0 0 0 0 0
→ x y kx ky fx fy xi yi
xa ya Kx Ky
« x y kx ky fx fy
xiyi1
2 →LIST
{ xi yi } STO
y fy EVAL 'xa' STO
x fx EVAL 'ya' STO
y xi yi kx ky xa Kx1 'Kx' STO
x xi yi kx ky ya Ky1 'Ky' STO
x Kx xa Kx2 'Kx' STO
y Ky ya Ky2 'Ky' STO
Kx "Kx" →TAG
Ky "Ky" →TAG
Kx xa x - * "Nax" →TAG
Ky y ya - * "Nay" →TAG
```

»
»

Haciendo correr el programa se obtiene:

K_x: 1.51164916754E-3
K_y: 1.13808541335E-3
N_{ax}: 3.74763022785E-4
N_{ay}: 3.73292015579E-4

Que son los coeficientes globales y flujos específicos respectivos. Como se puede observar, estos coeficientes difieren algo de los calculados en el ejemplo 5.3.5.1, esto se debe a que las ecuaciones (32) y (33) son menos rigurosas que las ecuaciones (30) y (31).

5.3.6.2. Ejercicio

12. Empleando los datos del ejercicio 9, calcule los coeficientes globales reales (K_x , K_y) y los flujos específicos respectivos (N_{Ax} , N_{Ay}).

5.4. Cálculo de la altura en columnas de absorción rellenas

Dependiendo de los coeficientes de transferencia de masa con los que se cuenta, existen una serie de relaciones que permiten calcular la altura de una columna de absorción rellena. Si se tienen los *coeficientes individuales* para las fases gaseosa y líquida, la altura puede ser calculada con una de las siguientes ecuaciones:

$$z = \int_{x_2}^{x_1} \frac{(1-x)_{im} L}{k'_{x} a S (1-x)(x_i - x)} dx \quad (34)$$

$$z = \int_{y_2}^{y_1} \frac{(1-y)_{im} V}{k'_{y} a S (1-y)(y - y_i)} dy \quad (35)$$

Donde k'_{xa} es el coeficiente volumétrico ideal para la fase líquida (kgmol/s.m^3), k'_{ya} es el coeficiente volumétrico ideal para la fase gaseosa (kgmol/s.m^3), S es la sección transversal de la columna (m^2), L es el flujo molar de líquido (kgmol/s), V es el flujo molar de gas (kgmol/s), x , x_i , $(1-x)_{im}$, y , y_i , $(1-y)_{im}$, tal como fueron definidas en las ecuaciones (14), (18) y (19).

Si se cuenta con los *coeficientes globales* para las fases gaseosa y líquida, la altura puede ser calculada con una de las siguientes ecuaciones:

$$z = \int_{x_2}^{x_1} \frac{(1-x)_{*m} L}{K'_{x} a S (1-x)(x^* - x)} dx \quad (36)$$

$$z = \int_{y_2}^{y_1} \frac{(1-y)_{*m} V}{K'_{y} a S (1-y)(y - y^*)} dy \quad (37)$$

Donde K'_{xa} y K'_{ya} son los coeficientes volumétricos globales ideales basados en las fases líquida y gaseosa respectivamente (kgmol/s.m^3), x^* , $(1-x)_{*m}$, y^* , $(1-y)_{*m}$ como fueron definidas en las ecuaciones (23), (25), (32) y (33).

Los coeficientes volumétricos que aparecen en estas ecuaciones obedecen a las mismas relaciones que los coeficientes superficiales estudiados en los acápites anteriores, en realidad son esos coeficientes multiplicados por el área interfacial por m^3 de torre rellena (a). Por consiguiente el cálculo de las composiciones de la interfaz puede ser llevado a cabo con los programas

"*xiy1*" o "*xiy2*" dependiendo si los datos disponibles son los coeficientes reales o ideales.

Los coeficientes individuales ideales están relacionados con los coeficientes individuales reales, de la misma forma que los coeficientes superficiales respectivos, es decir:

$$k_x a = \frac{k'_x a}{(1-x)_{im}} \quad (38)$$

$$k_y a = \frac{k'_y a}{(1-y)_{im}} \quad (39)$$

De igual manera, los coeficientes globales ideales están relacionados con los coeficientes globales reales de la misma forma que los coeficientes superficiales respectivos:

$$K_x a = \frac{K'_x a}{(1-x)_{*m}} \quad (40)$$

$$K_y a = \frac{K'_y a}{(1-y)_{*m}} \quad (41)$$

Empleando estos coeficientes en las ecuaciones (34) a (37) se tiene:

$$z = \int_{x_2}^{x_1} \frac{L}{k_x a S (1-x)(x_i - x)} dx \quad (42)$$

$$z = \int_{y_2}^{y_1} \frac{V}{k_y a S (1-y)(y - y_i)} dy \quad (43)$$

$$z = \int_{x_2}^{x_1} \frac{L}{K_x a S (1-x)(x^* - x)} dx \quad (44)$$

$$z = \int_{y_2}^{y_1} \frac{V}{K_y a S (1-y)(y - y^*)} dy \quad (45)$$

En el caso más general, donde las líneas de operación y de equilibrio son curvas, estas ecuaciones deben integrarse gráfica o numéricamente calculando para cada punto de integración las composiciones en la interfaz gas-líquido (siguiendo el procedimiento iterativo descrito en el acápite 5.3.3).

Se trata entonces de un procedimiento moroso que es conveniente automatizar, más aún si tomamos en cuenta que en un caso real, los coeficientes de transferencia de masa no son constantes, sino funciones de los flujos totales de gas y líquido.

5.4.1. Cálculo de la altura para mezclas gaseosas diluidas

Antes de abordar el cálculo de la altura para soluciones concentradas, estudiaremos el cálculo simplificado que se realiza con soluciones diluidas. Se considera que una solución está diluida si su concentración es menor o igual a 10%, en el caso de la absorción esta es una situación bastante frecuente, razón por la cual se justifica su estudio.

Puesto que en las soluciones diluidas los flujos molares del gas y líquido no varían mucho, se trabaja con los flujos medios. Se procede de igual manera con los coeficientes volumétricos de transferencia de masa y las

fracciones molares de gas y líquido. Por consiguiente estos términos pueden ser sacados de la integral quedando las ecuaciones (34) a (37) de la siguiente forma:

$$z = \left[\frac{L}{k'_x aS} \frac{(1-x)_{im}}{(1-x)} \right]_{prom} \int_{x_2}^{x_1} \frac{1}{(x_i - x)} dx \quad (46)$$

$$z = \left[\frac{V}{k'_y aS} \frac{(1-y)_{im}}{(1-y)} \right]_{prom} \int_{y_2}^{y_1} \frac{1}{(y - y_i)} dy \quad (47)$$

$$z = \left[\frac{L}{K'_x aS} \frac{(1-x)_{*m}}{(1-x)} \right]_{prom} \int_{x_2}^{x_1} \frac{1}{(x^* - x)} dx \quad (48)$$

$$z = \left[\frac{V}{K'_y aS} \frac{(1-y)_{*m}}{(1-y)} \right]_{prom} \int_{y_2}^{y_1} \frac{1}{(y - y^*)} dy \quad (49)$$

Puesto que las soluciones son diluidas, la línea de operación es esencialmente una línea recta. Si además se asume que la línea de equilibrio varía linealmente con $(y-y_i)$, $(x-x_i)$, (x^*-x) o $(y-y^*)$, entonces las integrales pueden ser resueltas obteniéndose las siguientes expresiones:

$$z = \left[\frac{L}{k'_x aS} \frac{(1-x)_{im}}{(1-x)} \right]_{prom} \frac{x_1 - x_2}{(x_i - x)_m} \quad (50)$$

$$z = \left[\frac{V}{k'_y aS} \frac{(1-y)_{im}}{(1-y)} \right]_{prom} \frac{y_1 - y_2}{(y - y_i)_m} \quad (51)$$

$$z = \left[\frac{L}{K'_x aS} \frac{(1-x)_{*m}}{(1-x)} \right]_{prom} \frac{x_1 - x_2}{(x^* - x)_m} \quad (52)$$

$$z = \left[\frac{V}{K'_y aS} \frac{(1-y)_{*m}}{(1-y)} \right]_{prom} \frac{y_1 - y_2}{(y - y^*)_m} \quad (53)$$

Donde:

$$(x_i - x)_m = \frac{(x_{i1} - x_1) - (x_{i2} - x_2)}{\ln \left(\frac{x_{i1} - x_1}{x_{i2} - x_2} \right)} \quad (54)$$

$$(y - y_i)_m = \frac{(y_1 - y_{i1}) - (y_2 - y_{i2})}{\ln \left(\frac{y_1 - y_{i1}}{y_2 - y_{i2}} \right)} \quad (55)$$

$$(x^* - x)_m = \frac{(x^*_1 - x_1) - (x^*_2 - x_2)}{\ln \left(\frac{x^*_1 - x_1}{x^*_2 - x_2} \right)} \quad (56)$$

$$(y - y^*)_m = \frac{(y_1 - y^*_1) - (y_2 - y^*_2)}{\ln \left(\frac{y_1 - y^*_1}{y_2 - y^*_2} \right)} \quad (57)$$

Que son las expresiones para el cálculo de la altura de una torre de ab-

sorción cuando las soluciones son diluidas. En estas expresiones los subíndices 1 se refieren al fondo de la columna y los subíndices 2 a la cabeza.

Tal como ocurre en las torres de enfriamiento, estas expresiones suelen reagruparse en dos términos: la altura equivalente a una unidad de transferencia (H) y el número de unidades de transferencia (N):

$$z = H_L N_L = \left(\left[\frac{L}{k'_x aS} \right]_{prom} \right) \left(\left[\frac{(1-x)_{im}}{(1-x)} \right]_{prom} \frac{x_1 - x_2}{(x_i - x)_m} \right) \quad (58)$$

$$z = H_G N_G = \left(\left[\frac{V}{k'_y aS} \right]_{prom} \right) \left(\left[\frac{(1-y)_{im}}{(1-y)} \right]_{prom} \frac{y_1 - y_2}{(y - y_i)_m} \right) \quad (59)$$

$$z = H_{OL} N_{OL} = \left(\left[\frac{L}{K'_x aS} \right]_{prom} \right) \left(\left[\frac{(1-x)_{*m}}{(1-x)} \right]_{prom} \frac{x_1 - x_2}{(x^* - x)_m} \right) \quad (60)$$

$$z = H_{OG} N_{OG} = \left(\left[\frac{V}{K'_y aS} \right]_{prom} \right) \left(\left[\frac{(1-y)_{*m}}{(1-y)} \right]_{prom} \frac{y_1 - y_2}{(y - y^*)_m} \right) \quad (61)$$

Cuando las soluciones son diluidas se asume además que los coeficientes volumétricos son constantes, por lo tanto, estas expresiones pueden ser simplificadas un tanto más quedando finalmente de la siguiente forma:

$$z = H_L N_L = \left(\frac{L_{prom}}{k'_x aS} \right) \left(\left[\frac{(1-x)_{im}}{(1-x)} \right]_{prom} \frac{x_1 - x_2}{(x_i - x)_m} \right) \quad (62)$$

$$z = H_G N_G = \left(\frac{V_{prom}}{k'_y aS} \right) \left(\left[\frac{(1-y)_{im}}{(1-y)} \right]_{prom} \frac{y_1 - y_2}{(y - y_i)_m} \right) \quad (63)$$

$$z = H_{OL} N_{OL} = \left(\frac{L_{prom}}{K'_x aS} \right) \left(\left[\frac{(1-x)_{*m}}{(1-x)} \right]_{prom} \frac{x_1 - x_2}{(x^* - x)_m} \right) \quad (64)$$

$$z = H_{OG} N_{OG} = \left(\frac{V_{prom}}{K'_y aS} \right) \left(\left[\frac{(1-y)_{*m}}{(1-y)} \right]_{prom} \frac{y_1 - y_2}{(y - y^*)_m} \right) \quad (65)$$

Estas son las expresiones que emplearemos en el cálculo de la altura de una torre de absorción para soluciones diluidas (cuando la concentración de soluto es menor o igual al 10%). En estas ecuaciones H_L es la altura individual de una unidad de transferencia basada en la fase líquida; H_G es la altura individual de una unidad de transferencia basada en la fase gaseosa; H_{OL} es la altura global de una unidad de transferencia basada en la fase líquida y H_{OG} es la altura global de una unidad de transferencia basada en la fase gaseosa (todas ellas en metros). N_L , N_G , N_{OL} y N_{OG} son los números de unidades de transferencia respectivos.

Si se conocen los coeficientes individuales de transferencia de masa, la altura de la torre se calcula con la ecuación (62) o la ecuación (63). Si se conoce el coeficiente global de transferencia de masa basado en la fase líquida, la altura de la torre se calcula con la ecuación (64) y si se conoce el coeficiente global de transferencia de masa basado en la fase gaseosa, la altura de la torre se calcula con la ecuación (65).

Si se conocen los coeficientes reales en lugar de los ideales, se reemplazan las ecuaciones (38) a (41) en las ecuaciones (62) a (65) quedando en

la siguiente forma:

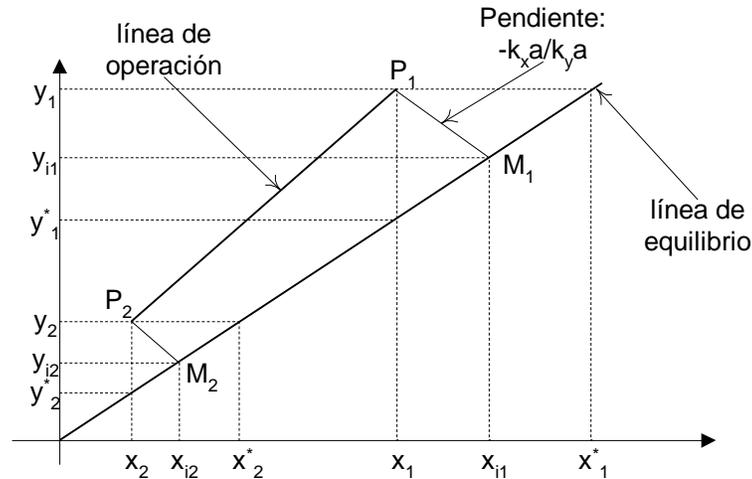
$$z = H_L N_L = \left(\frac{1}{k_x a S} \left[\frac{L}{(1-x)_{im}} \right]_{prom} \right) \left(\left[\frac{(1-x)_{im}}{(1-x)} \right]_{prom} \frac{x_1 - x_2}{(x_i - x)_m} \right) \quad (66)$$

$$z = H_G N_G = \left(\frac{1}{k_y a S} \left[\frac{V}{(1-y)_{im}} \right]_{prom} \right) \left(\left[\frac{(1-y)_{im}}{(1-y)} \right]_{prom} \frac{y_1 - y_2}{(y - y_i)_m} \right) \quad (67)$$

$$z = H_{OL} N_{OL} = \left(\frac{1}{K_x a S} \left[\frac{L}{(1-x)_{*m}} \right]_{prom} \right) \left(\left[\frac{(1-x)_{*m}}{(1-x)} \right]_{prom} \frac{x_1 - x_2}{(x^* - x)_m} \right) \quad (68)$$

$$z = H_{OG} N_{OG} = \left(\frac{1}{K_y a S} \left[\frac{V}{(1-y)_{*m}} \right]_{prom} \right) \left(\left[\frac{(1-y)_{*m}}{(1-y)} \right]_{prom} \frac{y_1 - y_2}{(y - y^*)_m} \right) \quad (69)$$

Las ecuaciones (62) a (65) (o (66) a (69)) pueden ser resueltas gráfica o numéricamente. El procedimiento gráfico para resolver las ecuaciones (62) y (63) (o (66) y (67)) es el que se detalla a continuación (un esquema del procedimiento gráfico se presenta en la siguiente figura):



- En una gráfica "x" (fracción molar en la fase líquida) versus "y" (fracción molar en la fase gaseosa), se traza la línea de equilibrio asumiendo que es una línea recta (por consiguiente sólo se requieren dos puntos).
- Con la ecuación (12) o la ecuación (13) se traza la línea de operación, asumiendo que se trata también de una línea recta (por lo tanto, al igual que el caso anterior, sólo se requieren dos puntos).
- Con las fracciones molares en el fondo (x_1, y_1) y cabeza (x_2, y_2) de la columna se trazan los puntos P_1 y P_2 .
- Partiendo del punto P_1 se ubica el punto M_1 siguiendo el procedimiento detallado en el acápite 5.3.1 cuando se conocen los coeficientes individuales reales o siguiendo el procedimiento detallado en el acápite 5.3.3 cuando se conocen los coeficientes individuales ideales. De esta manera se determina la composición de la interfaz gas-líquido (x_{i1}, y_{i1}) en el fondo de la columna.
- Se repite el procedimiento del inciso anterior para el punto P_2 , encontrándose así el punto M_2 y los respectivos valores de x_{i2}, y_{i2} .

- f) Se reemplazan los valores calculados en los incisos anteriores en las ecuaciones (19), (54) y (55) obteniéndose así los valores de $(1-x)_{im}$ y $(1-y)_{im}$ para el fondo y cabeza de la columna, así como los valores de $(x_i-x)_m$ y $(y-y_i)_m$.
- g) Finalmente se reemplazan los valores calculados en el inciso anterior en las ecuaciones (62) y (63) (o en las ecuaciones (66) y (67) si los coeficientes son reales) obteniéndose así la altura de la torre.

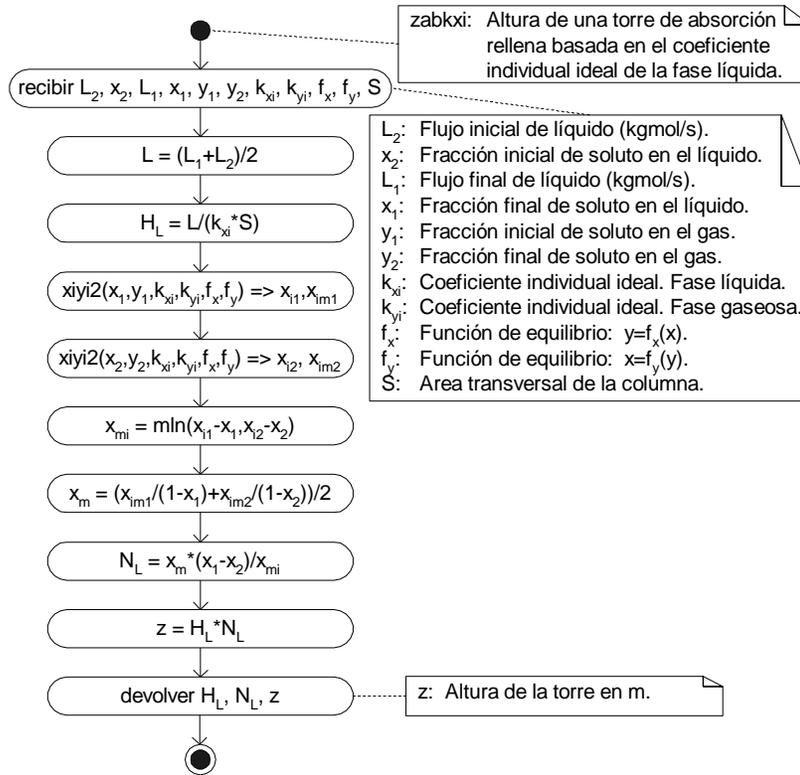
El procedimiento gráfico que se sigue para calcular la altura de la torre con las ecuaciones (64) y (65) (o (68) y (69)) es el que se detalla a continuación.

- a) En una gráfica "x" (fracción molar en la fase líquida) versus "y" (fracción molar en la fase gaseosa), se traza la línea de equilibrio asumiendo que es una línea recta (por consiguiente sólo se requieren dos puntos).
- b) Se trazan líneas rectas desde x_1, y_1, x_2, y_2 hasta la línea de equilibrio (ver la figura de la página anterior), determinándose así los valores de y^*_1, x^*_1, y^*_2 y x^*_2 .
- c) Dependiendo del coeficiente global conocido, se reemplazan los valores calculados en el inciso anterior en las ecuaciones (32) o (33) y (56) o (57) obteniéndose así los valores de $(1-x)^*_m$ o $(1-y)^*_m$, para el fondo y cabeza de la columna, así como el valor de $(x^*-x)_m$ o $(y-y^*)_m$.
- d) Se reemplazan los valores calculados en el inciso anterior en las ecuaciones (64) o (65) (o (68) o (69)) obteniéndose así la altura de la torre.

Como se puede apreciar los procedimientos gráficos son morosos y requieren además una serie de cálculos manuales, razón por la cual es conveniente automatizarlos mediante programas.

El algoritmo que calcula la altura de la torre con la ecuación (62) se presenta en el diagrama de actividades de la siguiente página y el código elaborado en base al mismo es el siguiente:

```
xNAME zabkxi ( Altura de una torre de absorción basada en )
::          ( el coeficiente individual ideal de la fase líquida )
%11 CKNNOLASTWD DROP      ( Datos: L2=20= Flujo inicial, líquido )
CK&DISPATCH1             ( x2=19= Fracción inicial, líquido )
# 11881                   ( L1=18= Flujo final, líquido )
::                        ( x1=17= Fracción final, líquido )
TEN ROLL TEN ROLL TEN ROLL ( y1=16= Fracción inicial, gas )
TEN ROLL TEN ROLL         ( y2=15= Fracción final, gas )
CK&DISPATCH1             ( kxi=14= Coef. ind. ideal, líquido )
# 11111                   ( kyi=13= Coef. ind. ideal, gas )
::                        ( fx=12= func. de equilibrio: y=fx[x] )
10UNROLL 10UNROLL 10UNROLL ( fy=11= func. de equilibrio: x=fy[y] )
10UNROLL 10UNROLL         ( S=10= Area transversal )
ELEVEN ROLL               ( Temporales: L=9; HL=8; xi1=7; )
CK&DISPATCH1             ( xim1=6; xi2=5; xim2=4; )
ONE                       ( xmi=3; xm=2; NL=1 )
::
ELEVEN UNROLL
%0 NINE NDUPN DROP
' NULLLAM TWENTY NDUPN DOBIND
18GETLAM 20GETLAM %+ %2 %/ 9PUTLAM ( L=[L1+L2]/2 )
```



```

9GETLAM 14GETLAM 10GETLAM %* %/
8PUTLAM                                     ( HL=L/[kxi*S] )
17GETLAM 16GETLAM 14GETLAM 13GETLAM
12GETLAM 11GETLAM xxiyi2 3DROP             ( xiyi2[x1,y1,kxi,kyi,fx,fy] )
6PUTLAM DROP 7PUTLAM                       ( => xim1,xi1 )
19GETLAM 15GETLAM 14GETLAM 13GETLAM
12GETLAM 11GETLAM xxiyi2 3DROP             ( xiyi2[x2,y2,kxi,kyi,fx,fy] )
4PUTLAM DROP 5PUTLAM                       ( => xim2,xi2 )
7GETLAM 17GETLAM %- 5GETLAM
19GETLAM %- mln 3PUTLAM                    ( xmi=mln[xi1-x1,xi2-x2] )
6GETLAM %1 17GETLAM %- %/ 4GETLAM         ( [xim1/[1-x1]+xim2 )
%1 19GETLAM %- %/ %+ %2 %/                ( /[1-x2]]/2 )
2PUTLAM                                     ( => xm )
2GETLAM 17GETLAM 19GETLAM %- %*          ( xm*[x1-x2] )
3GETLAM %/ 1PUTLAM                         ( /xmi => NL )
8GETLAM 1GETLAM 2DUP %*                    ( HL, NL, z )
ABND
;
ZERO
::
    ELEVEN UNROLL
    # 202 DO#EXIT                           ( 202 = Bad argument type )
;
;
ZERO
::
    10UNROLL 10UNROLL 10UNROLL
    10UNROLL 10UNROLL
    # 202 DO#EXIT                           ( 202 = Bad argument type )
;
;
;
    
```

5.4.1.1. Ejemplo

Se absorbe acetona con agua en una torre rellena que tiene una sección transversal de 0.186 m^2 a 293 K y 101.32 kPa (1 atm). El aire de entrada contiene 2.6% mol de acetona y el de salida 0.5% . El flujo de gas es 13.65 kgmol de aire inerte/h (30.1 lbmol/h). El flujo de entrada de agua pura es de 45.36 kgmol de agua/h (100 lbmol/h). Los coeficientes de película para los flujos dados en la torre son $k'_{ya} = 3.78 \times 10^{-2} \text{ kgmol/s.m}^3$ ($8.50 \text{ lb mol/h.pie}^3$) y $k'_{xa} = 6.16 \times 10^{-2} \text{ kgmol/s.m}^3$ ($13.85 \text{ lbmol/h.pie}^3$). Los datos de equilibrio a 293 K son los siguientes:

x	0	0.0333	0.0720	0.117	0.171
$P_a(\text{mmHg})$	0	30.0	62.8	85.4	103.0

Calcule la altura de una unidad de transferencia, el número de unidades de transferencia y la altura de la torre en base al coeficiente individual ideal de la fase líquida.

Solución:

Los datos con los que contamos son:

$S = 0.186 \text{ m}^2$
 $P = 1 \text{ atm} = 760 \text{ mmHg}$
 $y_1 = 0.026$
 $y_2 = 0.005$
 $x_2 = 0.0$
 $C = 13.65 \text{ kgmol/h} = 13.65 \text{ kgmol/h} \cdot 1 \text{ kgmol/s} \text{ CONVERT } 6 \text{ RND}$
 $= 0.003792 \text{ kgmol/s}$
 $B = 45.36 \text{ kgmol/h} = 45.36 \text{ kgmol/h} \cdot 1 \text{ kgmol/s} \text{ CONVERT } 6 \text{ RND} = 0.0126 \text{ kgmol/s}$
 $k'_{xa} = 0.0616 \text{ kgmol/s.m}^3$
 $k'_{ya} = 0.0378 \text{ kgmol/s.m}^3$

Se tienen además los datos de equilibrio:

$[0 \ 0.033 \ 0.0720 \ 0.117 \ 0.171] \text{ 'vx' STO}$
 $[0 \ 30 \ 62.8 \ 85.4 \ 103] \ 760 / \text{ 'vy' STO}$

Puesto que el líquido inicial es puro:

$L_2 = B = 0.0126 \text{ kgmol/s}$

Con "C" y la ecuación (3) podemos calcular V_1 y V_2 :

$V_1 = C/(1-y_1) = 0.003792 \cdot 1 \ 0.026 - / 6 \text{ RND} = 0.003893 \text{ kgmol/s}$
 $V_2 = C/(1-y_2) = 0.003792 \cdot 1 \ 0.005 - / 6 \text{ RND} = 0.003811 \text{ kgmol/s}$

Entonces el porcentaje absorbido (ecuación 4) es:

$p_a = (V_1 \cdot y_1 - V_2 \cdot y_2) / (V_1 \cdot y_1)$
 $= 0.003893 \ 0.026 * 0.003811 \ 0.005 * - 0.003893 \ 0.026 * / 100 * 3 \text{ RND}$
 $= 81.174$

Con este valor podemos calcular los flujos y composiciones faltantes empleando el programa *abrcl*:

```

« 0.0126 0 0.003893 0.026 81.174
  abrcl
  DROP DROP 2 →LIST 6 RND
  { L1 x1 } →TAG
  LIST→ DROP
»

```

Haciendo correr el programa se obtiene:

$L1: .012682$
 $x1: .006479$

Que son los flujos y composiciones faltantes.

Ahora contamos con todos los datos para resolver el problema empleando "zabkxi". El programa que resuelve el problema es el siguiente:

```
« 0.0126 0 0.012682 0.006479 0.026 0.005
0.0616 0.0378
« vx vy fun1 »
« vy vx fun1 »
0.186
zabkxi
3 →LIST
{ HL NL z } →TAG LIST→ DROP
»
```

Haciendo correr el programa se obtiene:

```
HL: 1.1032851557
NL: 1.77724584809
z: 1.96080896223
```

5.4.1.2. Ejercicios

13. Se empleará una torre de absorción rellena (0.4 m de diámetro) para absorber el 80% del SO₂ contenido en 10 kgmol/h de una mezcla gaseosa de SO₂-aire, empleando 400 kgmol/h de agua pura a 293.2 K. El gas se encuentra a una presión de 1.984x10⁵ Pa y la presión parcial inicial del SO₂ en la mezcla es 1.567x10⁴ Pa. Los coeficientes volumétricos individuales de transferencia de masa son: k'_{xa} = 0.848 kgmol/s.m³ y k'_{ya} = 0.034 kgmol/s.m³. Calcule la altura de una unidad de transferencia, el número de unidades de transferencia y la altura de la torre basada en el coeficiente de la fase líquida. Los datos de equilibrio son:

x_A	P_A (mm Hg)
0.0000000	0.0
0.0000562	0.5
0.0001403	1.2
0.0002800	3.2
0.0004220	5.8
0.0005640	8.5
0.0008420	14.1
0.0014030	26.0
0.0019650	39.0
0.0027900	59.0
0.0042000	92.0
0.0069800	161.0
0.0138500	336.0
0.0206000	517.0
0.0273000	698.0

14. Una corriente de gas contiene 4.0 mol% de NH₃ y su contenido de amoníaco se reduce a 0.5 mol% en una torre de absorción rellena que opera a 293 K y 1.013x10⁵ Pa. El flujo de agua pura de entrada es de 68.0 kgmol/h y el flujo total de gas de entrada es de 57.8 kgmol/h. (El diámetro de la torre es 0.747 m.) Los coeficientes de transferencia de masa individuales son k'_{ya} = 0.0739 kgmol/s.m³ y k'_{xa} = 0.169 kgmol/s.m³. Calcule la altura de una unidad de transferencia, el número de unidades de transferencia y la altura de la torre en base al coeficiente individual ideal de la fase líquida. Los datos de equilibrio son:

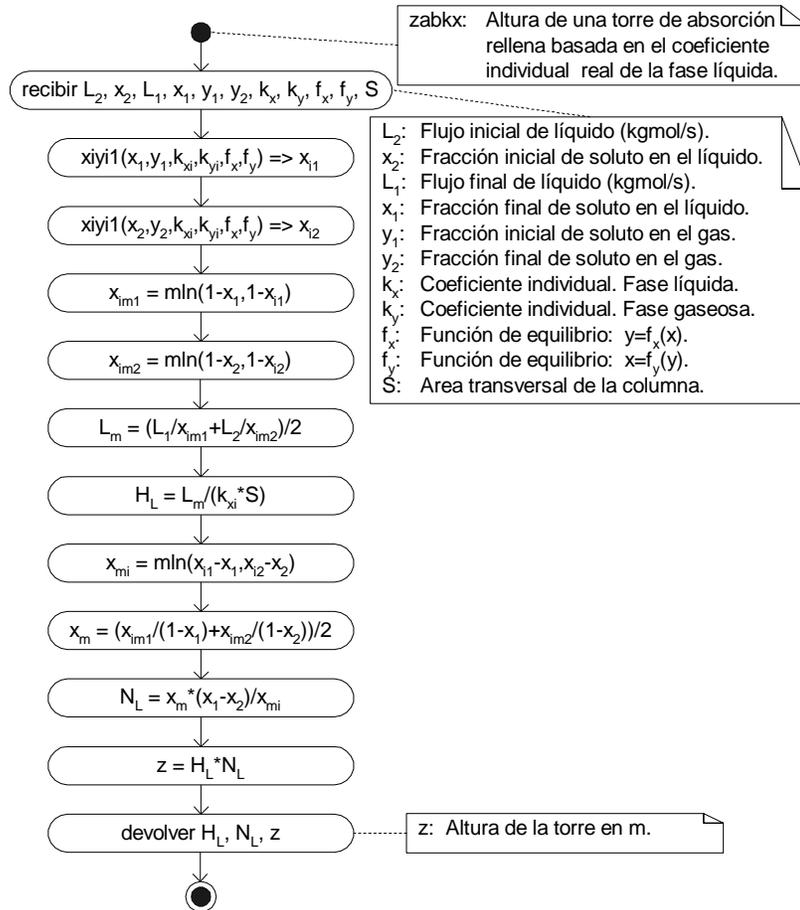
x_A	P_A (mm Hg)
0.0000	0.0
0.0208	12.0
0.0258	15.0
0.0309	18.2
0.0405	24.9
0.0503	31.7
0.0737	50.0
0.0960	69.6
0.1370	114.0
0.1750	166.0
0.2100	227.0
0.2410	298.0
0.2970	470.0

15. Se empleará una torre de absorción rellena para recuperar el 90% del metanol contenido en 450 kgmol/h de una mezcla gaseosa de metanol-aire que se encuentra a una presión de 720 mm de Hg y una temperatura de 45°C (diámetro de la torre 0.5 m). La presión parcial inicial del metanol es 65 mm de Hg. Los coeficientes individuales de transferencia de masa son $k'_{xa} = 18.7$ lbmol/h.pie³ y $k'_{ya} = 9.03$ lbmol/h.pie³. Para la absorción se empleará 500 kgmol/h de agua pura. Calcule la altura de una unidad de transferencia, el número de unidades de transferencia y al altura de la torre. Los datos de equilibrio para este sistema son los siguientes:

x_A	P_A (39.9°C)	P_A (59.4°C)
0.00	0	0
0.05	25.0	50
0.10	46.0	102
0.15	66.5	151

El cálculo de la altura de la torre cuando se conocen los coeficientes reales de transferencia de masa (ecuación 66) es similar al de la ecuación (62), excepto que las composiciones de la interfaz gas-líquido se calculan con "xiy1" en lugar de "xiy2". El algoritmo para resolver esta ecuación se presenta en la siguiente página y el código elaborado en base al mismo es el siguiente:

```
xNAME zabkx ( Altura de una torre de absorción basada en )
:: ( el coeficiente individual real de la fase líquida )
%11 CKNNOLASTWD DROP ( Datos: L2=20= Flujo inicial, líquido )
CK&DISPATCH1 ( x2=19= Fracción inicial, líquido )
# 11881 ( L1=18= Flujo final, líquido )
:: ( x1=17= Fracción final, líquido )
TEN ROLL TEN ROLL TEN ROLL ( y1=16= Fracción inicial, gas )
TEN ROLL TEN ROLL ( y2=15= Fracción final, gas )
CK&DISPATCH1 ( kx=14= Coef. ind. real, líquido )
# 11111 ( ky=13= Coef. ind. real, gas )
:: ( fx=12= func. de equilibrio: y=fx[x] )
10UNROLL 10UNROLL 10UNROLL ( fy=11= func. de equilibrio: x=fy[y] )
10UNROLL 10UNROLL ( S=10= Area transversal )
ELEVEN ROLL ( Temporales: L=9; HL=8; xi1=7; )
CK&DISPATCH1 ( xim1=6; xi2=5; xim2=4; )
ONE ( xmi=3; xm=2; NL=1 )
::
```



```

ELEVEN UNROLL
%0 NINE NDUPN DROP
' NULLLAM TWENTY NDUPN DOBIND
17GETLAM 16GETLAM 14GETLAM 13GETLAM
12GETLAM 11GETLAM xxiyi1 DROP      ( xiyi1[x1,y1,kx,ky,fx,fy] )
7PUTLAM                               ( => xi1 )
19GETLAM 15GETLAM 14GETLAM 13GETLAM
12GETLAM 11GETLAM xxiyi1 DROP      ( xiyi1[x2,y2,kx,ky,fx,fy] )
5PUTLAM                               ( => xi2 )
%1 17GETLAM %- %1 7GETLAM %- mln    ( mln[1-x1,1-xi1] )
6PUTLAM                               ( => xim1 )
%1 19GETLAM %- %1 5GETLAM %- mln    ( mln[1-x2,1-xi2] )
4PUTLAM                               ( => xim2 )
18GETLAM 6GETLAM %/ 20GETLAM
4GETLAM %/ %+ %2 %/ 9PUTLAM        ( Lm=[L1/xim1+L2/xim2]/2 )
9GETLAM 14GETLAM 10GETLAM %* %/
8PUTLAM                             ( HL=Lm/[kx*S] )
7GETLAM 17GETLAM %- 5GETLAM
19GETLAM %- mln 3PUTLAM            ( xmi=mln[xi1-x1,xi2-x2] )
6GETLAM %1 17GETLAM %- %/ 4GETLAM  ( [xim1/[1-x1]+xim2 ]
%1 19GETLAM %- %/ %+ %2 %/        ( /[1-x2]]/2 )
2PUTLAM                             ( => xm )
2GETLAM 17GETLAM 19GETLAM %- %*    ( xm*[x1-x2] )
3GETLAM %/ 1PUTLAM                 ( /xmi => NL )
8GETLAM 1GETLAM 2DUP %*           ( HL, NL, z )
ABND
    
```

;

```

ZERO
::
  ELEVEN UNROLL
  # 202 DO#EXIT                    ( 202 = Bad argument type )
;
;
ZERO
::
  10UNROLL 10UNROLL 10UNROLL
  10UNROLL 10UNROLL
  # 202 DO#EXIT                    ( 202 = Bad argument type )
;
;
;

```

5.4.1.3. Ejemplo

Volveremos a resolver el ejemplo 5.4.1.1 pero empleando los coeficientes reales: $k_{ya} = 0.0380$; $k_{xa} = 0.0626$.

Solución:

Los datos y el programa son los mismos que en el mencionado ejemplo, con excepción por supuesto de los coeficientes y el programa "zabkx":

```

« 0.0126 0 0.012682 0.006479 0.026 0.005
  0.0626 0.0381
  « vx vy fun1 »
  « vy vx fun1 »
  0.186
  zabkx 3 →LIST
  { HL NL z } →TAG LIST→ DROP
»

```

Haciendo correr el programa se obtiene:

```

HL: 1.09146660373
NL: 1.79653542417
z: 1.9608584179

```

Que son los tres valores requeridos.

5.4.1.4. Ejercicios

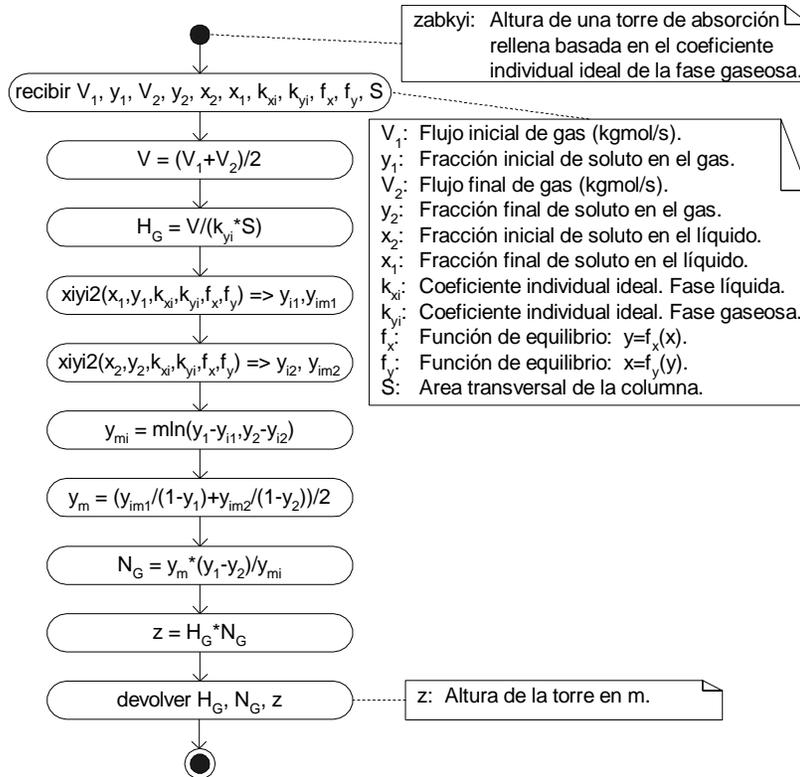
16. Repita el ejercicio 13 empleando los coeficientes reales: $k_{xa} = 0.86$ $kgmol/s.m^3$ y $k_{ya} = 0.038$ $kgmol/s.m^3$.
17. Repita el ejercicio 14 empleando los coeficientes reales: $k_{ya} = 0.0746$ $kgmol/s.m^3$ y $k_{xa} = 0.174$ $kgmol/s.m^3$.
18. Repita el ejercicio 15 empleando los coeficientes reales: $k_{xa} = 19.4$ $lbmol/h.pie^3$ y $k_{ya} = 9.67$ $lbmol/h.pie^3$.

La resolución de la ecuación (63) es muy similar al de la ecuación (62), sólo que se trabaja con el coeficiente y las fracciones en la fase gaseosa en lugar de los de la fase líquida. El algoritmo para esta ecuación se presenta en la siguiente página y el programa elaborado en base al mismo es:

```

xNAME zabkyi ( Altura de una torre de absorción basada en )
::          ( el coeficiente individual ideal de la fase gaseosa )
  %11 CKNNOLASTWD DROP      ( Datos: V1=20= Flujo inicial, líquido )
  CK&DISPATCH1            ( y1=19= Fracción inicial, líquido )

```



```

# 11881          ( V2=18= Flujo final, líquido )
::              ( y2=17= Fracción final, líquido )
TEN ROLL TEN ROLL TEN ROLL ( x2=16= Fracción inicial, gas )
TEN ROLL TEN ROLL          ( x1=15= Fracción final, gas )
CK&DISPATCH1          ( kxi=14= Coef. ind. ideal, líquido )
# 11111          ( kyi=13= Coef. ind. ideal, gas )
::              ( fx=12= func. de equilibrio: y=fx[x] )
10UNROLL 10UNROLL 10UNROLL ( fy=11= func. de equilibrio: x=fy[y] )
10UNROLL 10UNROLL          ( S=10= Área transversal )
ELEVEN ROLL          ( Temporales: V=9; HG=8; yil=7; )
CK&DISPATCH1          ( yim1=6; yi2=5; yim2=4; )
ONE                  ( ymi=3; ym=2; NG=1 )
::
ELEVEN UNROLL
%0 NINE NDUPN DROP
' NULLLAM TWENTY NDUPN DOBIND
20GETLAM 18GETLAM %+ %2 %/ 9PUTLAM ( V=[V1+V2]/2 )
9GETLAM 13GETLAM 10GETLAM %* %/
8PUTLAM ( HG=V/[kyi*S] )
15GETLAM 19GETLAM 14GETLAM 13GETLAM
12GETLAM 11GETLAM xxiyi2 2DROP ( xiyi2[x1,y1,kxi,kyi,fx,fy] )
6PUTLAM DROP 7PUTLAM DROP ( => yim1,yil )
16GETLAM 17GETLAM 14GETLAM 13GETLAM
12GETLAM 11GETLAM xxiyi2 2DROP ( xiyi2[x2,y2,kxi,kyi,fx,fy] )
4PUTLAM DROP 5PUTLAM DROP ( => yim2,yi2 )
19GETLAM 7GETLAM %- 17GETLAM
5GETLAM %- mln 3PUTLAM ( ymi=mln[y1-yil,y2-yi2] )
6GETLAM %1 19GETLAM %- %/ 4GETLAM ( [yim1/[1-y1]+yim2 )
%1 17GETLAM %- %/ %+ %2 %/ ( /[1-y2]]/2 )
2PUTLAM ( => ym )
2GETLAM 19GETLAM 17GETLAM %- %* ( ym*[y1-y2] )
  
```

```

      3GETLAM %/ 1PUTLAM          ( /ymi => NG )
      8GETLAM 1GETLAM 2DUP %*    ( HG, NG, z )
      ABND
    ;
    ZERO
  ::
    ELEVEN UNROLL
    # 202 DO#EXIT                ( 202 = Bad argument type )
  ;
;
ZERO
::
  10UNROLL 10UNROLL 10UNROLL
  10UNROLL 10UNROLL
  # 202 DO#EXIT                ( 202 = Bad argument type )
;
;
;

```

5.4.1.5. Ejemplo

Volveremos a resolver el ejemplo 5.4.1.1 pero calculando el resultado en función al coeficiente individual de la fase gaseosa.

Solución:

Los datos son los mismos que en el mencionado ejemplo. El programa que resuelve el problema es el siguiente:

```

« 0.003893 0.026 0.003811 0.005 0 0.006479
  0.0616 0.0378
  « vx vy fun1 »
  « vy vx fun1 »
  0.186
  zabkyi
  3 →LIST
  { HG NG z } →TAG LIST→ DROP
»

```

Haciendo correr el programa se obtiene:

```

HG: .547875064004
NG: 3.58225984667
z: 1.96263084277

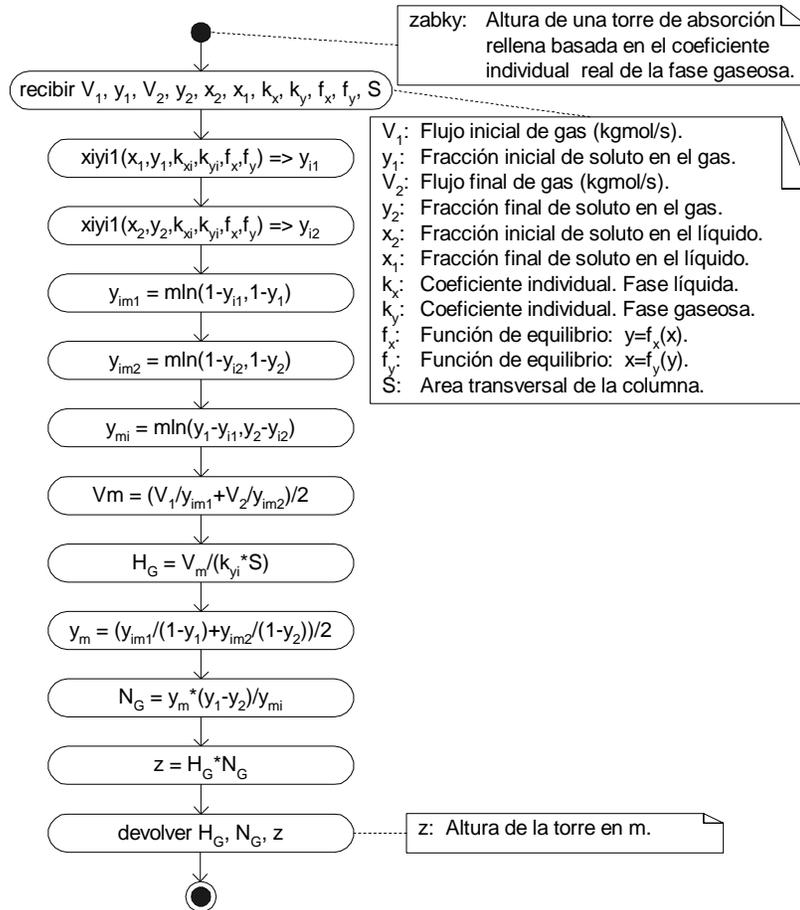
```

Que es la altura de una unidad de transferencia, el número de unidades de transferencia y la altura de la torre en metros.

5.4.1.6. Ejercicios

19. Repita el ejercicio 13 basado en el coeficiente de la fase gaseosa.
20. Repita el ejercicio 14 basado en el coeficiente de la fase gaseosa.
21. Repita el ejercicio 15 basado en el coeficiente de la fase gaseosa.

La resolución de la ecuación (67) es similar al de la ecuación (66), sólo que se trabaja con el coeficiente y fracciones de la fase gaseosa. El algoritmo que resuelve la ecuación (67) se presenta en el diagrama de actividades de la siguiente página.



Y el código respectivo es:

```

xNAME zabky ( Altura de una torre de absorción basada en )
::
%11 CKNNOLASTWD DROP ( Datos: V1=20= Flujo inicial, líquido )
CK&DISPATCH1 ( y1=19= Fracción inicial, líquido )
# 11881 ( V2=18= Flujo final, líquido )
::
TEN ROLL TEN ROLL TEN ROLL ( y2=17= Fracción final, líquido )
TEN ROLL TEN ROLL ( x2=16= Fracción inicial, gas )
CK&DISPATCH1 ( x1=15= Fracción final, gas )
# 11111 ( kx=14= Coef. ind. ideal, líquido )
::
10UNROLL 10UNROLL 10UNROLL ( ky=13= Coef. ind. ideal, gas )
10UNROLL 10UNROLL ( fx=12= func. de equilibrio: y=fx[x] )
ELEVEN ROLL ( fy=11= func. de equilibrio: x=fy[y] )
CK&DISPATCH1 ( S=10= Área transversal )
ONE ( Temporales: Vm=9; HG=8; y11=7; )
::
ELEVEN UNROLL ( yim1=6; y12=5; yim2=4; )
%0 NINE NDUPN DROP ( ymi=3; ym=2; NG=1 )
' NULLLAM TWENTY NDUPN DOBIND
15GETLAM 19GETLAM 14GETLAM 13GETLAM
12GETLAM 11GETLAM xxiy1 7PUTLAM ( xiy1[x1,y1,kx,ky,fx,fy] )
DROP ( => y11 )
16GETLAM 17GETLAM 14GETLAM 13GETLAM
12GETLAM 11GETLAM xxiy1 5PUTLAM ( xiy1[x2,y2,kx,ky,fx,fy] )
DROP ( => y12 )
    
```

```

%1 7GETLAM %- %1 19GETLAM %- mln      ( mln[1-yi1,1-y1] )
6PUTLAM                                ( => yim1 )
%1 5GETLAM %- %1 17GETLAM %- mln      ( mln[1-yi2,1-y2] )
4PUTLAM                                ( => yim2 )
19GETLAM 7GETLAM %- 17GETLAM
5GETLAM %- mln 3PUTLAM                ( ymi=mln[y1-yi1,y2-yi2] )
20GETLAM 6GETLAM %/ 18GETLAM
4GETLAM %/ %+ %2 %/ 9PUTLAM          ( Vm=[V1/yim1+V2/yim2]/2 )
9GETLAM 13GETLAM 10GETLAM %* %/
8PUTLAM                                ( HG=Vm/[ky*S] )
6GETLAM %1 19GETLAM %- %/ 4GETLAM     ( [yim1/[1-y1]+yim2 ]
%1 17GETLAM %- %/ %+ %2 %/           ( /[1-y2]]/2 )
2PUTLAM                                ( => ym )
2GETLAM 19GETLAM 17GETLAM %- %*      ( ym*[y1-y2] )
3GETLAM %/ 1PUTLAM                    ( /ymi => NG )
8GETLAM 1GETLAM 2DUP %*              ( HG, NG, z )
ABND
;
ZERO
::
    ELEVEN UNROLL
    # 202 DO#EXIT                      ( 202 = Bad argument type )
;
;
ZERO
::
    10UNROLL 10UNROLL 10UNROLL
    10UNROLL 10UNROLL
    # 202 DO#EXIT                      ( 202 = Bad argument type )
;
;
;

```

5.4.1.7. Ejemplo

Volveremos a resolver el ejemplo 5.4.1.3 pero calculando el resultado en función al coeficiente individual real de la fase gaseosa.

Solución:

Los datos son los mismos que en el mencionado ejemplo. El programa que resuelve el problema es el siguiente:

```

« 0.003893 0.026 0.003811 0.005 0 0.006479
0.0623 0.0381
« vx vy fun1 »
« vy vx fun1 »
0.186
zabky 3 →LIST
{ HG NG z } →TAG LIST→ DROP
»

```

Haciendo correr el programa se obtiene:

```

HG: .550338149778
NG: 3.56530144981
z: 1.96212140329

```

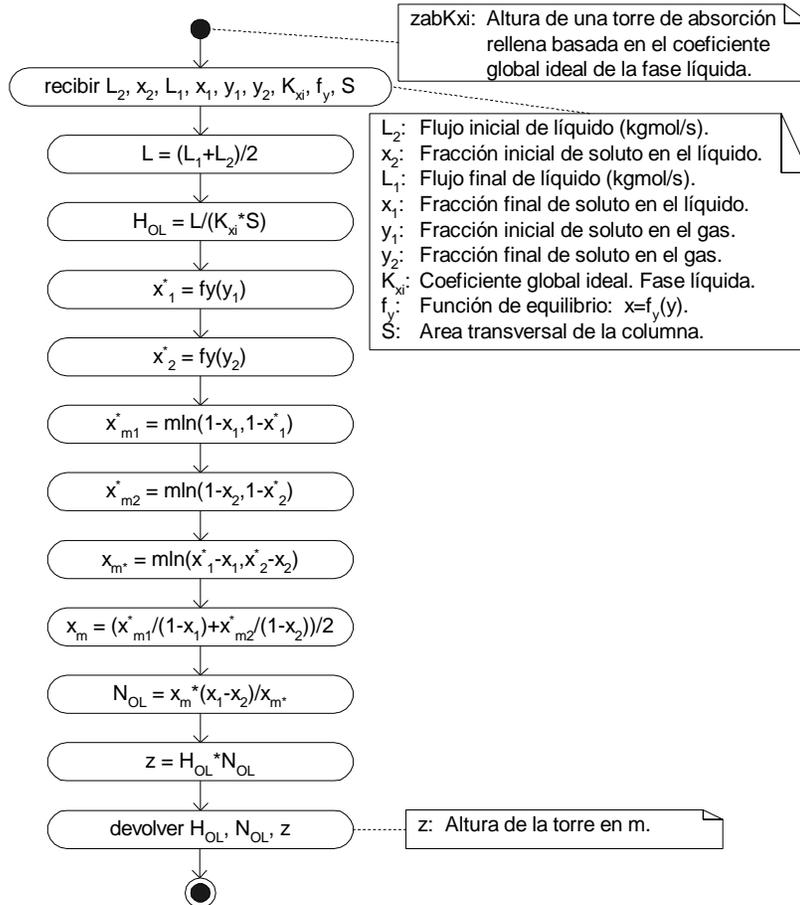
5.4.1.8. Ejercicios

22. Repita el ejercicio 16 basado en el coeficiente de la fase gaseosa.

23. Repita el ejercicio 17 basado en el coeficiente de la fase gaseosa.

24. Repita el ejercicio 18 basado en el coeficiente de la fase gaseosa.

El algoritmo para el cálculo de la altura de una torre de absorción empleando el coeficiente global ideal de la fase líquida (ecuación 64) es el siguiente:



Y el código elaborado en base al mismo es:

```

xNAME zabKxi ( Altura de una torre de absorción basada en )
::          ( el coeficiente global ideal de la fase líquida )
%9 KKNOLASTWD DROP      ( Datos: L2=18= Flujo inicial, líquido )
CK&DISPATCH1          ( x2=17= Fracción inicial, líquido )
# 11181                ( L1=16= Flujo final, líquido )
::                    ( x1=15= Fracción final, líquido )
NINE ROLL NINE ROLL    ( y1=14= Fracción inicial, gas )
NINE ROLL NINE ROLL    ( y2=13= Fracción final, gas )
CK&DISPATCH1          ( Kxi=12= Coef. glob. ideal, líquido )
# 1111                ( fy=11= func. de equilibrio: x=fy[y] )
::                    ( S=10= Área transversal )
NINE UNROLL NINE UNROLL ( Temporales: L=9; HOL=8; xa1=7; )
NINE UNROLL NINE UNROLL (          xam1=6; xa2=5; xam2=4; )
%0 NINE NDUPN DROP     (          xma=3; xm=2; NOL=1 )
' NULLLAM EIGHTEEN NDUPN DOBIND
16GETLAM 18GETLAM %+ %2 %/ 9PUTLAM ( L=[L1+L2]/2 )
9GETLAM 12GETLAM 10GETLAM %* %/
8PUTLAM                ( HOL=L/[Kxi*S] )
    
```

```

14GETLAM 11GETLAM EVAL 7PUTLAM      ( xa1=fy[y1] )
13GETLAM 11GETLAM EVAL 5PUTLAM      ( xa2=fy[y2] )
%1 15GETLAM %- %1 7GETLAM %- mln    ( mln[1-x1,1-xa1] )
6PUTLAM                              ( => xam1 )
%1 17GETLAM %- %1 5GETLAM %- mln    ( mln[1-x2,1-xa2] )
4PUTLAM                              ( => xam2 )
7GETLAM 15GETLAM %- 5GETLAM
17GETLAM %- mln 3PUTLAM              ( xma=mln[xa1-x1,xa2-x2] )
6GETLAM %1 15GETLAM %- %/ 4GETLAM    ( [xam1/[1-x1]+xam2 ] )
%1 17GETLAM %- %/ %+ %2 %/          ( /[1-x2]]/2 )
2PUTLAM                              ( => xm )
2GETLAM 15GETLAM 17GETLAM %- %*      ( xm*[x1-x2] )
3GETLAM %/ 1PUTLAM                  ( /xma => NOL )
8GETLAM 1GETLAM 2DUP %*              ( HOL, NOL, z )
ABND
;
ZERO
::
  NINE UNROLL NINE UNROLL
  NINE UNROLL NINE UNROLL
  # 202 DO#EXIT                      ( 202 = Bad argument type )
;
;
;

```

5.4.1.9. Ejemplo

Volveremos a resolver el ejemplo 5.4.1.1 pero calculando el resultado con el coeficiente global ideal $K'_x = 0.026 \text{ kgmol/s.m}^3$.

Solución:

Los datos son los mismos que en el mencionado ejemplo. El programa que resuelve el problema es el siguiente:

```

« 0.0126 0 0.012682 0.006479 0.026 0.005
  0.026
« vy vx fun1 »
  0.186
  zabKxi 3 →LIST
  { HOL NOL z } →TAG LIST→ DROP
»

```

Haciendo correr el programa se obtiene:

```

HOL: 2.61393713813
NOL: .753586483847
z: 1.96982769692

```

Cuando no se conocen los coeficientes de transferencia de masa, que es el caso más frecuente en la práctica, es posible determinarlos realizando una prueba experimental en condiciones similares a las del equipo industrial. Con la altura del equipo experimental y los datos de operación se puede calcular el coeficiente global K'_x empleando el programa *zabKxi*. Así por ejemplo si asumimos que los datos de este ejemplo son experimentales siendo la altura del equipo 1.97 m, el coeficiente global ideal, basado en la fase líquida, se calcula con el siguiente programa:

```

«
« → Kxi
« 0.0126 0 0.012682 0.006479
  0.026 0.005 Kxi

```

```

« vy vx fun1 »
0.186
zabKxi
3 ROLL 2 DROPN
1.97 -
»
»
0.03 0.031 Seca 4 RND
"Kxi" →TAG
»

```

Haciendo correr el programa se obtiene

Kxi: .026

Que es el coeficiente individual empleado en el ejemplo. Una vez calculado el coeficiente puede ser empleado para llevar a cabo el diseño del equipo real o simular condiciones de operación diferentes.

5.4.1.10. Ejercicios

25. Repita el ejercicio 13 empleando el coeficiente global ideal promedio (K'_x), calculado con los coeficientes individuales.
26. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global ideal.
27. Repita el ejercicio 14 empleando el coeficiente global ideal promedio (K'_x), calculado con los coeficientes individuales.
28. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global ideal.
29. Repita el ejercicio 15 empleando el coeficiente global ideal promedio (K'_x), calculado con los coeficientes individuales.
30. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global ideal.

El cálculo de la altura basado en el coeficiente global real de la fase líquida (ecuación 68) es muy similar al del coeficiente global ideal. El algoritmo se presenta en la siguiente página y el código elaborado en base al mismo es el siguiente:

```

xNAME zabKx ( Altura de una torre de absorción basada en )
:: ( el coeficiente global real de la fase líquida )
%9 CKNNOLASTWD DROP ( Datos: L2=18= Flujo inicial, líquido )
CK&DISPATCH1 ( x2=17= Fracción inicial, líquido )
# 11181 ( L1=16= Flujo final, líquido )
:: ( x1=15= Fracción final, líquido )
NINE ROLL NINE ROLL ( y1=14= Fracción inicial, gas )
NINE ROLL NINE ROLL ( y2=13= Fracción final, gas )
CK&DISPATCH1 ( Kx=12= Coef. glob. real, líquido )
# 1111 ( fy=11= func. de equilibrio: x=fy[y] )
:: ( S=10= Area transversal )
NINE UNROLL NINE UNROLL ( Temporales: Lm=9; HOL=8; xa1=7; )
NINE UNROLL NINE UNROLL ( xam1=6; xa2=5; xam2=4; )
%0 NINE NDUPN DROP ( xma=3; xm=2; NOL=1 )
' NULLLAM EIGHTEEN NDUPN DOBIND
14GETLAM 11GETLAM EVAL 7PUTLAM ( xa1=fy[y1] )

```


5.4.1.11. Ejemplo

Volveremos a resolver el ejemplo 5.4.1.1 pero calculando el resultado con el coeficiente global real $K_x = 0.0263 \text{ kgmol/s.m}^3$.

Solución:

Los datos son los mismos que en el mencionado ejemplo. El programa que resuelve el problema es el siguiente:

```
« 0.0126 0 0.012682 0.006479 0.026 0.005
  0.0263
  « vy vx fun1 »
  0.186
  zabKx 3 →LIST
  { HOL NOL z } →TAG LIST→ DROP
»
```

Haciendo correr el programa se obtiene:

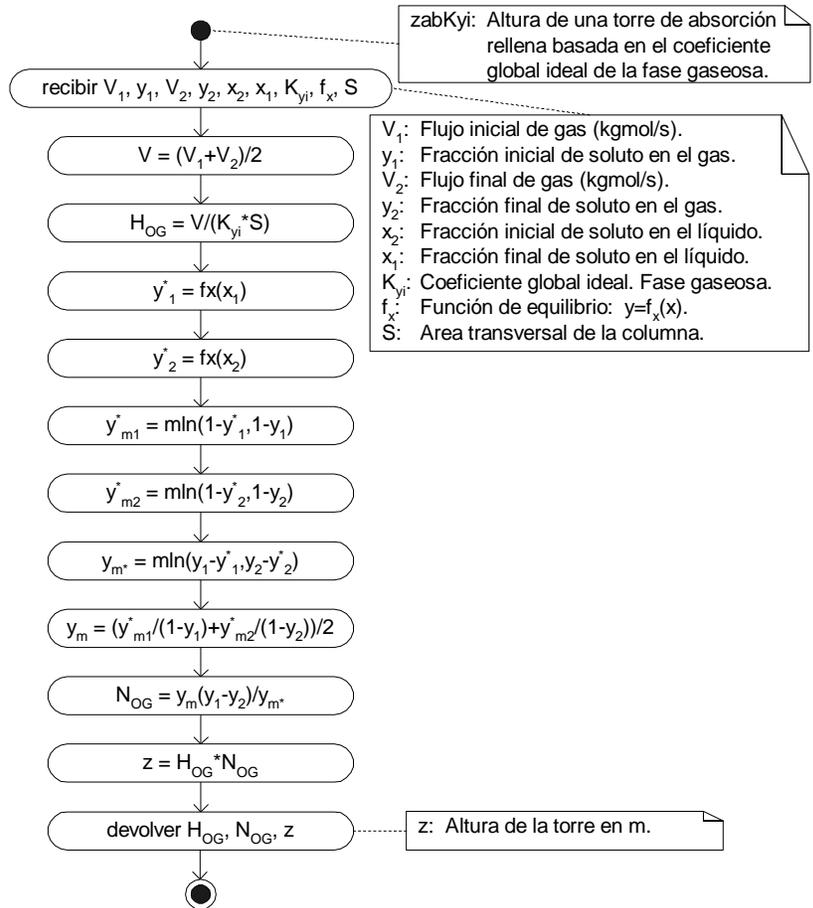
```
HOL: 2.60539430921
NOL: .753586483847
z: 1.96338993651
```

5.4.1.12. Ejercicios

31. Repita el ejercicio 13 empleando el coeficiente global real promedio (K_x), calculado con los coeficientes individuales ideales.
32. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global real.
33. Repita el ejercicio 14 empleando el coeficiente global real promedio (K_x), calculado con los coeficientes individuales ideales.
34. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global real.
35. Repita el ejercicio 15 empleando el coeficiente global real promedio (K_x), calculado con los coeficientes individuales ideales.
36. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global real.

El algoritmo para calcular la altura cuando se conoce el coeficiente global ideal basado en la fase gaseosa (ecuación 65) se presenta en la siguiente página y el código elaborado en base al mismo es el siguiente:

```
xNAME zabKyi ( Altura de una torre de absorción basada en )
::          ( el coeficiente global ideal de la fase gaseosa )
%9 CKNNOLASTWD DROP      ( Datos: V1=18= Flujo inicial, gas )
CK&DISPATCH1           ( y1=17= Fracción inicial, gas )
# 11181                 ( V2=16= Flujo final, gas )
::                     ( y2=15= Fracción final, gas )
NINE ROLL NINE ROLL    ( x2=14= Fracción inicial, líquido )
NINE ROLL NINE ROLL    ( x1=13= Fracción final, líquido )
CK&DISPATCH1         ( Kyi=12= Coef. glob. ideal, gas )
# 1111                 ( fx=11= func. de equilibrio: y=fx[x] )
::                     ( S=10= Area transversal )
    NINE UNROLL NINE UNROLL ( Temporales: V=9; HOG=8; ya1=7; )
    NINE UNROLL NINE UNROLL (          yam1=6; ya2=5; yam2=4; )
```



```

%0 NINE NDUPN DROP ( yma=3; ym=2; NOG=1 )
' NULLLAM EIGHTEEN NDUPN DOBIND
18GETLAM 16GETLAM %+ %2 %/ 9PUTLAM ( V=[V1+V2]/2 )
9GETLAM 12GETLAM 10GETLAM %* %/
8PUTLAM ( HOG=V/[Kyi*S] )
13GETLAM 11GETLAM EVAL 7PUTLAM ( ya1=fx[x1] )
14GETLAM 11GETLAM EVAL 5PUTLAM ( ya2=fx[x2] )
%1 7GETLAM %- %1 17GETLAM %- mln ( mln[1-ya1,1-y1] )
6PUTLAM ( => yam1 )
%1 5GETLAM %- %1 15GETLAM %- mln ( mln[1-ya2,1-y2] )
4PUTLAM ( => yam2 )
17GETLAM 7GETLAM %- 15GETLAM
5GETLAM %- mln 3PUTLAM ( yma=mln[y1-ya1,y2-ya2] )
6GETLAM %1 17GETLAM %- %/ 4GETLAM ( [yam1/[1-y1]+yam2 )
%1 15GETLAM %- %/ %+ %2 %/ ( /[1-y2]]/2 )
2PUTLAM ( => ym )
2GETLAM 17GETLAM 15GETLAM %- %* ( ym*[y1-y2] )
3GETLAM %/ 1PUTLAM ( /yma => NOG )
8GETLAM 1GETLAM 2DUP %* ( HOG, NOG, z )
ABND
;
ZERO
::
NINE UNROLL NINE UNROLL
NINE UNROLL NINE UNROLL
# 202 DO#EXIT ( 202 = Bad argument type )
;
  
```

;
;

5.4.1.13. Ejemplo

Volveremos a resolver el ejemplo 5.4.1.1 pero calculando el resultado con el coeficiente global ideal $K'_y = 0.02178 \text{ kgmol/s.m}^3$.

Solución:

Los datos son los mismos que en el mencionado ejemplo. El programa que resuelve el problema es el siguiente:

```
« 0.003893 0.026 0.003811 0.005 0 0.006479
0.02178
« vx vy fun1 »
0.186
zabKy1 3 →LIST
{ HOG NOG z } →TAG LIST→ DROP
»
```

Haciendo correr el programa se obtiene:

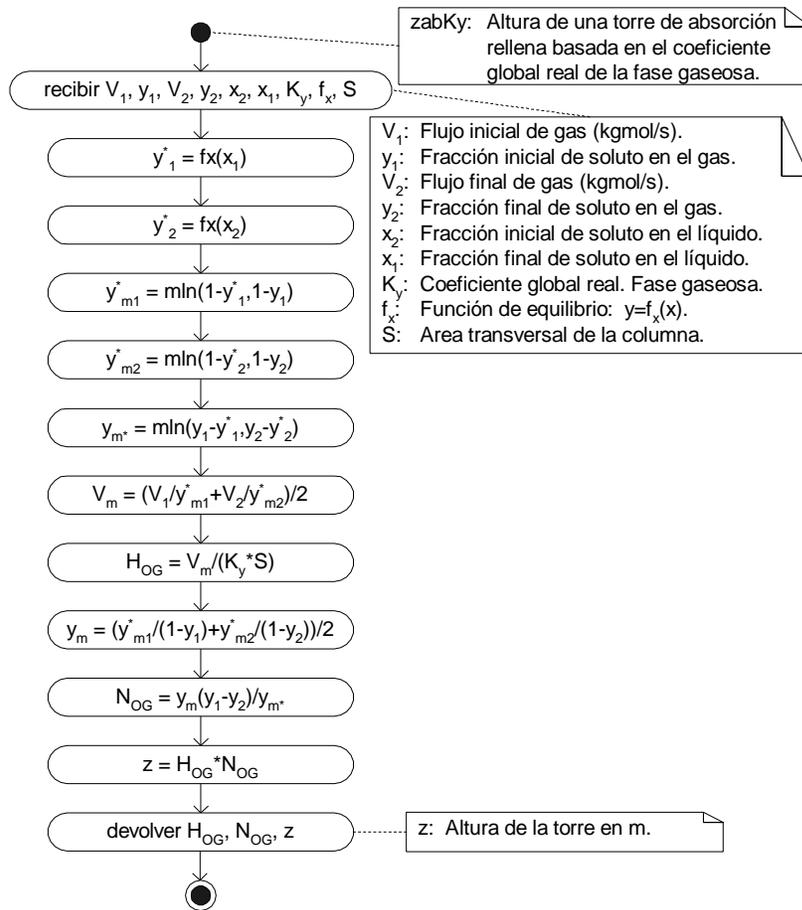
```
HOG: .950857549098
NOG: 2.06417869998
z: 1.96273989956
```

5.4.1.14. Ejercicios

37. Repita el ejercicio 13 empleando el coeficiente global ideal promedio (K'_y), calculado con los coeficientes individuales ideales.
38. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global ideal de la fase gaseosa.
39. Repita el ejercicio 14 empleando el coeficiente global ideal promedio (K'_y), calculado con los coeficientes individuales ideales.
40. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global ideal de la fase gaseosa.
41. Repita el ejercicio 15 empleando el coeficiente global ideal promedio (K'_y), calculado con los coeficientes individuales ideales.
42. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global ideal de la fase gaseosa.

El algoritmo para calcular la altura de la torre empleando el coeficiente global real de la fase gaseosa (ecuación 69) se presenta en el diagrama de actividades de la siguiente página y el código respectivo es el siguiente:

```
xNAME zabKy ( Altura de una torre de absorción basada en )
::          ( el coeficiente global real de la fase gaseosa )
%9 KKNOLASTWD DROP      ( Datos: V1=18= Flujo inicial, gas )
CK&DISPATCH1          ( y1=17= Fracción inicial, gas )
# 11181                ( V2=16= Flujo final, gas )
::                    ( y2=15= Fracción final, gas )
NINE ROLL NINE ROLL    ( x2=14= Fracción inicial, líquido )
NINE ROLL NINE ROLL    ( x1=13= Fracción final, líquido )
CK&DISPATCH1          ( Ky=12= Coef. glob. real, gas )
# 1111                ( fx=11= func. de equilibrio: y=fx[x] )
```



```

:: ( S=10= Area transversal )
NINE UNROLL NINE UNROLL ( Temporales: Vm=9; HOG=8; ya1=7; )
NINE UNROLL NINE UNROLL ( ya1=6; ya2=5; yam2=4; )
%0 NINE NDUPN DROP ( yma=3; ym=2; NOG=1 )
' NULLLAM EIGHTEEN NDUPN DOBIND
13GETLAM 11GETLAM EVAL 7PUTLAM ( ya1=fx[x1] )
14GETLAM 11GETLAM EVAL 5PUTLAM ( ya2=fx[x2] )
%1 7GETLAM %- %1 17GETLAM %- mln ( mln[1-ya1,1-y1] )
6PUTLAM ( => yam1 )
%1 5GETLAM %- %1 15GETLAM %- mln ( mln[1-ya2,1-y2] )
4PUTLAM ( => yam2 )
17GETLAM 7GETLAM %- 15GETLAM
5GETLAM %- mln 3PUTLAM ( yma=mln[y1-ya1,y2-ya2] )

18GETLAM 6GETLAM %/ 16GETLAM
4GETLAM %/ %+ %2 %/ 9PUTLAM ( Vm=[V1/yam1+V2/yam2]/2 )
9GETLAM 12GETLAM 10GETLAM %* %/
8PUTLAM ( HOG=Vm/[Ky*S] )
6GETLAM %1 17GETLAM %- %/ 4GETLAM ( [yam1/[1-y1]+yam2 )
%1 15GETLAM %- %/ %+ %2 %/ ( /[1-y2]]/2 )
2PUTLAM ( => ym )
2GETLAM 17GETLAM 15GETLAM %- %* ( ym*[y1-y2] )
3GETLAM %/ 1PUTLAM ( /yma => NOG )
8GETLAM 1GETLAM 2DUP %* ( HOG, NOG, z )
ABND

;
ZERO

```

```

::
  NINE UNROLL NINE UNROLL
  NINE UNROLL NINE UNROLL
  # 202 DO#EXIT                    ( 202 = Bad argument type )
;
;
;

```

5.4.1.15. Ejemplo

Volveremos a resolver el ejemplo 5.4.1.1 pero calculando el resultado con el coeficiente global real $K_y = 0.02199 \text{ kgmol/s.m}^3$.

Solución:

Los datos son los mismos que en el mencionado ejemplo. El programa que resuelve el problema es el siguiente:

```

« 0.003893 0.026 0.003811 0.005 0 0.006479
  0.02178 « vx vy fun1 » 0.186
  zabKy 3 →LIST
  { HOG NOG z } →TAG LIST→ DROP
»

```

Haciendo correr el programa se obtiene:

```

HOG: .951128212176
NOG: 2.06417869998
z: 1.96329859652

```

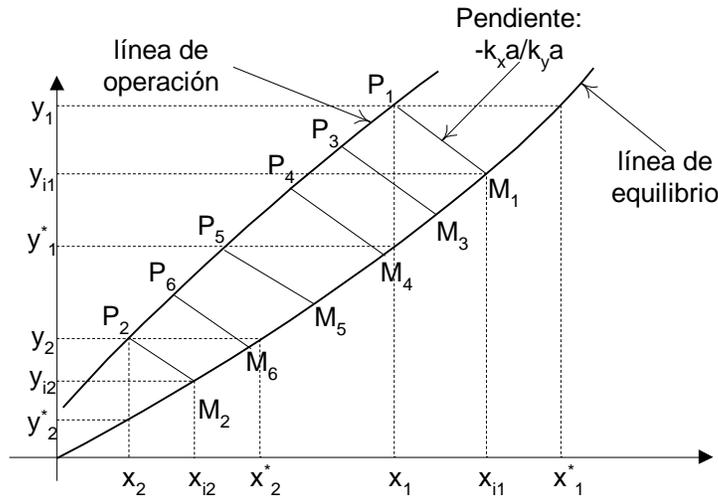
5.4.1.16. Ejercicios

43. Repita el ejercicio 13 empleando el coeficiente global real promedio (K_y), calculado con los coeficientes individuales ideales.
44. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global real de la fase gaseosa.
45. Repita el ejercicio 14 empleando el coeficiente global real promedio (K_y), calculado con los coeficientes individuales ideales.
46. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global real de la fase gaseosa.
47. Repita el ejercicio 15 empleando el coeficiente global real promedio (K_y), calculado con los coeficientes individuales ideales.
48. Empleando la altura calculada en el anterior ejercicio calcule el coeficiente global real de la fase gaseosa.

5.4.2. Cálculo de la altura para mezclas gaseosas concentradas

Cuando las soluciones son concentradas la altura se determina resolviendo las integrales de las ecuaciones (34) a (37) o (42) a (45). No obstante es necesario recalcar que aún empleando estas ecuaciones el cálculo es simplificado (no riguroso) pues no se toman en cuenta una serie de factores como ser la variación de temperatura a lo largo de la columna, la pérdida de calor, la dependencia de los coeficientes de transferencia de la composición, la absorción del gas inerte en el líquido, la absorción del líquido inerte en el gas, la caída de presión, etc.

El procedimiento gráfico que se sigue cuando se conocen los coeficientes volumétricos individuales de transferencia de masa es el siguiente (vea la figura):

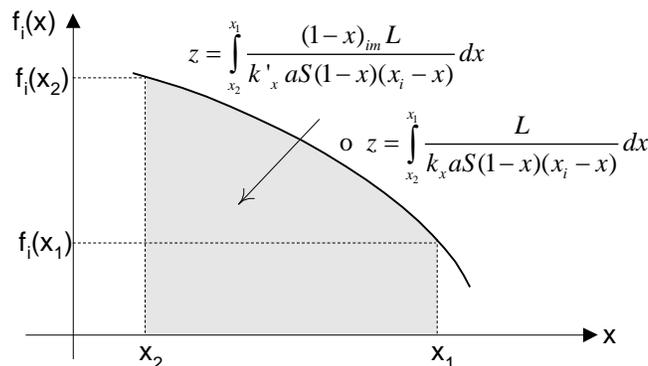


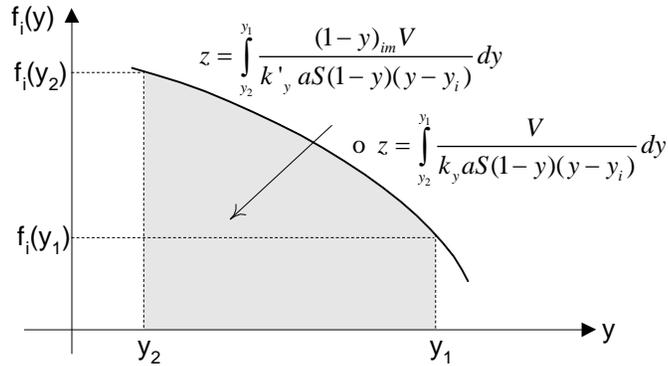
- En una gráfica "x" (fracción molar de soluto en el líquido) versus "y" (fracción molar de soluto en el gas), se traza la curva de equilibrio empleando los datos o ecuación de equilibrio.
- Con la ecuación (12) (o la ecuación 13) se traza la línea de operación, que generalmente no es una línea recta.
- Se eligen los puntos inicial y final de la torre y al menos cuatro puntos intermedios y para cada uno de ellos se trazan líneas de pendiente $(-k_x a / k_y a)$ ubicando los puntos "M" (composición de la interfaz gas-líquido), siguiendo el procedimiento descrito en el acápite 5.3.1 o 5.3.3, según se conozcan los coeficientes reales o ideales.
- Con las composiciones de la interfaz gas-líquido (puntos M), se calcula, para cada punto, los valores de la función a integrar:

$$f_i(x) = \frac{(1-x)_{im} L}{k'_x a S (1-x)(x_i - x)} \quad \text{o} \quad f_i(x) = \frac{L}{k'_x a S (1-x)(x_i - x)}$$

$$f_i(y) = \frac{(1-y)_{im} V}{k'_y a S (1-y)(y - y_i)} \quad \text{o} \quad f_i(y) = \frac{V}{k'_y a S (1-y)(y - y_i)}$$

- Con los valores de las funciones calculados en el inciso anterior se traza la gráfica " $f_i(x)$ " versus "x" o " $f_i(y)$ " versus "y" y se mide el área bajo la curva determinando así la altura.





Frecuentemente además, los coeficientes volumétricos de transferencia de masa son funciones de los flujos molares por lo que deben ser calculados para cada uno de los puntos, por lo que el proceso gráfico resulta bastante moroso.

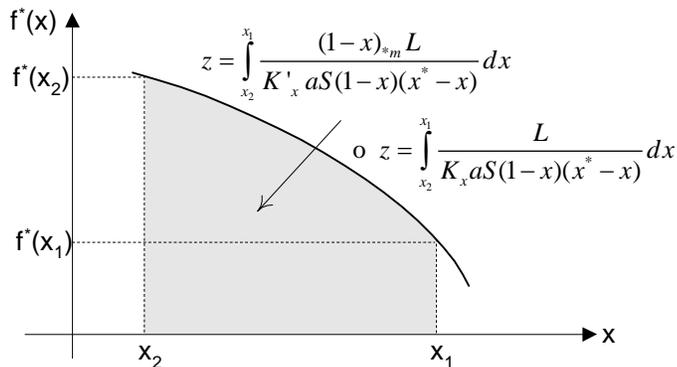
El procedimiento gráfico que se sigue cuando se conocen los coeficientes volumétricos globales de transferencia de masa es el siguiente:

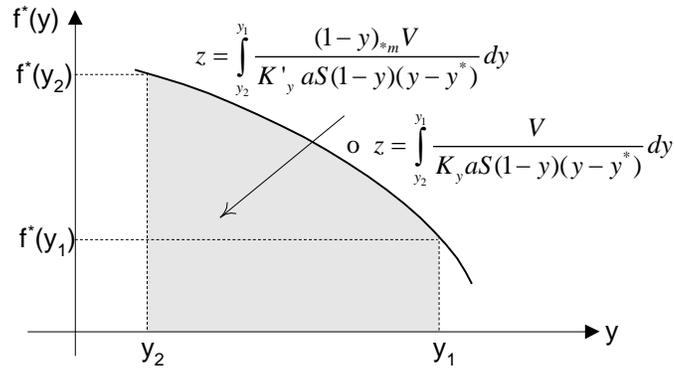
- a) En una gráfica "x" (fracción molar de soluto en el líquido) versus "y" (fracción molar de soluto en el gas), se traza la curva de equilibrio empleando los datos o ecuación de equilibrio.
- b) Se toman los puntos inicial y final de la torre y al menos cuatro puntos intermedios y para cada uno de ellos se determinan los valores de "x*" y "y*" que están en equilibrio con los correspondientes valores "x" y "y" en cada uno de los puntos considerados.
- c) Con los valores calculados en el anterior inciso se determinan los valores de la función a integrar para cada uno de los puntos:

$$f_i(x) = \frac{(1-x)_{*m} L}{K'_x a S (1-x)(x^* - x)} \quad \text{o} \quad f_i(x) = \frac{L}{K_x a S (1-x)(x^* - x)}$$

$$f_i(y) = \frac{(1-y)_{*m} V}{K'_y a S (1-y)(y - y^*)} \quad \text{o} \quad f_i(y) = \frac{V}{K_y a S (1-y)(y - y^*)}$$

- d) Con los valores de la función a integrar, calculados en el anterior inciso, se trazan las gráficas "f*(x)" versus "x" (o "f*(y)" versus "y"), como se muestra en las siguientes figuras, entonces se mide el área bajo la curva, determinándose de esta manera la altura de la torre.





Puesto que los procedimientos gráficos son morosos y tediosos, es conveniente automatizarlos mediante programas de computadora. La integral puede ser resuelta con los programas de integración "Trapf" y "Simpf" presentados en el capítulo 3. No obstante, es posible también emplear los valores de las funciones calculados en el inciso (d) o (c) e integrarlos directamente con los métodos del Trapecio y Simpson en su forma tabular.

A continuación se presentan los programas que integran directamente datos tabulados por el método del Trapecio (Trapt) y el método de Simpson (Simpt), los mismos que deberán ser añadidos al archivo Metodos1.s del proyecto PRQ205. No se entra en detalle con relación a estos métodos pues han sido estudiados en otra materia.

```
xNAME Trapt ( Integración de datos tabulados con el método del Trapecio)
::          ( Datos:  x= Vector con los datos independientes )
CK2&Dispatch          ( y= Vector con los datos dependientes )
# 44                  ( Locales: n= N° de elementos )
::                  (          s= Sumatoria )
OVER MDIMSDROP %0
SWAP ONE
DO
ROT INDEX@ #1+ PULLREALEL ( x[i+1] )
SWAP INDEX@ PULLREALEL ( x[i] )
SWAP 5UNROLL %- ( - )
ROT INDEX@ PULLREALEL ( y[i] )
SWAP INDEX@ #1+ PULLREALEL ( y[i+1] )
SWAP 5UNROLL %+ %* %2 %/ ( + * 2 / )
%+ ( + )
LOOP
3UNROLL 2DROP
;
;

xNAME Simpt ( Integración de datos tabulados con el método de Simpson )
::          ( Datos: x= Vector con los datos independientes )
CK2&Dispatch          ( y= Vector con los datos dependientes )
#44                  ( Locales: n=13; i=12; j=11; k=10; s=9; dxij=8; )
::          ( dxij2=7; dxik=6; dxik2=5; )
OVER MDIMSDROP #1- BINT1 ( dyij=4; a=3; b=2; c=1 )
BINT2 BINT3 %0 NINE NDUPN DROP
' NULLLAM THIRTEEN NDUPN DOBIND
13GETLAM TWO #/ DROP #0<> ( n es impar? )
IT
::
coefs
9GETLAM 3GETLAM ( s a )
```

```

4ROLL 11GETLAM PULLREALEL %3 %^ SWAP      ( x[j] 3 ^ )
12GETLAM PULLREALEL %3 %^ SWAP 6UNROLL    ( x[i] 3 ^ )
%- %* %3 %/ %+ 2GETLAM 7GETLAM %*        ( - * 3 / + b dxij2 * )
%2 %/ %- 1GETLAM 8GETLAM %* %- 9PUTLAM   ( 2 / c dxij * - => s )
12GETLAM #1+ 12PUTLAM                      ( i=i+1 )
11GETLAM #1+ 11PUTLAM                      ( j=j+1 )
10GETLAM #1+ 10PUTLAM                      ( k=k+1 )
;
13GETLAM #1+ 13PUTLAM                      ( n=n+1 )
BEGIN
  coefs
  9GETLAM 3GETLAM                          ( s a )
  4ROLL 10GETLAM PULLREALEL %3 %^ SWAP    ( x[k] 3 ^ )
  12GETLAM PULLREALEL %3 %^ SWAP 6UNROLL  ( x[i] 3 ^ )
  %- %* %3 %/ %+ 2GETLAM 5GETLAM %*      ( - * 3 / + b dxik2 * )
  %2 %/ %- 1GETLAM 6GETLAM %* %- 9PUTLAM  ( 2 / c dxij * - => s )
  12GETLAM #2+ 12PUTLAM                    ( i=i+1 )
  11GETLAM #2+ 11PUTLAM                    ( j=j+1 )
  10GETLAM #2+ 10PUTLAM                    ( k=k+1 )
  12GETLAM 13GETLAM #=                     ( i=n? )
UNTIL
2DROP 9GETLAM                              ( s )
ABND
;
;
NULLNAME coefs ( Coeficientes para el método de Simpson tabular )
::
SWAP 11GETLAM PULLREALEL SWAP 12GETLAM
PULLREALEL ROT %- 8PUTLAM                  ( x[i] x[j] - => dxij )
11GETLAM PULLREALEL %2 %^ SWAP 12GETLAM
PULLREALEL %2 %^ ROT %- 7PUTLAM           ( x[i] 2 ^ x[j] 2 ^ - => dxij2 )
10GETLAM PULLREALEL SWAP 12GETLAM
PULLREALEL ROT %- 6PUTLAM                  ( x[i] x[k] - => dxik )
10GETLAM PULLREALEL %2 %^ SWAP 12GETLAM
PULLREALEL %2 %^ ROT %- 5PUTLAM           ( x[i] 2 ^ x[k] 2 ^ - => dxik2 )
SWAP 11GETLAM PULLREALEL SWAP 12GETLAM
PULLREALEL ROT %- 4PUTLAM                  ( y[i] y[j] - => dyij )
10GETLAM PULLREALEL SWAP 12GETLAM
PULLREALEL ROT %- 8GETLAM %* 4GETLAM      ( y[i] y[k] - dxij * dyij )
6GETLAM %* %- 5GETLAM 8GETLAM %*         ( dxik * - dxik2 dxij * )
7GETLAM 6GETLAM %* %- %/ 3PUTLAM         ( dxij2 dxik * - / => a )
4GETLAM 3GETLAM 7GETLAM %* %- 8GETLAM    ( dyij a dxij2 * - dxij / )
%/ 2PUTLAM                                 ( / => b )
12GETLAM PULLREALEL ROT 12GETLAM          ( y[i] )
PULLREALEL SWAP 4UNROLL DUP %2 %^        ( x[i] 2 ^ )
3GETLAM %* SWAP 2GETLAM %* %+ %- 1PUTLAM ( a * b x[i] * + - => c )
;

```

Probaremos estos programas calculando, con los datos tabulados que se presentan a continuación, el valor de la integral:

$$\int_{2.5}^{7.2} f(x)dx = \int_{2.5}^{7.2} ydx$$

x	y
2.0	4.800
2.5	6.625

2.7	7.509
3.0	9.000
3.3	10.689
3.8	13.944
4.2	16.944
4.5	19.425
5.1	24.981
5.7	31.329
6.0	34.800
6.4	39.736
6.9	46.401
7.2	50.664
7.5	55.125

El programa que calcula el valor de la integral con el método del trapecio es:

```
« [ 2.5 2.7 3.0 3.3 3.8 4.2 4.5 5.1 5.7 6.0 6.4 6.9 7.2 ]
  [ 6.625 7.509 9 10.689 13.944 16.944 19.425 24.981 31.329 34.8
    39.736 46.401 50.664]
  Trapt "Integral" →TAG
»
```

Haciendo correr el programa se obtiene:

Integral: 115.76965

Para el método de Simpson, sólo se cambia *Trapt* por *Trapf*:

```
« [ 2.5 2.7 3.0 3.3 3.8 4.2 4.5 5.1 5.7 6.0 6.4 6.9 7.2 ]
  [ 6.625 7.509 9 10.689 13.944 16.944 19.425 24.981 31.329 34.8
    39.736 46.401 50.664]
  Simpt "Integral" →TAG
»
```

Haciendo correr el programa se obtiene:

Integral: 115.594933333

Si en lugar de *Trapt* y *Simpt* empleamos *Trapf* y *Simpf* la solución sería:

```
« « [ 2.5 2.7 3.0 3.3 3.8 4.2 4.5 5.1 5.7 6.0 6.4 6.9 7.2 ]
  [ 6.625 7.509 9 10.689 13.944 16.944 19.425 24.981 31.329 34.8
    39.736 46.401 50.664] fun1 » 2.5 7.2 20
  Trapf "Integral" →TAG
»
```

Programa con el que se obtiene:

Integral: 115.81900975

Y para el método de *Simpson*:

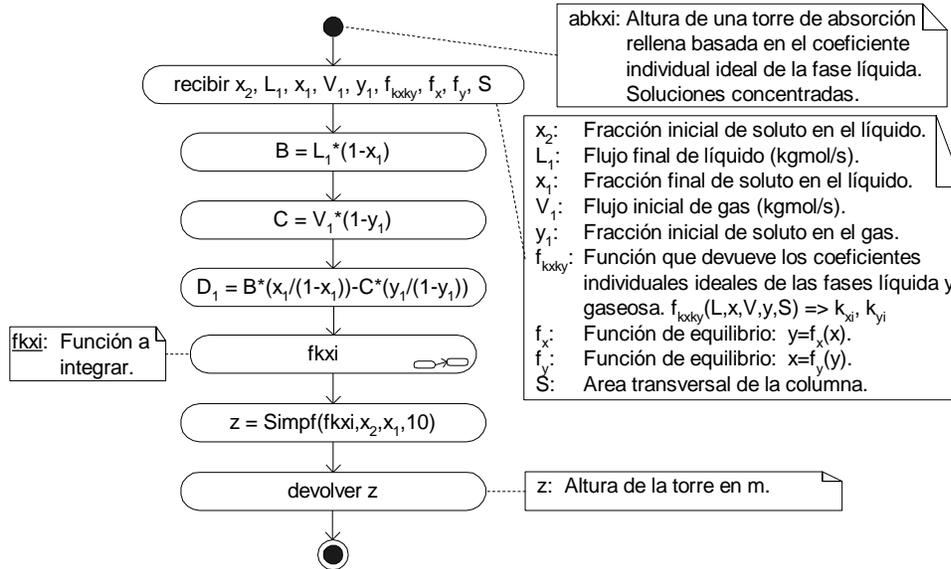
```
« « [ 2.5 2.7 3.0 3.3 3.8 4.2 4.5 5.1 5.7 6.0 6.4 6.9 7.2 ]
  [ 6.625 7.509 9 10.689 13.944 16.944 19.425 24.981 31.329 34.8
    39.736 46.401 50.664] fun1 » 2.5 7.2 20
  Simpf "Integral" →TAG
»
```

Con el que se obtiene:

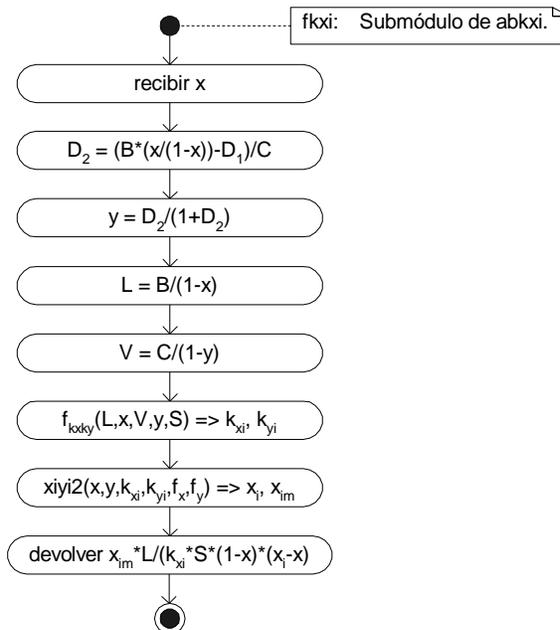
Integral: 115.762269

En la resolución numérica de las ecuaciones (34) a (37) o (42) a (45) se puede emplear cualquiera de las dos formas de integración, no obstante, la forma tabular es más conveniente cuando se sigue el procedimiento gráfico.

Como ya se dijo, los coeficientes de transferencia de masa suelen variar con los flujos y composición de las fases gaseosa o líquida, por lo que en el caso más general se debe contar con una función que devuelva los valores de los coeficientes de transferencia de masa en base a los flujos y composiciones de las fases gaseosa y líquida. El algoritmo para resolver la ecuación (34) es el siguiente:



Donde el algoritmo de la función "fkxi" es el siguiente:



El código elaborado en base a estos algoritmos es:

```

xNAME abkxi ( Altura de una torre de absorción basada en el coeficiente )
::          ( individual ideal de la fase líquida. Soluciones concentradas )
%9 CKNNOLASTWD DROP ( Datos: x2= Fracción inicial, fase líquida )
CK&DISPATCH1      ( L1= Flujo final, fase líquida )
# 18881            ( x1= Fracción final, fase líquida )
::                ( V1= Flujo inicial, fase gaseosa )
NINE ROLL NINE ROLL ( y1= Fracción inicial, fase gaseosa )
    
```

```

NINE ROLL NINE ROLL      ( fkxky= Función que devuelve kxi,kyi )
CK&DISPATCH1           ( fx= Función de equilibrio, y=fx[x] )
# 1111                   ( fy= Función de equilibrio, x=fy[y] )
::                        ( S= Area transversal de la columna )
  NINE UNROLL NINE UNROLL ( Locales: B, C, D1, D2, x, y, L, V, )
  NINE UNROLL NINE UNROLL      ( kxi, kyi, xi, xim )
  %0 TWELVE NDUPN DROP
  { LAM x2 LAM L1 LAM x1 LAM V1
    LAM y1 LAM fkxky LAM fx LAM fy
    LAM S LAM B LAM C LAM D1 LAM D2
    LAM x LAM y LAM L LAM V LAM kxi
    LAM kyi LAM xi LAM xim }
  BIND
  LAM L1 %1 LAM x1 %- %* ' LAM B STO
  LAM V1 %1 LAM y1 %- %* ' LAM C STO
  LAM B LAM x1 %1 LAM x1 %- %/ %*
  LAM C LAM y1 %1 LAM y1 %- %/ %* %- ' LAM D1 STO
  '
  :: ( función a integrar: fkxi )
    ' LAM x STO
    LAM B LAM x %1 LAM x %- %/ %* LAM D1 %- LAM C %/ ' LAM D2 STO
    LAM D2 %1 LAM D2 %+ %/ ' LAM y STO
    LAM B %1 LAM x %- %/ ' LAM L STO
    LAM C %1 LAM y %- %/ ' LAM V STO
    LAM L LAM x LAM V LAM y LAM S LAM fkxky EVAL
    ' LAM kyi STO ' LAM kxi STO
    LAM x LAM y LAM kxi LAM kyi LAM fx LAM fy
    xxiyi2 3DROP ' LAM xim STO DROP ' LAM xi STO
    LAM xim LAM L %* LAM kxi LAM S %* %1 LAM x
    %- %* LAM xi LAM x %- %* %/
  ;
  LAM x2 LAM x1 %10 xSimpf
  ABND
;
ZERO
::
  NINE UNROLL NINE UNROLL
  NINE UNROLL NINE UNROLL
  # 202 DO#EXIT ( 202 = Bad argument type )
;
;
;

```

5.4.2.1. Ejemplo

En una industria se empleará una torre empacada con anillos cerámicos de 25.4 mm para absorber SO₂ del aire mediante agua pura a 293 K y 1.01325x10⁵ Pa de presión absoluta. El gas de entrada contiene 20 mol% de SO₂ y el de salida 2 mol%. El flujo de aire inerte es 6.53x10⁻⁴ kgmol/s y el flujo de agua inerte es 4.20x10⁻² kgmol/s. El área transversal de la torre es 0.0929 m². Los coeficientes ideales de transferencia de masa a 293 K para anillos de 25.4 mm (1 pulgada) pueden ser calculados con las siguientes relaciones:

$$k'_{y,a} = 0.0594 G_y^{0.7} G_x^{0.25} \quad k'_{x,a} = 0.152 G_x^{0.82}$$

Donde $k'_{y,a}$ y $k'_{x,a}$ están en kgmol/s.m³ y G_x y G_y son los flujos totales de líquido y gas respectivamente en kg/s.m². Calcule la altura de la torre basada en el coeficiente de la fase líquida. Los datos de equilibrio para este sistema son los del ejercicio 13.

Solución

Los datos con los que contamos en este problema son los siguientes:

$x_2 = 0.0$
 $y_1 = 0.2$
 $y_2 = 0.02$
 $C = 6.53 \times 10^{-4} \text{ kgmol/s}$
 $B = L_2 = 4.2 \times 10^{-2} \text{ kgmol/s}$
 $S = 0.0929 \text{ m}^2$
 $P = 1.013 \times 10^5 \text{ Pa} = 1.01325 \text{E}5 \text{ Pa} \text{ 1_mmHg CONVERT} = 760 \text{ mmHg}$

Contamos además con las correlaciones para el cálculo de los coeficientes individuales ideales:

$$k'_y a = 0.0594 G_y^{0.7} G_x^{0.25} \quad k'_x a = 0.152 G_x^{0.82}$$

Y los datos de equilibrio:

```
[ 0 0.0000562 0.0001403 0.000280 0.000422 0.000564 0.000842 0.001403
0.001965 0.00279 0.00420 0.00698 0.01385 0.0206 0.0273] 'vx' STO
[ 0 0.5 1.2 3.2 5.8 8.5 14.1 26 39 59 92 161 336 517 698 ] 760 / 'vy' STO
```

Los flujos iniciales y finales del gas son:

$V_1 = C / (1 - y_1) = 6.53 \text{E-}4 / 0.2 = 0.00081625 \text{ kgmol/s}$
 $V_2 = C / (1 - y_2) = 6.53 \text{E-}4 / 0.02 = 6.66326530612 \text{E-}4 \text{ kgmol/s}$

Entonces el porcentaje de soluto absorbido (ecuación 4) es:

$p_a = (V_1 y_1 - V_2 y_2) / V_1 y_1$
 $= 0.00081625 \cdot 0.2 - 6.66326530612 \text{E-}4 \cdot 0.02 / 0.00081625 \cdot 0.2 \cdot 100$
 $= 91.836734694 \%$

Los flujos y composiciones faltantes pueden ser calculados con *abrc1*:

```
« 4.2E-2 0.0 0.00081625 0.2 91.836734694
abrc1 2 DROPN 2 →LIST
{ L1 x1 } →TAG LIST→ DROP
»
```

Haciendo correr el programa obtenemos el flujo y composición faltante:

$L_1: 4.21499234694 \text{E-}2$
 $x_1: 3.55690964653 \text{E-}3$

Las correlaciones para el cálculo de los coeficientes individuales deben ser escritas en función de los flujos y composiciones de las dos fases, ello se consigue tomando en cuenta que el flujo total de líquido (G_x) es la suma del flujo de agua ($L \cdot (1-x)$) y el de soluto ($L \cdot x$), recordando además que el peso molecular del agua es 18.02 y el del soluto (SO_2) 64.1, siendo la sección transversal de la torre S :

$$G_x = \frac{L \cdot (1-x) \cdot 18.02 + L \cdot x \cdot 64.1}{S}$$

De igual manera flujo total de gas (G_y) es la suma del flujo de aire ($V \cdot (1-y)$) y el flujo de soluto ($V \cdot y$), siendo el peso molecular medio del aire 28.97:

$$G_y = \frac{V \cdot (1-y) \cdot 28.97 + V \cdot y \cdot 64.1}{S}$$

Ahora contamos con todos los datos para resolver el problema con *abkxi*:

```
« 0.0 4.21499234694E-2 3.55690964653E-3 0.00081625 0.2
```

```

« 0 0 → L x V y S Gx Gy
« L 1 x - * 18.02 * L x * 64.1 * + S / 'Gx' STO
  V 1 y - * 28.97 * V y * 64.1 * + S / 'Gy' STO
    0.152 Gx 0.82 ^ *
    0.0594 Gy 0.7 ^ * Gx 0.25 ^ *
  »
»
« vx vy fun1 »
« vy vx fun1 »
0.0929
abkxi
"z" →TAG
»

```

Haciendo correr el programa se obtiene la altura de la torre en aproximadamente 51 segundos:

z: 1.57228254565

5.4.2.2. Ejercicios

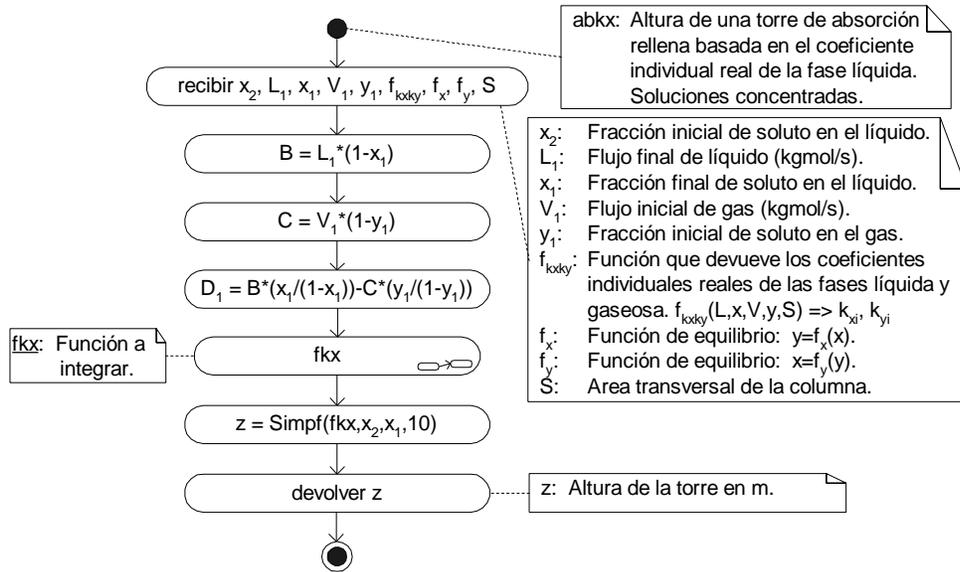
49. Repita el ejemplo 5.4.2.1 considerando que el porcentaje molar inicial de soluto es 25% y el final 1.5%.
50. Repita el ejemplo 5.4.2.1 considerando que el porcentaje de soluto absorbido es del 96%.
51. Repita el ejemplo 5.4.2.1 considerando que el porcentaje absorbido es del 95% y que los datos de equilibrio siguen la ley de *Henry*: $y = 0.0234x$.
52. Repita el ejemplo 5.4.2.1 considerando que el porcentaje absorbido es del 97% y que los coeficientes individuales de transferencia de masa son constantes e iguales a: $k'_{xa} = 0.85 \text{ kgmol/s.m}^3$; $k'_{ya} = 0.04 \text{ kgmol/s.m}^3$.

El cálculo de la altura basado en el coeficiente individual real de la fase líquida es muy similar al del coeficiente ideal. El algoritmo que resuelve la ecuación (42) se presenta en la siguiente página y el código elaborado en base al mismo es el siguiente:

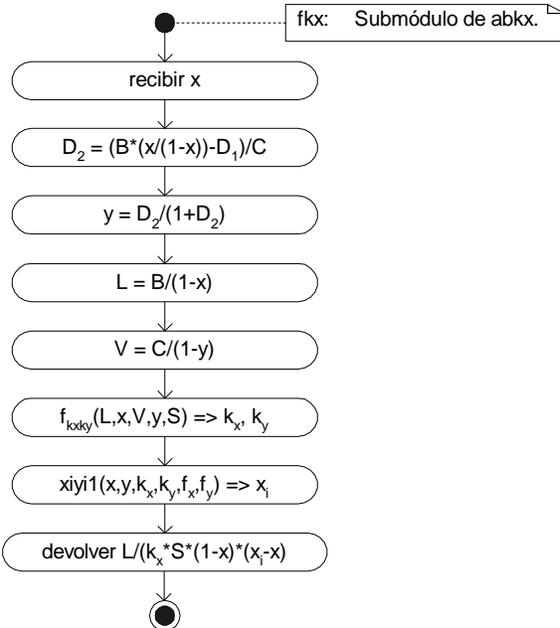
```

xNAME abkx ( Altura de una torre de absorción basada en el coeficiente )
:: ( individual real de la fase líquida. Soluciones concentradas )
%9 CKNNOLASTWD DROP ( Datos: x2= Fracción inicial, fase líquida )
CK&DISPATCH1 ( L1= Flujo final, fase líquida )
# 18881 ( x1= Fracción final, fase líquida )
:: ( V1= Flujo inicial, fase gaseosa )
NINE ROLL NINE ROLL ( y1= Fracción inicial, fase gaseosa )
NINE ROLL NINE ROLL ( fkxky= Función que devuelve kx,ky )
CK&DISPATCH1 ( fx= Función de equilibrio, y=fx[x] )
# 1111 ( fy= Función de equilibrio, x=fy[y] )
:: ( S= Area transversal de la columna )
NINE UNROLL NINE UNROLL ( Locales: B, C, D1, D2, x, y, L, V, )
NINE UNROLL NINE UNROLL ( kx, ky, xi )
%0 ELEVEN NDUPN DROP
{ LAM x2 LAM L1 LAM x1 LAM V1
  LAM y1 LAM fkxky LAM fx LAM fy
  LAM S LAM B LAM C LAM D1 LAM D2
  LAM x LAM y LAM L LAM V LAM kx
  LAM ky LAM xi }

```



Donde el algoritmo de la función *f_kx* es:



```

BIND
LAM L1 %1 LAM x1 %- %* ' LAM B STO
LAM V1 %1 LAM y1 %- %* ' LAM C STO
LAM B LAM x1 %1 LAM x1 %- %/ %*
LAM C LAM y1 %1 LAM y1 %- %/ %* %- ' LAM D1 STO
'
:: ( función a integrar: f_kx )
' LAM x STO
LAM B LAM x %1 LAM x %- %/ %* LAM D1 %- LAM C %/ ' LAM D2 STO
LAM D2 %1 LAM D2 %+ %/ ' LAM y STO
LAM B %1 LAM x %- %/ ' LAM L STO
LAM C %1 LAM y %- %/ ' LAM V STO
LAM L LAM x LAM V LAM y LAM S LAM f_kxky EVAL
' LAM ky STO ' LAM kx STO
LAM x LAM y LAM kx LAM ky LAM fx LAM fy
    
```

```

      xxiyil DROP ' LAM xi STO
      LAM L LAM kx LAM S %* %1 LAM x
      %- %* LAM xi LAM x %- %* %/
    ;
    LAM x2 LAM x1 %10 xSimpf
    ABND
  ;
  ZERO
  ::
  NINE UNROLL NINE UNROLL
  NINE UNROLL NINE UNROLL
  # 202 DO#EXIT ( 202 = Bad argument type )
;
;
;

```

5.4.2.3. Ejemplo

Volveremos a resolver el ejemplo 5.4.2.1 pero asumiendo que las correlaciones para los coeficientes individuales son:

$$k_y a = 0.0599 G_y^{0.7} G_x^{0.25} \quad k_x a = 0.159 G_x^{0.82}$$

Solución:

Con excepción de las correlaciones para los coeficientes individuales, los datos son los mismos que en el mencionado ejemplo, por lo tanto el programa que resuelve el problema es también prácticamente el mismo:

```

« 0.0 4.21499234694E-2 3.55690964653E-3 0.00081625 0.2
« 0 0 → L x V y S Gx Gy
« L 1 x - * 18.02 * L x * 64.1 * + S / 'Gx' STO
  V 1 y - * 28.97 * V y * 64.1 * + S / 'Gy' STO
  0.159 Gx 0.82 ^ *
  0.0599 Gy 0.7 ^ * Gx 0.25 ^ *
»
»
« vx vy fun1 »
« vy vx fun1 »
0.0929
abkx
"z" →TAG
»

```

Haciendo correr el programa se obtiene la altura de la torre en aproximadamente 34 segundos:

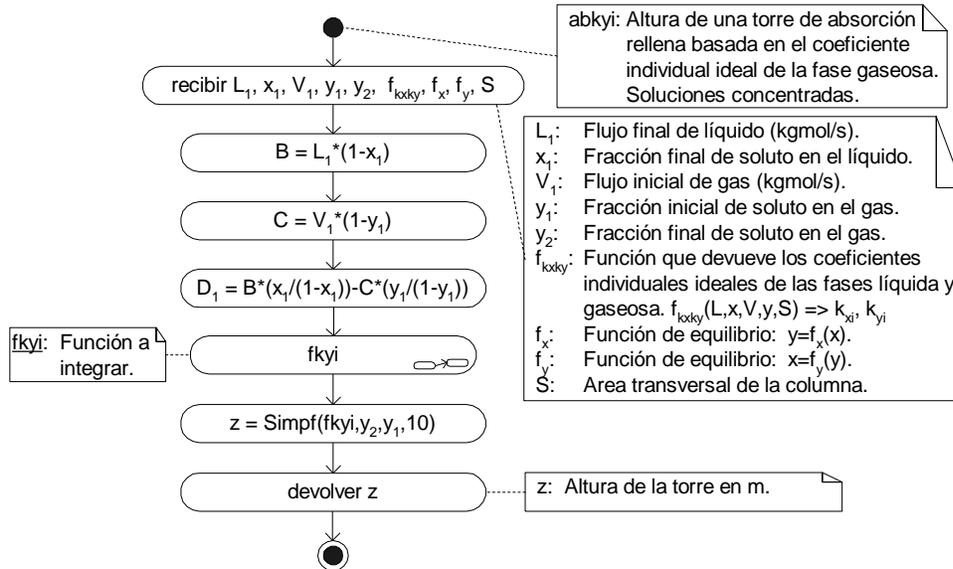
z: 1.58260687651

5.4.2.4. Ejercicios

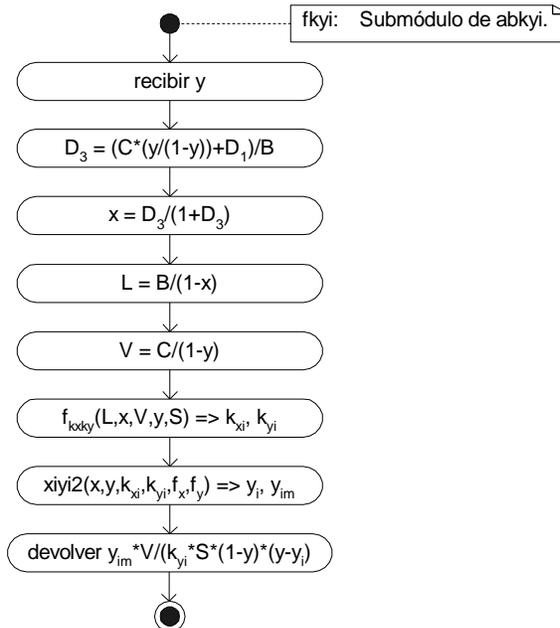
53. Repita el ejemplo 5.4.2.3 considerando que el porcentaje molar inicial de soluto es 25% y el final 1.5%.
54. Repita el ejemplo 5.4.2.3 considerando que el porcentaje de soluto absorbido es del 96%.
55. Repita el ejemplo 5.4.2.3 considerando que el porcentaje absorbido es del 95% y que los datos de equilibrio siguen la ley de Henry: $y = 0.0234x$.

56. Repita el ejemplo 5.4.2.3 considerando que el porcentaje absorbido es del 97% y que los coeficientes individuales de transferencia de masa son constantes e iguales a: $k'_{xa} = 0.7 \text{ kgmol/s.m}^3$; $k'_{ya} = 0.012 \text{ kgmol/s.m}^3$.

El cálculo de la altura basado en el coeficiente individual ideal de la fase gaseosa es muy similar al de la fase líquida. El algoritmo que resuelve la ecuación (35) es el siguiente:



Donde el algoritmo de la función $fkyi$ es:



El código elaborado en base a estos algoritmos es:

```
xNAME abkyi ( Altura de una torre de absorción basada en el coeficiente )
::          ( individual ideal de la fase gaseosa. Soluciones concentradas )
%9 CKNNOLASTWD DROP ( Datos: L1= Flujo final, fase líquida )
CK&DISPATCH1      ( x1= Fracción final, fase líquida )
# 18881            ( V1= Flujo inicial, fase gaseosa )
::                ( y1= Fracción inicial, fase gaseosa )
```

```

NINE ROLL NINE ROLL      ( y2= Fracción final, fase gaseosa)
NINE ROLL NINE ROLL      ( fkxky= Función que devuelve kxi,kyi )
CK&DISPATCH1           ( fx= Función de equilibrio, y=fx[x] )
# 1111                   ( fy= Función de equilibrio, x=fy[y] )
::                         ( S= Area transversal de la columna )
  NINE UNROLL NINE UNROLL ( Locales: B, C, D1, D3, x, y, L, V, )
  NINE UNROLL NINE UNROLL      ( kxi, kyi, yi, yim )
  %0 TWELVE NDUPN DROP
  { LAM L1 LAM x1 LAM V1 LAM y1
    LAM y2 LAM fkxky LAM fx LAM fy
    LAM S LAM B LAM C LAM D1 LAM D3
    LAM x LAM y LAM L LAM V LAM kxi
    LAM kyi LAM yi LAM yim }
  BIND
  LAM L1 %1 LAM x1 %- %* ' LAM B STO
  LAM V1 %1 LAM y1 %- %* ' LAM C STO
  LAM B LAM x1 %1 LAM x1 %- %/ %*
  LAM C LAM y1 %1 LAM y1 %- %/ %* %- ' LAM D1 STO
  '
  :: ( función a integrar: fkyi )
    ' LAM y STO
    LAM C LAM y %1 LAM y %- %/ %* LAM D1 %+ LAM B %/ ' LAM D3 STO
    LAM D3 %1 LAM D3 %+ %/ ' LAM x STO
    LAM B %1 LAM x %- %/ ' LAM L STO
    LAM C %1 LAM y %- %/ ' LAM V STO
    LAM L LAM x LAM V LAM y LAM S LAM fkxky EVAL
    ' LAM kyi STO ' LAM kxi STO
    LAM x LAM y LAM kxi LAM kyi LAM fx LAM fy
    xxiyi2 2DROP ' LAM yim STO DROP ' LAM yi STO DROP
    LAM yim LAM V %* LAM kyi LAM S %* %1 LAM y
    %- %* LAM y LAM yi %- %* %/
  ;
  LAM y2 LAM y1 %10 xSimpf
  ABND
;
ZERO
::
  NINE UNROLL NINE UNROLL
  NINE UNROLL NINE UNROLL
  # 202 DO#EXIT ( 202 = Bad argument type )
;
;
;

```

5.4.2.5. Ejemplo

Volveremos a resolver el ejemplo 5.4.2.1 pero basado en el coeficiente individual de la fase gaseosa.

Solución:

Los datos son los mismos que en el mencionado ejemplo, por lo tanto el programa que resuelve el problema es esencialmente el mismo, excepto que ahora se llama a *abkyi* en lugar de *abkxi*:

```

« 4.21499234694E-2 3.55690964653E-3 0.00081625 0.2 0.02
« 0 0 → L x V y S Gx Gy
« L 1 x - * 18.02 * L x * 64.1 * + S / 'Gx' STO
  V 1 y - * 28.97 * V y * 64.1 * + S / 'Gy' STO
  0.152 Gx 0.82 ^ *

```

```

0.0594 Gy 0.7 ^ * Gx 0.25 ^ *
»
»
« vx vy fun1 »
« vy vx fun1 »
0.0929
abkyi
"z" →TAG
»
    
```

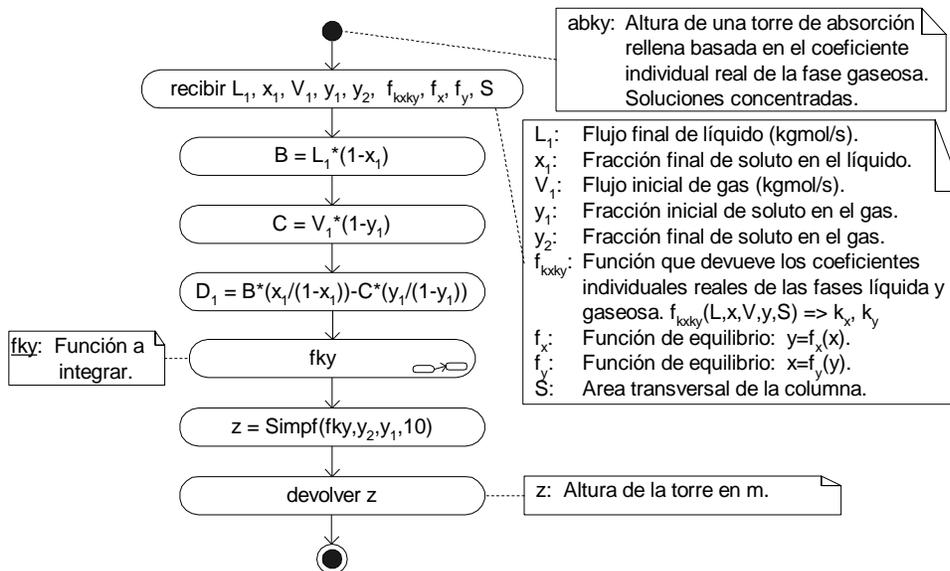
Haciendo correr el programa se obtiene la altura de la torre en aproximadamente 50 segundos:

z: 1.57056448882

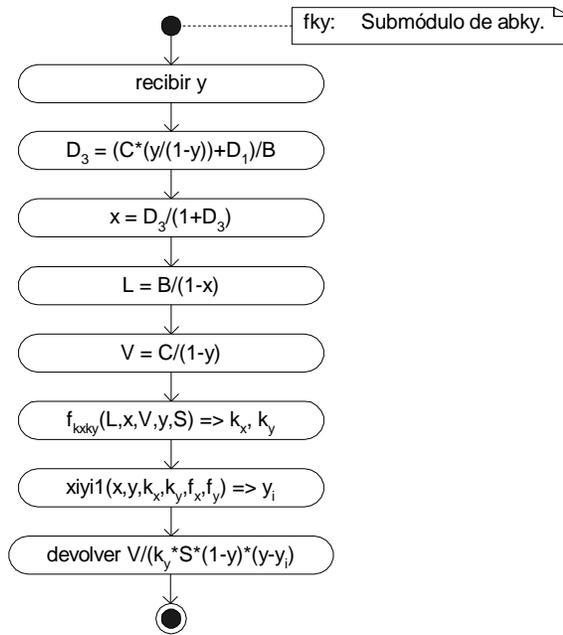
5.4.2.6. Ejercicios

- 57. Repita el ejemplo 5.4.2.1 basado en el coeficiente individual ideal de la fase gaseosa y considerando que el porcentaje molar inicial de soluto es 25% y el final 1.5%.
- 58. Repita el ejemplo 5.4.2.1 basado en el coeficiente individual ideal de la fase gaseosa y considerando que el porcentaje de soluto absorbido es del 96%.
- 59. Repita el ejemplo 5.4.2.1 basado en el coeficiente individual ideal de la fase gaseosa y considerando que el porcentaje absorbido es del 95% y que los datos de equilibrio siguen la ley de Henry: $y = 0.0234x$.
- 60. Repita el ejemplo 5.4.2.1 basado en el coeficiente individual ideal de la fase gaseosa y considerando que el porcentaje absorbido es del 97% y que los coeficientes individuales de transferencia de masa son constantes e iguales a: $k'_x a = 0.85 \text{ kgmol/s.m}^3$; $k'_y a = 0.04 \text{ kgmol/s.m}^3$.

El cálculo de la altura basado en el coeficiente individual real de la fase gaseosa es muy similar al del coeficiente ideal. El algoritmo que resuelve la ecuación (43) es el siguiente:



El algoritmo de la función fky es:



El código elaborado en base a estos algoritmos es:

```

xNAME abky ( Altura de una torre de absorción basada en el coeficiente )
:: ( individual real de la fase gaseosa. Soluciones concentradas )
%9 CKNNOLASTWD DROP ( Datos: L1= Flujo final, fase líquida )
CK&DISPATCH1 ( x1= Fracción final, fase líquida )
# 18881 ( V1= Flujo inicial, fase gaseosa )
:: ( y1= Fracción inicial, fase gaseosa )
NINE ROLL NINE ROLL ( y2= Fracción final, fase gaseosa )
NINE ROLL NINE ROLL ( fkxky= Función que devuelve kx,ky )
CK&DISPATCH1 ( fx= Función de equilibrio, y=fx[x] )
# 1111 ( fy= Función de equilibrio, x=fy[y] )
:: ( S= Area transversal de la columna )
NINE UNROLL NINE UNROLL ( Locales: B, C, D1, D3, x, y, L, V, )
NINE UNROLL NINE UNROLL ( kx, ky, yi )
%0 ELEVEN NDUPN DROP
{ LAM L1 LAM x1 LAM V1 LAM y1
  LAM y2 LAM fkxky LAM fx LAM fy
  LAM S LAM B LAM C LAM D1 LAM D3
  LAM x LAM y LAM L LAM V LAM kx
  LAM ky LAM yi }
BIND
LAM L1 %1 LAM x1 %- %* ' LAM B STO
LAM V1 %1 LAM y1 %- %* ' LAM C STO
LAM B LAM x1 %1 LAM x1 %- %/ %*
LAM C LAM y1 %1 LAM y1 %- %/ %* %- ' LAM D1 STO
'
:: ( función a integrar: fky )
' LAM y STO
LAM C LAM y %1 LAM y %- %/ %* LAM D1 %+ LAM B %/ ' LAM D3 STO
LAM D3 %1 LAM D3 %+ %/ ' LAM x STO
LAM B %1 LAM x %- %/ ' LAM L STO
LAM C %1 LAM y %- %/ ' LAM V STO
LAM L LAM x LAM V LAM y LAM S LAM fkxky EVAL
' LAM ky STO ' LAM kx STO
LAM x LAM y LAM kx LAM ky LAM fx LAM fy
xxiyi1 ' LAM yi STO DROP
  
```

```

    LAM V LAM ky LAM S %* %1 LAM y
    %- %* LAM y LAM yi %- %* %/
;
LAM y2 LAM y1 %10 xSimpf
ABND
;
ZERO
::
NINE UNROLL NINE UNROLL
NINE UNROLL NINE UNROLL
# 202 DO#EXIT ( 202 = Bad argument type )
;
;
;

```

5.4.2.7. Ejemplo

Volveremos a resolver el ejemplo 5.4.2.3 pero basado en el coeficiente real de la fase gaseosa.

Solución:

Los datos son los mismos que en el mencionado ejemplo y la solución prácticamente la misma, excepto que ahora se llama a *abky* en lugar de *abkyi*:

```

« 4.21499234694E-2 3.55690964653E-3 0.00081625 0.2 0.02
« 0 0 → L x V y S Gx Gy
« L 1 x - * 18.02 * L x * 64.1 * + S / 'Gx' STO
  V 1 y - * 28.97 * V y * 64.1 * + S / 'Gy' STO
  0.159 Gx 0.82 ^ *
  0.0599 Gy 0.7 ^ * Gx 0.25 ^ *
»
»
« vx vy fun1 »
« vy vx fun1 »
0.0929
abky
"z" →TAG
»

```

Haciendo correr el programa se obtiene la siguiente altura de la torre en aproximadamente 34 segundos:

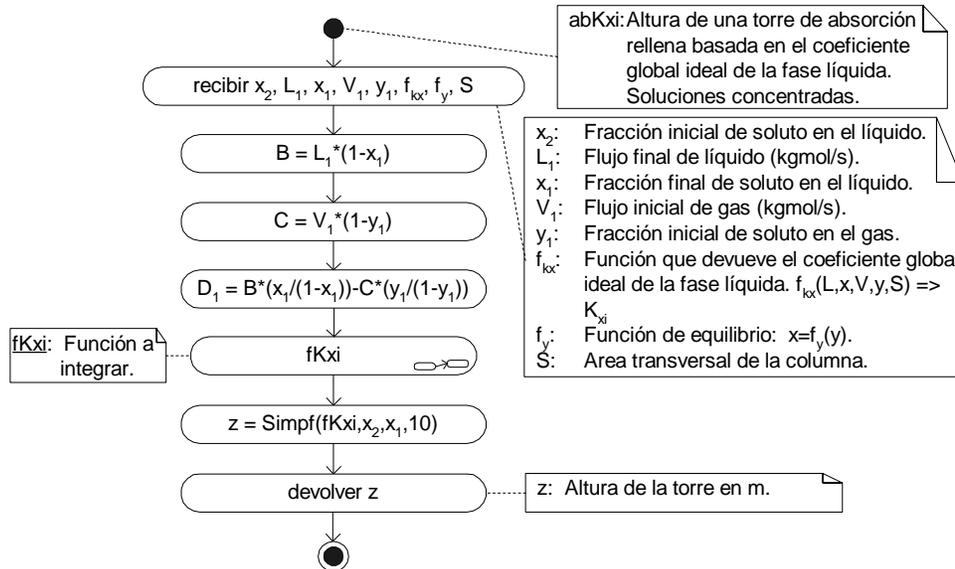
```
z: 1.58105387451
```

5.4.2.8. Ejercicios

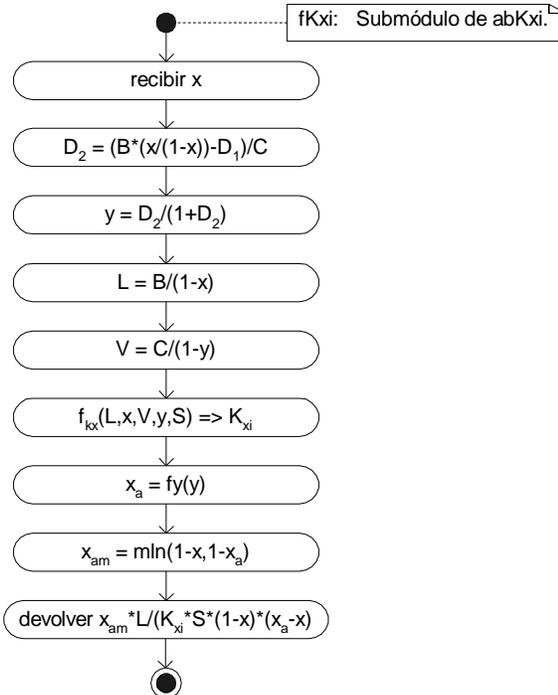
61. Repita el ejemplo 5.4.2.3 basado en el coeficiente real de la fase gaseosa y considerando que el porcentaje molar inicial de soluto es 25% y el final 1.5%.
62. Repita el ejemplo 5.4.2.3 basado en el coeficiente real de la fase gaseosa y considerando que el porcentaje de soluto absorbido es del 96%.
63. Repita el ejemplo 5.4.2.3 basado en el coeficiente real de la fase gaseosa y considerando que el porcentaje absorbido es del 95% y que los datos de equilibrio siguen la ley de *Henry*: $y = 0.0234x$.
64. Repita el ejemplo 5.4.2.3 basado en el coeficiente real de la fase gaseosa y considerando que el porcentaje absorbido es del 97% y que los

coeficientes individuales de transferencia de masa son constantes e iguales a: $k'_{xa} = 0.85 \text{ kgmol/s.m}^3$; $k'_{ya} = 0.04 \text{ kgmol/s.m}^3$.

El algoritmo para el cálculo de la altura basado en el coeficiente global ideal de la fase gaseosa (ecuación 36) es el siguiente:



El algoritmo del submódulo *fKxi* es el siguiente:



El código elaborado en base a estos algoritmos es el siguiente:

```

xNAME abKxi ( Altura de una torre de absorción basada en el coeficiente )
::          ( global ideal de la fase líquida. Soluciones concentradas )
%8 CKNNOLASTWD DROP ( Datos: x2= Fracción inicial, fase líquida )
CK&DISPATCH1      ( L1= Flujo final, fase líquida )
# 11881            ( x1= Fracción final, fase líquida )
::                ( V1= Flujo inicial, fase gaseosa )
    
```

```

8ROLL 8ROLL 8ROLL      ( y1= Fracción inicial, fase gaseosa)
CK&DISPATCH1          ( fKx= Función que devuelve Kxi )
# 1111                  ( fy= Función de equilibrio, x=fy[y] )
::                      ( S= Area transversal de la columna )
  8UNROLL 8UNROLL      ( Locales: B, C, D1, D2, x, y, L, V, )
  8UNROLL                ( Kxi, xa, xam )
%0 ELEVEN NDUPN DROP
{ LAM x2 LAM L1 LAM x1 LAM V1
  LAM y1 LAM fKx LAM fy
  LAM S LAM B LAM C LAM D1 LAM D2
  LAM x LAM y LAM L LAM V LAM Kxi
  LAM xa LAM xam }
BIND
LAM L1 %1 LAM x1 %- %* ' LAM B STO
LAM V1 %1 LAM y1 %- %* ' LAM C STO
LAM B LAM x1 %1 LAM x1 %- %/ %*
LAM C LAM y1 %1 LAM y1 %- %/ %* %- ' LAM D1 STO
'
:: ( función a integrar: fKxi )
  ' LAM x STO
  LAM B LAM x %1 LAM x %- %/ %* LAM D1 %- LAM C %/ ' LAM D2 STO
  LAM D2 %1 LAM D2 %+ %/ ' LAM y STO
  LAM B %1 LAM x %- %/ ' LAM L STO
  LAM C %1 LAM y %- %/ ' LAM V STO
  LAM L LAM x LAM V LAM y LAM S LAM fKx EVAL
  ' LAM Kxi STO
  LAM y LAM fy EVAL ' LAM xa STO
  %1 LAM x %- %1 LAM xa %- mln ' LAM xam STO
  LAM xam LAM L %* LAM Kxi LAM S %* %1 LAM x
  %- %* LAM xa LAM x %- %* %/
;
LAM x2 LAM x1 %10 xSimpf
ABND
;
ZERO
::
  8UNROLL 8UNROLL 8UNROLL
  # 202 DO#EXIT ( 202 = Bad argument type )
;
;
;
;

```

Cuando se trabaja con coeficientes globales, es frecuente asumir que dichos coeficientes son constantes y que en consecuencia no dependen de los flujos y composiciones de las fases. Generalmente el coeficiente global se determina experimentalmente empleando un equipo de laboratorio de altura conocida, operando con flujos superficiales y temperaturas iguales a las del equipo industrial, entonces el coeficiente calculado se emplea para diseñar y/o simular el equipo industrial.

5.4.2.9. Ejemplo

Asumiendo que los datos y resultado del ejemplo 5.4.2.1 corresponden a un equipo experimental, calcule el coeficiente global ideal basado en la fase líquida y emplee luego el coeficiente calculado para predecir la altura que debería tener la torre si el porcentaje de absorción es del 98%.

Solución:

Los datos son los mismos que en el ejemplo mencionado, siendo la altura del equipo 1.57 m (que es el resultado calculado).

Para resolver el problema debemos asumir valores del coeficiente global ideal (K'_x) hasta que la altura calculada sea igual a la altura asumida, es decir hasta que se cumpla la función:

$$f(K'_x) = z_{\text{calculado}} - z_{\text{conocido}} = 0$$

Puesto que se trata de una función de una sola variable K'_x , puede ser resuelta con la ayuda de un método iterativo como el de la *Secante (Seca)* o *Newton - Raphson (Newt)*. El programa elaborado para este fin es:

```

«
« → K
« 0.0 4.21499234694E-2 3.55690964653E-3 0.00081625 0.2
« 5 DROPN K »
« vy vx fun1 »
0.0929
abKxi 1.57 -
»
»
0.8 0.81 Seca
"Kxi" →TAG
»

```

Haciendo correr el programa se obtiene (en aproximadamente 26 segundos):

Kxi: 0.510231419535

Que es el coeficiente global buscado. Con este coeficiente podemos calcular la altura de la torre cuando el porcentaje de absorción (p_a) es del 98%. Previamente calculamos los flujos y composiciones desconocidas con *abrcl*:

```

« 4.2E-2 0.0 0.00081625 0.2 98
abrcl 2 DROPN 2 →LIST
{ L1 x1 } →TAG LIST→ DROP
»

```

Haciendo correr el programa se obtiene:

L1: 0.042159985
x1: 3.79471197629E-3

Ahora podemos calcular la altura de la torre con *abKxi*:

```

« 0.0 0.042159985 3.79471197629E-3 0.00081625 0.2
« 5 DROPN 0.510231419535 »
« vy vx fun1 »
0.0929
abKxi
"z" →TAG
»

```

Haciendo correr el programa se obtiene (en aproximadamente 2.5 segundos):

z: 2.3158630939

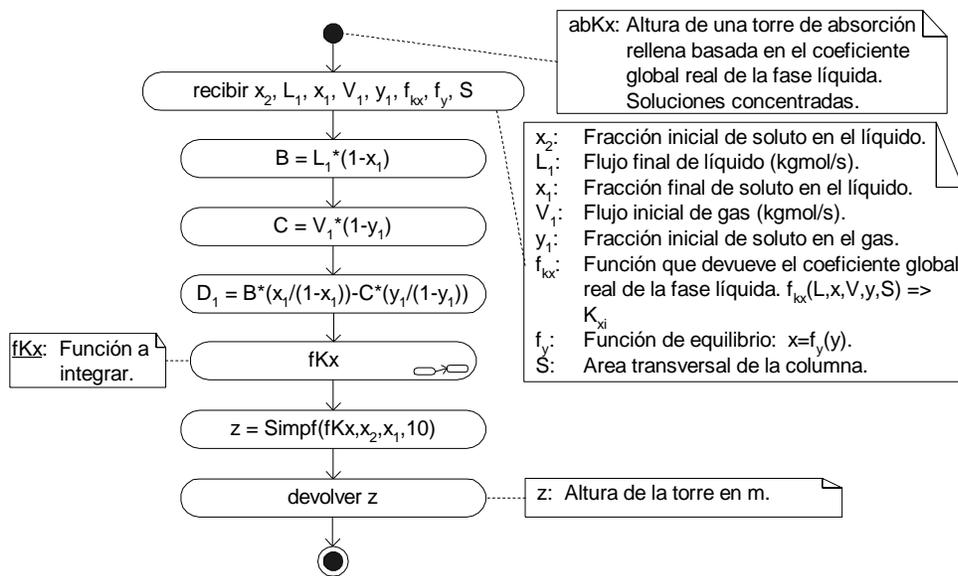
5.4.2.10. Ejercicios

65. Asumiendo que los datos y resultado del ejercicio 49 son de un equipo experimental, calcule el coeficiente global ideal basado en la fase líquida y emplee el coeficiente calculado para predecir la altura que de-

bería tener la torre si el porcentaje molar inicial de soluto es 30% y el final 0.5%.

66. Asumiendo que los datos y resultado del ejercicio 50 son de un equipo experimental, calcule el coeficiente global ideal basado en la fase líquida y emplee el coeficiente calculado para predecir la altura que debería tener la torre si el porcentaje de absorción es del 99%.
67. Asumiendo que los datos y resultado del ejercicio 51 son de un equipo experimental, calcule el coeficiente global ideal basado en la fase líquida y emplee el coeficiente calculado para predecir la altura que debería tener la torre si el porcentaje de absorción es del 99.5%.
68. Asumiendo que los datos y resultado del ejercicio 49 son de un equipo experimental, calcule el coeficiente global ideal basado en la fase líquida y emplee el coeficiente calculado para predecir la altura que debería tener la torre si el porcentaje de absorción es del 99.6%.

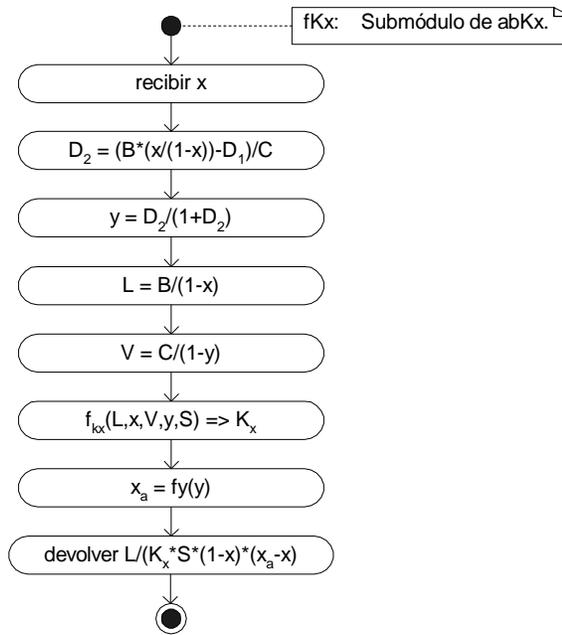
El cálculo de la altura basado en el coeficiente global real de la fase líquida es muy similar al del coeficiente ideal. El algoritmo que resuelve la ecuación (44) es el siguiente:



El algoritmo del submódulo fKx se presenta en la siguiente página. El código elaborado en base a estos algoritmos es el siguiente:

```

xNAME abKx ( Altura de una torre de absorción basada en el coeficiente )
::          ( global real de la fase líquida. Soluciones concentradas )
%8 CKNNOLASTWD DROP ( Datos: x2= Fracción inicial, fase líquida )
CK&DISPATCH1      ( L1= Flujo final, fase líquida )
# 11881            ( x1= Fracción final, fase líquida )
::                ( V1= Flujo inicial, fase gaseosa )
8ROLL 8ROLL 8ROLL ( y1= Fracción inicial, fase gaseosa )
CK&DISPATCH1      ( fKx= Función que devuelve Kx )
# 1111            ( fy= Función de equilibrio, x=fy[y] )
::                ( S= Área transversal de la columna )
8UNROLL 8UNROLL   ( Locales: B, C, D1, D2, x, y, L, V, )
8UNROLL           ( Kx, xa )
%0 TEN NDUPN DROP
{ LAM x2 LAM L1 LAM x1 LAM V1
    
```



```

LAM y1 LAM fkx LAM fy
LAM S LAM B LAM C LAM D1 LAM D2
LAM x LAM y LAM L LAM V LAM Kx
LAM xa }
BIND
LAM L1 %1 LAM x1 %- %* ' LAM B STO
LAM V1 %1 LAM y1 %- %* ' LAM C STO
LAM B LAM x1 %1 LAM x1 %- %/ %*
LAM C LAM y1 %1 LAM y1 %- %/ %* %- ' LAM D1 STO
'
:: ( función a integrar: fKx )
' LAM x STO
LAM B LAM x %1 LAM x %- %/ %* LAM D1 %- LAM C %/ ' LAM D2 STO
LAM D2 %1 LAM D2 %+ %/ ' LAM y STO
LAM B %1 LAM x %- %/ ' LAM L STO
LAM C %1 LAM y %- %/ ' LAM V STO
LAM L LAM x LAM V LAM y LAM S LAM fkx EVAL
' LAM Kx STO
LAM y LAM fy EVAL ' LAM xa STO
LAM L LAM Kx LAM S %* %1 LAM x
%- %* LAM xa LAM x %- %* %/
;
LAM x2 LAM x1 %10 xSimpf
ABND
;
ZERO
::
8UNROLL 8UNROLL 8UNROLL
# 202 DO#EXIT ( 202 = Bad argument type )
;
;
;

```

5.4.2.11. Ejemplo

Volveremos a resolver el ejemplo 5.4.2.9, pero en base al coeficiente global real de la fase líquida.

Solución:

Los datos y el programa que resuelve el problema son los mismos que en el mencionado ejemplo, sólo que ahora en lugar de llamar a *abKxi* se llama a *abKx*:

```
«
« → K
« 0.0 4.21499234694E-2 3.55690964653E-3 0.00081625 0.2
« 5 DROPN K »
« vy vx fun1 »
0.0929
abKx 1.57 -
»
»
0.8 0.81 Seca
"Kxi" →TAG
»
```

Haciendo correr el programa se obtiene (en aproximadamente 22 segundos):

Kxi: 0.511484268186

Ahora podemos calcular la altura de la torre con *abKx*:

```
« 0.0 0.042159985 3.79471197629E-3 0.00081625 0.2
« 5 DROPN 0.511484268186 »
« vy vx fun1 »
0.0929
abKx
"z" →TAG
»
```

Haciendo correr el programa se obtiene (en aproximadamente 2.1 segundos):

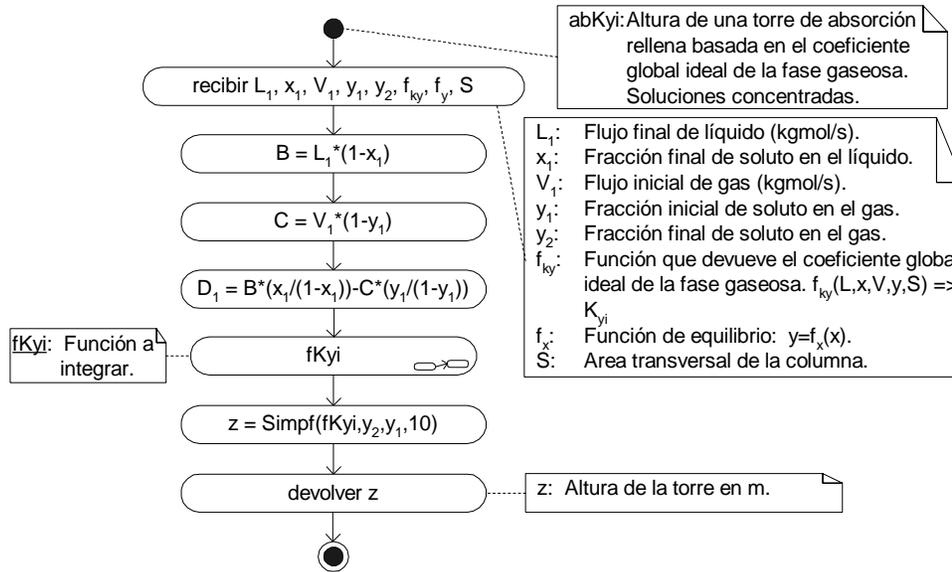
z: 2.31487687059

5.4.2.12. Ejercicios

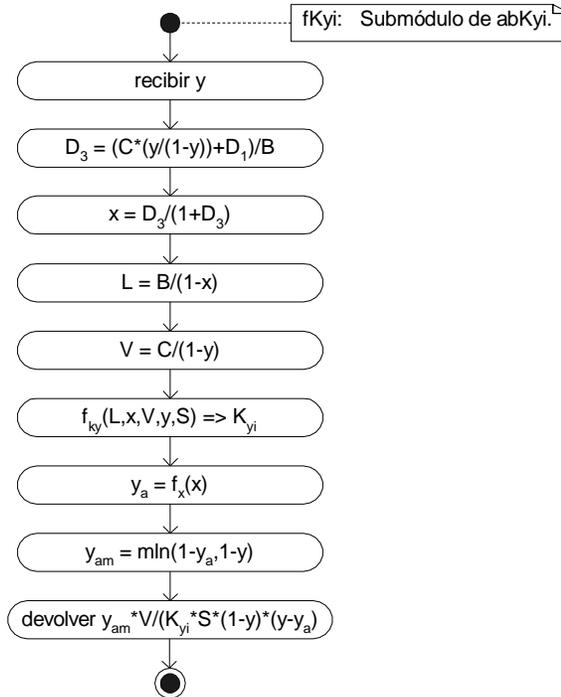
69. Repita el ejercicio 65 empleando el coeficiente global real basado en la fase líquida.
70. Repita el ejercicio 66 empleando el coeficiente global real basado en la fase líquida.
71. Repita el ejercicio 67 empleando el coeficiente global real basado en la fase líquida.
72. Repita el ejercicio 68 empleando el coeficiente global real basado en la fase líquida.

El cálculo de la altura basado en el coeficiente global ideal de la fase gaseosa es similar al de la fase líquida. El algoritmo que resuelve la ecuación (37) se presenta en la siguiente página y el código elaborado en base al mismo es:

```
xNAME abKyi ( Altura de una torre de absorción basada en el coeficiente )
::          ( global ideal de la fase gaseosa. Soluciones concentradas )
%8 CKNNOLASTWD DROP ( Datos: L1= Flujo final, fase líquida )
CK&DISPATCH1      ( x1= Fracción final, fase líquida )
# 11881            ( V1= Flujo inicial, fase gaseosa )
::                ( y1= Fracción inicial, fase gaseosa )
```



El algoritmo del submódulo fky es:



El código elaborado en base a estos algoritmos es el siguiente:

```

8ROLL 8ROLL 8ROLL      ( y2= Fracción final, fase gaseosa)
CK&DISPATCH1          ( fky= Función que devuelve Kyi )
# 1111                  ( fx= Función de equilibrio, y=fx[x] )
::                       ( S= Área transversal de la columna )
8UNROLL 8UNROLL        ( Locales: B, C, D1, D2, x, y, L, V, )
8UNROLL                 ( Kyi, ya, yam )
%0 ELEVEN NDUPN DROP
{ LAM L1 LAM x1 LAM V1
  LAM y1 LAM y2 LAM fky LAM fy
  LAM S LAM B LAM C LAM D1 LAM D3
  LAM x LAM y LAM L LAM V LAM Kyi

```

```

    LAM ya LAM yam }
  BIND
  LAM L1 %1 LAM x1 %- %* ' LAM B STO
  LAM V1 %1 LAM y1 %- %* ' LAM C STO
  LAM B LAM x1 %1 LAM x1 %- %/ %*
  LAM C LAM y1 %1 LAM y1 %- %/ %* %- ' LAM D1 STO
  '
  :: ( función a integrar: fKyI )
  ' LAM y STO
  LAM C LAM y %1 LAM y %- %/ %* LAM D1 %+ LAM B %/ ' LAM D3 STO
  LAM D3 %1 LAM D3 %+ %/ ' LAM x STO
  LAM B %1 LAM x %- %/ ' LAM L STO
  LAM C %1 LAM y %- %/ ' LAM V STO
  LAM L LAM x LAM V LAM y LAM S LAM fky EVAL
  ' LAM KyI STO
  LAM x LAM fy EVAL ' LAM ya STO
  %1 LAM ya %- %1 LAM y %- mln ' LAM yam STO
  LAM yam LAM V %* LAM KyI LAM S %* %1 LAM y
  %- %* LAM y LAM ya %- %* %/
  ;
  LAM y2 LAM y1 %10 xSimpf
  ABND
  ;
  ZERO
  ::
  8UNROLL 8UNROLL 8UNROLL
  # 202 DO#EXIT ( 202 = Bad argument type )
  ;
  ;
  ;
  ;

```

5.4.2.13. Ejemplo

Volveremos a resolver el ejemplo 5.4.2.9, pero en base al coeficiente global ideal de la fase gaseosa.

Solución:

Los datos y el programa que resuelve el problema son los mismos que en el mencionado ejemplo, sólo que ahora en lugar de llamar a *abKxi* se llama a *abKyI*:

```

«
« → K
« 4.21499234694E-2 3.55690964653E-3 0.00081625 0.2 0.02
« 5 DROPN K »
« vx vy fun1 »
0.0929
abKyI 1.57 -
»
»
0.005 0.0051 Seca
"KyI" →TAG
»

```

Haciendo correr el programa se obtiene (en aproximadamente 23 segundos):

KyI: 1.61823592955E-2

Ahora podemos calcular la altura de la torre con *abKx*:

```

« 0.042159985 3.79471197629E-3 0.00081625 0.2 0.02

```

```

« 5 DROPN 1.61823592955E-2 »
« vx vy fun1 »
0.0929
abKyI
"z" →TAG
»

```

Haciendo correr el programa se obtiene (en aproximadamente 2.3 segundos):

```
z: 1.79406432808
```

5.4.2.14. Ejercicios

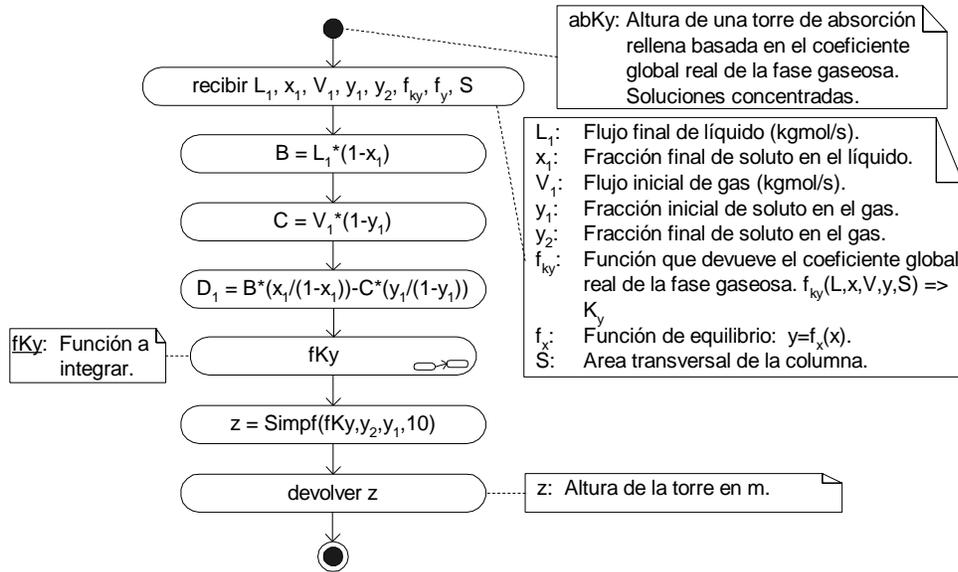
73. Repita el ejercicio 65 empleando el coeficiente global ideal basado en la fase gaseosa.
74. Repita el ejercicio 66 empleando el coeficiente global ideal basado en la fase gaseosa.
75. Repita el ejercicio 67 empleando el coeficiente global ideal basado en la fase gaseosa.
76. Repita el ejercicio 68 empleando el coeficiente global ideal basado en la fase gaseosa.

El cálculo de la altura basado en el coeficiente global real es similar al del coeficiente ideal. El algoritmo que resuelve la ecuación (45) se presenta en la siguiente página y el código elaborado en base al mismo es:

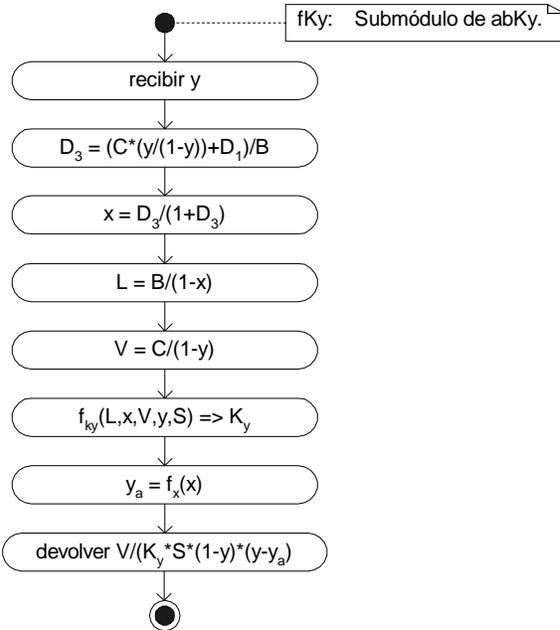
```

xNAME abKy ( Altura de una torre de absorción basada en el coeficiente )
:: ( global real de la fase gaseosa. Soluciones concentradas )
%8 CKNNOLASTWD DROP ( Datos: L1= Flujo final, fase líquida )
CK&DISPATCH1 ( x1= Fracción final, fase líquida )
# 11881 ( V1= Flujo inicial, fase gaseosa )
:: ( y1= Fracción inicial, fase gaseosa )
8ROLL 8ROLL 8ROLL ( y2= Fracción final, fase gaseosa )
CK&DISPATCH1 ( fky= Función que devuelve Ky )
# 1111 ( fx= Función de equilibrio, y=fx[x] )
:: ( S= Area transversal de la columna )
8UNROLL 8UNROLL ( Locales: B, C, D1, D2, x, y, L, V, )
8UNROLL ( Ky, ya )
%0 TEN NDUPN DROP
{ LAM L1 LAM x1 LAM V1
LAM y1 LAM y2 LAM fky LAM fx
LAM S LAM B LAM C LAM D1 LAM D3
LAM x LAM y LAM L LAM V LAM Ky
LAM ya }
BIND
LAM L1 %1 LAM x1 %- %* ' LAM B STO
LAM V1 %1 LAM y1 %- %* ' LAM C STO
LAM B LAM x1 %1 LAM x1 %- %/ %*
LAM C LAM y1 %1 LAM y1 %- %/ %* %- ' LAM D1 STO
'
:: ( función a integrar: fKy )
' LAM y STO
LAM C LAM y %1 LAM y %- %/ %* LAM D1 %+ LAM B %/ ' LAM D3 STO
LAM D3 %1 LAM D3 %+ %/ ' LAM x STO
LAM B %1 LAM x %- %/ ' LAM L STO
LAM C %1 LAM y %- %/ ' LAM V STO

```



El algoritmo del submódulo *fKy* es:



```

LAM L LAM x LAM V LAM y LAM S LAM fky EVAL
' LAM Ky STO
LAM x LAM fx EVAL ' LAM ya STO
LAM V LAM Ky LAM S %* %1 LAM y
%- %* LAM y LAM ya %- %* %/
;
LAM y2 LAM y1 %10 xSimpf
ABND
;
ZERO
::
8UNROLL 8UNROLL 8UNROLL
# 202 DO#EXIT ( 202 = Bad argument type )
;
;

```

;

5.4.2.15. Ejemplo

Volveremos a resolver el ejemplo 5.4.2.9, pero en base al coeficiente global real de la fase gaseosa.

Solución:

El programa que calcula el coeficiente global volumétrico es:

```
«
« → K
« 4.21499234694E-2 3.55690964653E-3 0.00081625 0.2 0.02
« 5 DROPN K »
« vx vy fun1 »
0.0929
abKy 1.57 -
»
»
0.005 0.0051 Seca
"Ky1" →TAG
»
```

Haciendo correr el programa se obtiene (en aproximadamente 25 segundos):

Ky: 1.73139143294E-2

Ahora podemos calcular la altura de la torre con *abKx*:

```
« 0.042159985 3.79471197629E-3 0.00081625 0.2 0.02
« 5 DROPN 1.73139143294E-2 »
« vx vy fun1 »
0.0929
abKy
"z" →TAG
»
```

Haciendo correr el programa se obtiene (en aproximadamente 2 segundos):

z: 1.79701048911

5.4.2.16. Ejercicios

77. Repita el ejercicio 65 empleando el coeficiente global real basado en la fase gaseosa.
78. Repita el ejercicio 66 empleando el coeficiente global real basado en la fase gaseosa.
79. Repita el ejercicio 67 empleando el coeficiente global real basado en la fase gaseosa.
80. Repita el ejercicio 68 empleando el coeficiente global real basado en la fase gaseosa.

6. EXTRACCIÓN LÍQUIDO - LÍQUIDO

En esta operación se ponen en contacto dos soluciones inmiscibles, una de las cuales tiene disuelto un compuesto que es soluble en ambas soluciones. Como resultado del contacto el compuesto soluble se distribuye en ambas soluciones, quedando en mayor proporción en aquella con la cual tiene mayor afinidad química y en la cual por consecuencia es más soluble. De esta manera se separa un compuesto de una solución empleando otra solución en la cual el soluto es más soluble.

Las dos fases resultantes de la extracción líquido - líquido reciben el nombre de *fase refinado* y *fase extracto*. La *fase refinado* es la solución que pierde soluto y la *fase extracto* es la solución que gana soluto.

La extracción líquido - líquido se emplea normalmente como alternativa a la destilación en aquellos casos donde la destilación es muy costosa desde el punto de vista energético.

Así por ejemplo en la fabricación del ácido acético, las soluciones acuosas suelen tener concentraciones de ácido acético de alrededor de un 5%, por lo tanto, para separar el ácido acético del agua por destilación se debe evaporar el 95% de la solución, lo que implica un gran gasto de energía. Una alternativa más económica consiste en extraer el ácido acético empleando un solvente orgánico (con el cual el ácido acético es más afín). Entonces las soluciones concentradas de ácido acético - solvente orgánico son sometidas a una destilación, pero ahora el porcentaje de líquido a evaporar (y por lo tanto el costo) es considerablemente inferior al 95% de la solución original.

Otro ejemplo lo constituyen los ácidos grasos de elevado peso molecular que pueden separarse de los aceites vegetales por destilación al vacío, proceso es bastante costoso. Una alternativa más económica consiste en llevar a cabo la extracción líquido - líquido empleando propano como líquido extractor. De esa manera se obtienen soluciones concentradas de ácidos grasos y propano que pueden ser purificadas por destilación, pero con un menor costo energético, pues la cantidad de líquido a evaporar es mucho menor a la de la solución original.

La extracción líquido - líquido se emplea también en la industria metalúrgica y farmacéutica, donde se aprovecha la afinidad de ciertos solventes con relación a compuestos específicos para lograr separaciones selectivas, ese es el caso por ejemplo de la penicilina que se separa de la solución fermentativa por extracción líquido - líquido.

6.1. Relaciones de equilibrio para la extracción líquido - líquido

Como todas las operaciones que estudiamos en esta materia, la extracción líquido - líquido será abordada desde un punto de vista simplificado pero práctico. Sin importar cuan complejas sean las soluciones a tratar, se asumirá en todos los casos que solo existen tres componentes: *A*: el soluto (que es soluble en ambas fases); *B*: El solvente de la fase refinado y *C*: El solvente de la fase extracto.

Estos tres componentes pueden corresponder a tres compuestos reales, como en el sistema agua - acetona - ácido acético o pueden ser mezclas multicomponentes como en el sistema agua - éter de petróleo - aceite esencial (donde el éter de petróleo y el aceite esencial son mezclas multicomponentes).

Para que este punto de vista simplificado sea de utilidad práctica es necesario que los datos sean obtenidos experimentalmente. Para ello se emplean equipos discontinuos provistos de un mecanismo de agitación, donde se ponen en contacto cantidades conocidas de los tres componentes hasta que alcancen el equilibrio. Entonces se separan las dos fases y se analiza su composición.

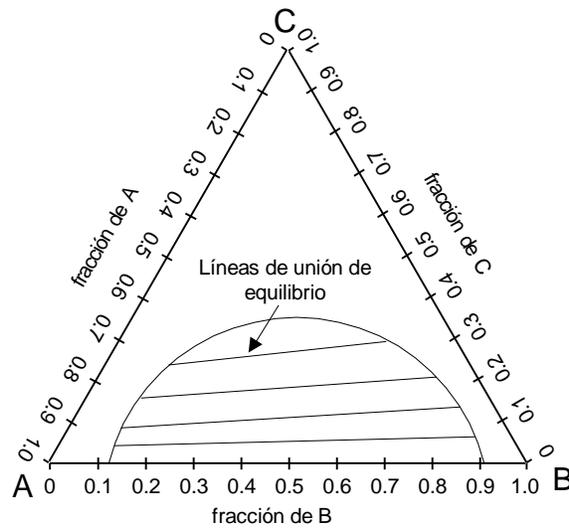
Los datos de cada prueba constituyen una fila en la tabla de equilibrio o una línea en la gráfica de equilibrio, siendo necesario realizar varias pruebas para tener una tabla o gráfica representativa.

Las tablas de equilibrio tienen usualmente 6 columnas, 3 para cada fase, tal como se muestra en el siguiente ejemplo:

Fase refinado			Fase extracto		
x_A	x_B	x_C	y_A	y_B	y_C
0.00	0.98	0.02	0.00	0.05	0.95
0.15	0.80	0.05	0.20	0.06	0.74
...

Aunque en la práctica sólo se requieren 4 columnas, pues la fracción del tercer componente puede ser calculada tomando en cuenta que la suma de las fracciones debe ser igual a 1.

En su forma gráfica los datos de equilibrio se presentan en diagramas triangulares como el que se muestra en la figura:



Donde las líneas de unión de equilibrio unen los puntos que corresponden a las fases refinado y extracto, de una misma fila, en la tabla de equilibrio.

Debido a lo tedioso, poco exacto y anticuado que resulta el trabajo con gráficas, no emplearemos los métodos gráficos en el dimensionamiento y diseño de los equipos de extracción líquido - líquido. Sin embargo es útil conocer la forma y disposición de estas gráficas pues en ocasiones los datos de equilibrio sólo están disponibles en esta forma y en esos casos deben ser

leídos de las mismas y colocados en la forma tabular, que es una forma más adecuada para el diseño asistido por computadora.

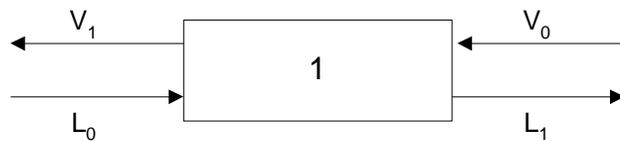
Como los datos de una gráfica pueden ser llevados a la forma tabular (y viceversa), en lo sucesivo asumiremos que los datos de equilibrio están disponibles en la forma tabular.

6.2. Extracción en una etapa de equilibrio

Abordaremos el diseño de los equipos de extracción desde el punto de vista de las etapas de equilibrio (o etapas teóricas).

En una etapa de equilibrio o etapa teórica se asume que las fases que entran en contacto tienen el tiempo suficiente como para alcanzar el equilibrio, de manera que las corrientes resultantes siempre están en equilibrio.

En la siguiente figura se muestra el esquema de una etapa de equilibrio:



Donde las corrientes V_1 y L_1 están en equilibrio. Tomando como base el esquema podemos realizar un balance general de materia:

$$L_0 + V_0 = L_1 + V_1 = M \quad (1)$$

Donde L_0 y V_0 son los flujos iniciales de las fases refinado y extracto, y L_1 y V_1 son los flujos finales.

En la mayoría de los casos se conocen los flujos iniciales (L_0 , V_0), de manera que la constante M puede ser calculada.

Realizamos ahora un balance de materia para los componentes A (solute) y B (solvente de la fase refinado):

$$L_0 x_{A0} + V_0 y_{A0} = L_1 x_{A1} + V_1 y_{A1} \quad (2)$$

$$L_0 x_{B0} + V_0 y_{B0} = L_1 x_{B1} + V_1 y_{B1} \quad (3)$$

Donde x_A (x_{A0} , x_{B0} , etc.) son las fracciones en la fase refinado en las corrientes respectivas y y_A (y_{A0} , y_{A1} , etc.) son las fracciones en la fase extracto.

No se requiere un balance de materia para el componente C (solvente de la fase extracto) pues sus fracciones pueden ser calculadas con:

$$x_C = 1 - x_A - x_B \quad (4)$$

$$y_C = 1 - y_A - y_B \quad (5)$$

Normalmente se conocen las fracciones molares iniciales de las corrientes entrantes (x_{A0} , x_{B0} , y_{A0} , y_{B0}), de manera que la parte izquierda de las ecuaciones (2) y (3) pueden ser calculadas.

Se tiene entonces un sistema de 3 ecuaciones (ecuaciones 1, 2 y 3) con 6 incógnitas (L_1 , V_1 , x_{A1} , x_{B1} , y_{A1} , y_{B1}), en consecuencia se requieren tres ecuaciones más para resolver el sistema.

Puesto que el sistema de extracción líquido - líquido tiene en el equilibrio 3 grados de libertad (3 (componentes) - 2 (fases) + 2 = 3), dos de los cuales son fijados al establecer la presión y la temperatura del sistema,

sólo nos queda un grado de libertad. Por lo tanto si fijamos una de las concentraciones, de las cuatro existentes: x_A , x_B , y_A , y_B , las otras tres quedan automáticamente fijadas. Este hecho se verifica en la practica mediante los datos de equilibrio, donde conociendo una de las concentraciones: x_A , x_B , y_A o y_B , se puede leer (o interpolar) el valor de las otras tres. Entonces las tres ecuaciones faltantes las constituyen los datos de equilibrio.

Así por ejemplo si se fija el valor de x_A , la fracción x_B puede ser calculada como una función de x_A :

$$x_B = f_{x_A x_B}(x_A) \tag{6}$$

Donde la función $f_{x_A x_B}$ devuelve el valor de x_B interpolando con los datos de las columnas x_A y x_B de la tabla de equilibrio.

La fracción y_A puede ser calculada también como una función de x_A :

$$y_A = f_{x_A y_A}(x_A) \tag{7}$$

Donde la función $f_{x_A y_A}$ devuelve el valor de y_A interpolando con los datos de las columnas x_A y y_A de la tabla de equilibrio.

De igual manera la fracción y_B puede ser calculada como una función de x_A :

$$y_B = f_{x_A y_B}(x_A) \tag{8}$$

Donde la función $f_{x_A y_B}$ devuelve el valor de y_B interpolando con los datos de las columnas x_A y y_B de la tabla de equilibrio.

Con las ecuaciones (6), (7) y (8) se tiene ahora un sistema de 6 ecuaciones con 6 incógnitas, por lo que el sistema puede ser resuelto. Estas funciones de pueden ser implementadas empleando los datos de la tabla de equilibrio y un método de interpolación.

En lo sucesivo asumiremos que los datos de equilibrio correspondientes a las columnas x_A , x_B , y_A y y_B de la tabla de equilibrio están almacenados como vectores en las variables globales x_A , x_B , y_A y y_B respectivamente.

Para realizar la interpolación se puede emplear uno de los métodos polinomiales (como el de Newton o Lagrange), un método de interpolación segmentario (como Spline Cúbico) o el método de interpolación lineal. En este capítulo emplearemos el método de interpolación lineal (*INTLP*), porque a pesar de ser el de menor precisión permite extrapolar datos, algo que es útil cuando los datos de equilibrio son limitados. Además este método, al ser lineal, nunca presenta un comportamiento oscilatorio entre los puntos, algo que si puede ocurrir con los métodos polinomiales y segmentarios.

Tomando en cuenta las anteriores consideraciones y empleando el método de interpolación lineal la función $f_{x_A x_B}$ sería:

« xA xB ROT INTLP »

La función $f_{x_A y_A}$ como:

« xA yA ROT INTLP »

Y la función $f_{x_A y_B}$ como:

« xA yB ROT INTLP »

Para utilizar estas funciones es necesario que la columna correspondiente a la variable independiente (en este caso x_A) debe estar ordenada ascendentemente.

Como estas funciones no son lineales, es más ni siquiera son funciones analíticas, la resolución del sistema de 6 ecuaciones es en la mayoría de

los casos iterativa (excepto cuando se conoce una de las composiciones de salida).

Generalmente la forma más sencilla de resolver el sistema de ecuaciones consiste en ordenarlas como una función de una sola variable y emplear un método iterativo, como del de Newton Raphson, para encontrar la solución.

Así en el caso más frecuente, donde se conocen los flujos y composiciones iniciales (L_0 , x_{A0} , x_{B0} , V_0 , y_{A0} , y_{B0}), el sistema de ecuaciones puede ser ordenado como una función de x_{A1} :

$$\begin{aligned}
 f(x_{A1}) &= L_1 + V_1 - M = 0 \\
 x_{B1} &= f_{x_{A1}B}(x_{A1}) \\
 y_{A1} &= f_{x_{A1}A}(x_{A1}) \\
 y_{B1} &= f_{x_{A1}B}(x_{A1}) \\
 \left. \begin{aligned}
 L_1 x_{A1} + V_1 y_{A1} &= C_{A0} \\
 L_1 x_{B1} + V_1 y_{B1} &= C_{B0}
 \end{aligned} \right\} \quad L_1 = \frac{C_{A0} y_{B1} - C_{B0} y_{A1}}{x_{A1} y_{B1} - x_{B1} y_{A1}}; \quad V_1 = \frac{C_{A0} - L_1 x_{A1}}{y_{A1}}
 \end{aligned}$$

Donde:

$$\begin{aligned}
 M &= L_0 + V_0 \\
 C_{A0} &= L_0 x_{A0} + V_0 y_{A0} \\
 C_{B0} &= L_0 x_{B0} + V_0 y_{B0}
 \end{aligned}$$

Un programa que automatiza la solución de esta función es el siguiente:

```

« 1 9 START 0 NEXT
  □ L0 XA0 XB0 V0 YA0 YB0 FXAXB FXAYA FXAYB
  M CA0 CB0 XA1 XB1 YA1 YB1 L1 V1
« L0 V0 + 'M' STO
  L0 XA0 * V0 YA0 * + 'CA0' STO
  L0 XB0 * V0 YB0 * + 'CB0' STO
« 'XA1' STO
  XA1 FXAXB EVAL 'XB1' STO
  XA1 FXAYA EVAL 'YA1' STO
  XA1 FXAYB EVAL 'YB1' STO
  CA0 YB1 * CB0 YA1 * - XA1 YB1 * XB1 YA1 * - / 'L1' STO
  CA0 L1 XA1 * - YA1 / 'V1' STO
  L1 V1 + M -
» CA0 M / 2 / 9 30 NEWT DROP
  L1 "L1" □TAG XA1 "XA1" □TAG XB1 "XB1" □TAG
  V1 "V1" □TAG YA1 "YA1" □TAG YB1 "YB1" □TAG
»
»

```

El valor inicial que se emplea en este programa para el método de Newton ($C_{A0}/2M$) es la mitad del valor medio del soluto en la mezcla.

En lo sucesivo se asumirá que este programa ha sido guardado en la variable global "ELLI". El programa recibe como datos y en ese orden: el flujo inicial de refinado (L_0); las fracciones iniciales de soluto y solvente A en la fase refinado (x_{A0} , x_{B0}); el flujo inicial de extracto (V_0); las composiciones iniciales de soluto y solvente A en la fase extracto (y_{A0} , y_{B0}); la función de equilibrio que devuelve el valor de x_B para un valor dado de x_A (FXAXB); la función de equilibrio que devuelve el valor de y_A para un valor dado de x_A (FXAYA) y la función de equilibrio que devuelve el valor de y_B para un valor dado de x_A (FXAYB).

6.2.1. Ejemplo

Se emplea 30 kg de eter isopropílico puro para extraer ácido acético de una mezcla que contiene 10 kg de ácido acético y 60 kg de agua, empleando un equipo discontinuo donde se alcanza el equilibrio. Los datos de equilibrio para este sistema son los siguientes:

Fase agua (% en peso)		Fase éter isopropílico (% en peso)	
Ácido acético	Agua	Ácido acético	Agua
0.00	98.8	0.00	0.6
0.69	98.1	0.18	0.5
1.41	97.1	0.37	0.7
2.89	95.5	0.79	0.8
6.42	91.7	1.93	1.0
13.30	84.4	4.82	1.9
25.50	71.1	11.40	3.9
36.70	58.9	21.60	6.9
44.30	45.1	31.10	10.8
46.40	37.1	36.20	15.1

Calcule

- a) Las masas y composiciones finales de las fases refinado y extracto así como el % de ácido acético extraído.
- b) Repita el inciso anterior pero empleando 100 kg de éter isopropílico.

Solución

a)

Este ejemplo corresponde al caso más frecuente, pues se conocen todos los flujos y composiciones iniciales y aunque en lugar de flujos másicos tenemos masas, el procedimiento de cálculo no varía pues en las ecuaciones del balance de materia (ecuaciones 1 a 3) da lo mismo si se emplean flujos másicos o masas simplemente.

Los datos con los que contamos son:

$L_0 = 10+60 = 70 \text{ kg}$
 $x_{c0} = 10/70 = 0.142857142857$
 $x_{A0} = 60/70 = 0.857142857143$
 $V_0 = 30 \text{ kg}$
 $y_{c0} = 0/30 = 0$
 $y_{A0} = 0/30 = 0$

Contamos además con los datos de equilibrio y como la columna correspondiente a la variable independiente ($x_c = \%$ de ácido acético) ya está en orden ascendente, podemos guardar los datos en las variables globales x_c , x_A , y_c y y_A , recordando dividirlos previamente entre 100 (pues los datos están forma de porcentajes):

```
[ 0. .69 1.41 2.89 6.42 13.3 25.5 36.7 44.3 46.4 ] 100 / 'xA' STO
[ 98.8 98.1 97.1 95.5 91.7 84.4 71.1 58.9 45.1 37.1 ] 100 / 'xB' STO
[ 0. .18 .37 .79 1.93 4.82 11.4 21.6 31.1 36.2 ] 100 / 'yA' STO
[ .6 .5 .7 .8 1. 1.9 3.9 6.9 10.8 15.1 ] 100 / 'yB' STO
```

Entonces podemos emplear el programa ELL1 para resolver el problema escribiendo lo siguiente en la pila:

```
70
10 70 /
60 70 /
```

```

30
0
0
« xA xB ROT INTLP »
« xA yA ROT INTLP »
« xA yB ROT INTLP»
ELL1

```

Con lo que obtenemos los resultados:

```

L1: 69.6747172756
XA1: .124161404768
XB1: .853378160639
V1: 30.3252827229
YA1: 4.44872761308E-2
YB1: 1.78437884144E-2

```

Que son las masas y composiciones finales de las fases refinado (*L*) y extracto (*V*). Por lo tanto la masa de ácido acético extraída es:

$$\text{Acido acético extraído} = V_1 Y_{A1} = (30.3252827229)(4.44872761308E-2) = 1.34908922624$$

Por lo tanto el porcentaje extraído es:

$$\% \text{ de Acido acético extraído} = \frac{1.34908922624}{10} * 100 = 13.49\%$$

b)

Para este inciso solo cambia la masa de éter isopropílico de 30 a 100 kg, entonces resolvemos el problema simplemente cambiando este valor:

```

70
10 70 /
60 70 /
100
0
0
« xA xB ROT INTLP »
« xA yA ROT INTLP »
« xA yB ROT INTLP»
ELL1

```

Con lo que obtenemos los resultados:

```

XA1: 9.81973146818E-2
XB1: .880927267852
V1: 103.588820587
YA1: 3.35808487545E-2
YB1: 1.44473231415E-2

```

Que son las masas y composiciones de las fases refinado (*L*) y extracto (*V*). La masa de ácido acético extraída es:

$$\text{Acido acético extraído} = V_1 Y_{A1} = (103.588820587)(3.35808487545E-2) = 3.47860051679 \text{ kg}$$

Por lo tanto el porcentaje extraído es:

$$\% \text{ de Acido acético extraído} = \frac{3.47860051679}{10} * 100 = 34.79\%$$

6.2.2. Preguntas y ejercicios

1. ¿En qué consiste la operación de extracción líquido - líquido?

2. ¿Cuándo se prefiere la extracción líquido - líquido en lugar de la destilación?
3. ¿Qué se asume en el diseño y/o dimensionamiento de equipos de extracción líquido - líquido?
4. ¿Cómo se obtienen los datos de equilibrio de la extracción líquido - líquido?
5. ¿Qué son las fases refinado y extracto?
6. ¿Qué representan las líneas de unión de equilibrio en un diagrama triangular?
7. ¿Qué es una etapa de equilibrio?
8. ¿Por qué se emplea el método de interpolación lineal aun cuando no es el método de interpolación más exacto?
9. Se emplearán 50 kg de éter isopropílico puro para extraer ácido acético de una mezcla que contiene 20 kg de ácido acético y 130 kg de agua, empleando un equipo discontinuo equivalente a una etapa de equilibrio. Calcule las masas y composiciones finales de las capas refinado y extracto así como el porcentaje de ácido acético extraído.
10. En un equipo de extracción continuo, equivalente a una etapa teórica, se emplea una corriente de éter isopropílico puro para extraer ácido acético de una corriente que contiene 35% de ácido acético en agua. Al equipo se alimentan 400 kg/h de éter isopropílico y 400 kg/h de la solución de ácido acético. Calcule los flujos y composiciones de las corrientes de extracto y refinado resultantes así como el % de ácido acético extraído.
11. En un equipo discontinuo equivalente a una etapa teórica se pusieron en contacto 280 kg de una mezcla que contenía 40% de ácido acético, 10% de agua y 50% de éter isopropílico con 200 kg de una mezcla de composición desconocida. Una vez que la mezcla alcanzó el equilibrio se separaron las fases y se determinó que la fase refinado pesaba 320 kg y estaba compuesta por 29.5% de ácido acético, 66.5% de agua y 4% de éter isopropílico. Calcule la composición de los 280 kg de mezcla alimentados al equipo así como la masa y composición de la fase extracto resultante.
12. Se emplearán 500 kg de metil isobutil cetona (MIC) puro, en un equipo discontinuo equivalente a una etapa teórica, para extraer acetona de 1000 kg de una solución que contiene 23.5% en peso de acetona y 76.5% de agua. Calcule las masas y composiciones de las fases extracto y refinado resultantes así como el porcentaje de acetona extraído. Los datos de equilibrio para este sistema son los siguientes:

<i>Datos de composición (% en peso)</i>			<i>Datos de distribución (% en peso)</i>	
<i>MIC</i>	<i>Acetona</i>	<i>Agua</i>	<i>Fase agua</i>	<i>Fase MIC</i>
98.0	0.00	2.00	2.5	4.5
93.2	4.60	2.33	5.5	10.0
77.3	18.95	3.86	7.5	13.5
71.0	24.40	4.66	10.0	17.5
65.5	28.90	5.53	12.5	21.3
54.7	37.60	7.82	15.5	25.5
46.2	43.20	10.7	17.5	28.2
12.4	42.70	45.0	20.0	31.2
5.01	30.90	64.2	22.5	34.0
3.23	20.90	75.8	25.0	36.5

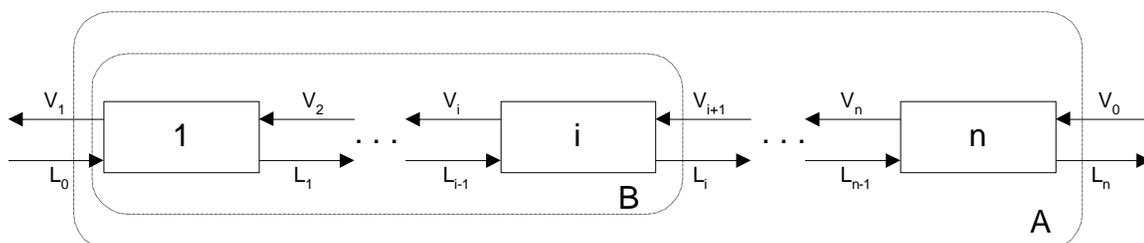
Datos de composición (% en peso)			Datos de distribución (% en peso)	
2.12	3.73	94.2	26.0	37.5
2.20	0.00	97.8		

13. Si para los datos del anterior ejercicio se emplean dos equipos de extracción donde el primero recibe los 1000 kg de solución y 250 kg de metil isobutil cetona puro y el segundo recibe la fase refinado resultante del primer equipo y 250 kg de metil isobutil cetona puro, ¿Cuál es el % de acetona extraído?. En base al resultado obtenido diga si es más eficiente o no el empleo de 2 equipos en lugar de uno.
14. En un equipo continuo se empleará agua pura para extraer ácido acético de 400 kg/h de una solución que contiene 25% de ácido acético en éter isopropílico.
- Si el equipo es equivalente a una etapa teórica y se emplean 400 kg de agua. Calcule los flujos y concentraciones de las fases extracto así como el porcentaje de ácido acético extraído.
 - Si el equipo es equivalente a un sistema de 4 etapas de equilibrio, donde a cada etapa se alimenta 100 kg de agua pura y el extracto de la etapa anterior, es decir a la primera etapa se alimenta la solución original, a la segunda el extracto de la primera, a la tercera el extracto de la segunda y a la cuarta el extracto de la tercera. Calcule los flujos y concentraciones de las fases extracto y refinado así como el porcentaje de ácido acético recuperado.

6.3. Extracción líquido - líquido en múltiples etapas de equilibrio

Con el propósito de mejorar el rendimiento en las operaciones de extracción líquido - líquido, las industrias suelen emplear una serie de equipos en contracorriente o un equipo cuyo funcionamiento es equivalente al de una serie de equipos en contracorriente.

El esquema general de este tipo de equipos es el siguiente:



Donde cada etapa es una etapa teórica (o de equilibrio), en consecuencia, las dos corrientes que salen de cada una de ellas están en equilibrio.

Como se puede observar en el esquema, en estos equipos se identifican dos sistemas: el sistema A conformado por el equipo en su conjunto y el sistema B conformado por las i primeras etapas.

El sistema A se resuelve para calcular todas las concentraciones y flujos de las corrientes que entran y salen del equipo, mientras que el sistema B se resuelve para calcular el número de etapas de equilibrio.

La situación que en lo relativo a su solución presenta el mayor grado de dificultada ocurre cuando se tiene como dato el número de etapas teóricas. En estos casos la resolución del problema es doblemente iterativo: Un proce-

so iterativo tiene lugar en el cálculo del flujo o concentración faltante, donde se deben asumir valores de ese flujo o concentración hasta que el número de etapas teóricas calculado sea igual al número de etapas conocido. El otro proceso iterativo tiene lugar en el cálculo del número de etapas pues como veremos luego es también un proceso iterativo.

La anterior situación se presenta en una industria cuando cuenta con un equipo de extracción que planifica emplear con otros flujos u otras soluciones. En esos casos resulta mucho más económico y rápido analizar todas las posibilidades mediante un programa de computadora que realizar pruebas en el equipo real.

6.3.1. Cálculo de los flujos y concentraciones de entrada y salida

Como se dijo anteriormente, para calcular los flujos y composiciones de las corrientes que entran y salen del equipo se resuelve el sistema A.

En este sistema se pueden presentar muchos casos, sin embargo el caso más frecuente es aquel donde se conocen los flujos y composiciones de las corrientes de entrada y una de las composiciones de salida (normalmente la fracción de soluto en la fase refinado X_{An}).

El balance general de materia en el sistema A es:

$$L_0 + V_0 = L_n + V_1 \quad (9)$$

Donde para el caso más frecuente se conocen los dos términos del lado izquierdo.

Los balances de materia para los componentes A (soluto) y B (solvente de la fase extracto) son:

$$L_0 x_{A0} + V_0 y_{A0} = L_n x_{An} + V_1 y_{A1} \quad (10)$$

$$L_0 x_{B0} + V_0 y_{B0} = L_n x_{Bn} + V_1 y_{B1} \quad (11)$$

Donde para el caso más frecuente se conocen todos los términos del lado izquierdo así como el valor de X_{An} .

Se tiene entonces un sistema de 3 ecuaciones con 5 incógnitas: L_n , V_1 , X_{Bn} , Y_{A1} y Y_{B1} , siendo necesarias dos ecuaciones más. Al igual que sucedía con una etapa teórica, las dos ecuaciones faltantes provienen de los datos de equilibrio. En este sistema están en equilibrio los componentes de la corriente L_n y los componentes de la corriente V_1 . Como X_{An} es un dato el valor de X_{Bn} puede ser calculado con los datos de equilibrio (ecuación 6):

$$x_B = f_{x_{AB}}(x_A) \quad (6)$$

Igualmente Y_{B1} puede ser calculado con los datos de equilibrio en función a Y_{A1} (o viceversa):

$$y_B = f_{y_{AB}}(y_A) \quad (12)$$

Donde $f_{y_{AB}}$ es una función que devuelve el valor de y_B interpolando con los datos de las columnas y_A y y_B de la tabla de equilibrio. Con estas dos ecuaciones se tiene ahora un sistema de 5 ecuaciones con 5 incógnitas.

La solución del sistema de ecuaciones que se forma en este y otros casos es casi siempre iterativa (excepto cuando se conocen las composiciones de las dos corrientes de salida).

Al igual que en una etapa de equilibrio, la forma más sencilla de resolver el sistema consiste en reordenarla como una función de una sola variable

(generalmente Y_{A1}) y resolver la función resultante con un método iterativo como el de Newton Raphson.

Así las ecuaciones (9) a (12) pueden ser colocadas como una función de Y_{A1} :

$$f(y_{A1}) = L_n + V_1 - M = 0$$

$$y_{B1} = f_{y_A y_B}(y_{A1})$$

$$\left. \begin{aligned} L_n x_{An} + V_1 y_{A1} &= C_{A0} \\ L_n x_{Bn} + V_1 y_{B1} &= C_{B0} \end{aligned} \right\} L_n = \frac{C_{A0} y_{B1} - C_{B0} y_{A1}}{x_{An} y_{B1} - x_{Bn} y_{A1}}; \quad V_1 = \frac{C_{A0} - L_n x_{An}}{y_{A1}}$$

Donde:

$$M = L_0 + V_0$$

$$C_{A0} = L_0 x_{A0} + V_0 y_{A0}$$

$$C_{B0} = L_0 x_{B0} + V_0 y_{B0}$$

$$x_{Bn} = f_{x_A x_B}(x_{An})$$

Un programa que automatiza la solución de esta función es el siguiente:

```

« 1 8 START 0 NEXT
□ L0 XA0 XB0 XAN V0 YA0 YB0 FXAXB FYAYB
  M CA0 CB0 YA1 YB1 XBN LN V1
« L0 V0 + 'M' STO
  L0 XA0 * V0 YA0 * + 'CA0' STO
  L0 XB0 * V0 YB0 * + 'CB0' STO
  XAN FXAXB EVAL 'XBN' STO
« 'YA1' STO
  YA1 FYAYB EVAL 'YB1' STO
  CA0 YB1 * CB0 YA1 * - XAN YB1 * XBN YA1 * - /
  'LN' STO
  CA0 LN XAN * - YA1 / 'V1' STO
  LN V1 + M -
» CA0 M / Newt DROP
LN "LN" □TAG XBN "XBN" □TAG
V1 "V1" □TAG YA1 "YA1" □TAG YB1 "YB1" □TAG
»
»

```

En lo sucesivo se asumirá que este programa ha sido guardado en la variable global "ELLN1". Los datos que requiere este programa y en ese orden son: el flujo y las fracciones iniciales de soluto y solvente B en la fase refinado (L_0 , x_{A0} , x_{B0}); la fracción final de soluto en la fase refinado (x_{An}); el flujo y fracciones iniciales de soluto y solvente B en la fase extracto (V_0 , y_{A0} , y_{B0}); la función de equilibrio que devuelve el valor de x_B para un valor dado de x_A ($F_{x_A x_B}$); la función de equilibrio que devuelve el valor de y_B para un valor dado de y_A ($F_{y_A y_B}$).

6.3.1.1. Ejemplo

En un equipo de múltiples etapas en contracorriente, se emplearán 600 kg/h de éter isopropílico puro para extraer ácido acético de una solución que contiene 30% en peso de ácido acético. El flujo de la solución de ácido acético es de 200 kg/h. La concentración final deseada de ácido acético en la fase extracto es del 4%. Calcule los flujos y composiciones de las fases refinado y extracto resultantes así como el porcentaje de ácido acético extraído. Los datos de equilibrio son los mismos que en el ejemplo 6.2.1.

Solución

Este problema corresponde al caso más frecuente, los datos con los que contamos son los siguientes:

```
L0 = 200 kg/h
XA0 = 0.3
XB0 = 0.7
XAn = 0.04
V0 = 600
YA0 = 0
YB0 = 0
```

Contamos además con los datos de equilibrio que deben ser almacenados en las variables globales x_A , x_B , y_A y y_B :

```
[ 0. .69 1.41 2.89 6.42 13.3 25.5 36.7 44.3 46.4 ] 100 / 'xA' STO
[ 98.8 98.1 97.1 95.5 91.7 84.4 71.1 58.9 45.1 37.1 ] 100 / 'xB' STO
[ 0. .18 .37 .79 1.93 4.82 11.4 21.6 31.1 36.2 ] 100 / 'yA' STO
[ .6 .5 .7 .8 1. 1.9 3.9 6.9 10.8 15.1 ] 100 / 'yB' STO
```

Se tiene entonces todos los datos para el programa *ELLN1*. Por lo tanto podemos resolver el problema escribiendo lo siguiente en la pila:

```
200
0.3
0.7
0.04
600
0
0
« xA xB ROT INTLP »
« yA yB ROT INTLP »
ELLN1
```

Con lo que obtenemos los resultados:

```
LN: 127.660812872
XBN: .943050991501
V1: 672.339187129
YA1: 8.16456463285E-2
YB1: 2.91658499479E-2
```

Que son los flujos y composiciones finales faltantes. La masa de ácido acético extraído es:

$$\text{Acido acético extraído} = V_1 Y_{A1} = (672.339187129)(8.16456463285E-2) = 54.8935674851 \text{ kg}$$

Y el porcentaje:

$$\% \text{ de Acido acético extraído} = \frac{V_1 Y_{A1}}{L_0 x_{A0}} * 100 = \frac{54.8935674851}{60} * 100 = 91.49\%$$

6.3.1.2. Preguntas y ejercicios

1. Si en un equipo de múltiples etapas en contracorriente tomamos la cuarta etapa ¿Qué corrientes de la misma estarían en equilibrio?
2. ¿Por qué cuando se conoce el número de etapas el cálculo es más complejo?
3. ¿En qué situación práctica se conoce el número de etapas?
4. ¿Por qué la solución del sistema de ecuaciones que se forma al resolver el sistema A es casi siempre iterativo?

5. En un equipo de múltiples etapas en contracorriente, se emplearán 600 kg/h de éter isopropílico puro para extraer ácido acético de una solución que contiene 25% en peso de ácido acético. El flujo de la solución de ácido acético es de 200 kg/h. La concentración final deseada de ácido acético en la fase extracto es del 3%. Calcule los flujos y composiciones de las fases refinado y extracto resultantes así como el porcentaje de ácido acético extraído. Los datos de equilibrio son los mismos que en el ejemplo 6.2.1.
6. En un equipo de múltiples etapas en contracorriente, se emplearán 800 kg/h de éter isopropílico que contiene 2% de ácido acético para extraer ácido acético de una solución que contiene 32% en peso de ácido acético. El flujo de la solución de ácido acético es de 300 kg/h. La concentración final deseada de ácido acético en la fase extracto es del 1%. Calcule los flujos y composiciones de las fases refinado y extracto resultantes así como el porcentaje de ácido acético extraído. Los datos de equilibrio son los mismos que en el ejemplo 6.2.1.

6.3.2. Cálculo del número de etapas teóricas

Una vez calculadas todos los flujos y concentraciones que entran y salen del equipo, es decir una vez resuelto el sistema A, se puede determinar el número de etapas teóricas resolviendo el sistema B.

El balance general de materia en este sistema es:

$$L_0 + V_{i+1} = L_i + V_i \quad (13)$$

Donde los flujos L_0 y V_i han sido calculados al resolver el sistema A y son por lo tanto valores conocidos. Por conveniencia agrupamos los términos conocidos de la ecuación (13) en el lado derecho:

$$L_i - V_{i+1} = L_0 - V_i \quad (13.a)$$

De manera similar, del balance de materia para los componentes A (solute) y B (solvente de la fase refinado) resulta:

$$L_i x_{Ai} - V_{i+1} y_{Ai+1} = L_0 x_{A0} - V_1 y_{A1} \quad (14)$$

$$L_i x_{Bi} - V_{i+1} y_{Bi+1} = L_0 x_{B0} - V_1 y_{B1} \quad (15)$$

Donde igualmente son valores conocidos todos los términos que se encuentra al lado derecho de ambas ecuaciones.

Puesto que el sistema B involucra las primeras etapas del equipo, los cálculos deben comenzar en la primera etapa (para comenzar los cálculos en la última etapa el sistema debería estar conformado por las últimas etapas del equipo).

Entonces, comenzando en la primera etapa ($i=1$), el valor de y_{A1} (y_{A1}) es un dato conocido y puesto que también son conocidos los datos de equilibrio el valor de x_{A1} (y_{A1}) puede ser calculado con la ecuación:

$$x_{Ai} = f_{y_{Ax}A}(y_{Ai}) \quad (16)$$

Donde $f_{y_{Ax}A}$ es la función de equilibrio que devuelve el valor de x_{Ai} para un valor conocido de y_{Ai} (empleando los datos de las columnas y_A y x_A de la tabla de equilibrio).

Igualmente con y_{A1} (y_{A1}) podemos calcular el valor de x_{B1} (x_{B1}) con:

$$x_{Bi} = f_{y_{Ax}B}(y_{Ai}) \quad (17)$$

Donde $f_{y_{Ax}B}$ es la función de equilibrio que devuelve el valor de x_{Bi} para un valor conocido de y_{Ai} (empleando los datos de las columnas y_A y x_B de la tabla de equilibrio).

Finalmente las fracciones y_{Ai+1} y y_{Bi+1} están también en equilibrio y se relacionan entre si mediante la ecuación (12):

$$y_{Bi+1} = f_{y_{Ay}B}(y_{Ai+1}) \quad (12)$$

Se tiene entonces un sistema de 6 ecuaciones (ecuaciones 12 a 17) con 6 incógnitas ($L_i, V_{i+1}, x_{Ai}, x_{Bi}, y_{Ai+1}, y_{Bi+1}$). Una vez más (debido a las funciones de equilibrio) la solución es iterativa y la forma más sencilla de resolverla es colocarla en forma de una función de una variable (y_{Ai+1}):

$$\begin{aligned} f(y_{Ai+1}) &= L_i - V_{i+1} - D_M = 0 \\ y_{Bi+1} &= f_{y_{Ay}B}(y_{Ai+1}) \\ \left. \begin{aligned} L_i x_{Ai} - V_{i+1} y_{Ai+1} &= D_A \\ L_i x_{Bi} - V_{i+1} y_{Bi+1} &= D_B \end{aligned} \right\} L_i &= \frac{D_A y_{Bi+1} - D_B y_{Ai+1}}{x_{Ai} y_{Bi+1} - x_{Bi} y_{Ai+1}}; \quad V_{i+1} = \frac{L_i x_{Ai} - D_A}{y_{Ai+1}} \end{aligned}$$

Donde:

$$\begin{aligned} D_M &= L_0 - V_1 \\ D_A &= L_0 x_{A0} - V_1 y_{A1} \\ D_B &= L_0 x_{B0} - V_1 y_{B1} \\ x_{Ai} &= f_{y_{Ax}A}(y_{Ai}) \\ x_{Bi} &= f_{y_{Ax}B}(y_{Ai}) \end{aligned} \quad (18)$$

La ecuación (18) se resuelve inicialmente para la primera etapa $i=1$, donde como ya se explicó, se conoce y_{Ai} (y_{A1}), obteniéndose así el valor de y_{Ai+1} (y_{A2}). Entonces se puede resolver la ecuación (18) para la segunda etapa $i=2$, pues ahora se conoce el valor de y_{A2} (y_{A2}), de esta manera se obtiene y_{A3} . Con y_{A3} se resuelve la ecuación (18) para la tercera etapa obteniéndose y_{A4} entonces se pasar a la cuarta etapa y así sucesivamente hasta que el valor de x_{Ai} sea menor o igual a x_{An} .

Un programa que automatiza el procedimiento antes descrito es el siguiente (donde J representa a I+1):

```

« 1 12 START 0 NEXT
□ L0 XA0 XB0 XAN V1 YA1 YB1 FYAYB FYAXA FYAXB
DM DA DB N XAI XBI YAI YBI YAJ YBJ LI VJ
« CLLCD L0 V1 - 'DM' STO
L0 XA0 * V1 YA1 * - 'DA' STO
L0 XB0 * V1 YB1 * - 'DB' STO YA1 'YAI' STO
DO
YAI FYAXA EVAL 'XAI' STO
IF XAI XAN ≤ THEN 1
ELSE XAI "XA" N + □TAG 3 DISP
YAI FYAXB EVAL 'XBI' STO
« 'YAJ' STO YAJ FYAYB EVAL 'YBJ' STO
DA YBJ * DB YAJ * - XAI YBJ * XBI YAJ * - /
'LI' STO LI XAI * DA - YAJ / 'VJ' STO
LI VJ - DM - » DUP YAI 0.5 * Newt
SWAP EVAL DROP YAJ 'YAI' STO
XAI 'XA0' STO 'N' 1 STO+ 0
END
UNTIL END N XAN XA0 - XAI XA0 - / +
»
»

```

En lo sucesivo se asumirá que este programa ha sido guardado en la variable global "ELLN". Los datos que requiere el programa y en ese orden son: los flujos y fracciones iniciales de la fase refinado (L_0 , X_{A0} , X_{B0}); la fracción final de soluto en la fase refinado (X_{An}); los flujos y fracciones finales de en la fase extracto (V_1 , Y_{A1} , Y_{B1}); la función de equilibrio que devuelve la fracción y_B para un valor dado de y_A ($FYAYB$); la función de equilibrio que devuelve la fracción x_A para un valor dado de y_A ($FYAXA$) y la función de equilibrio que devuelve la fracción x_B para un valor dado de y_A ($FYAXB$). Devuelve el número de etapas de equilibrio necesarias para lograr la separación deseada.

6.3.2.1. Ejemplos

1. Para los datos del ejemplo 6.3.1.1, calcule el número de etapas para lograr la separación requerida.

Solución

De la solución al sistema A, llevada a cabo en el ejemplo 6.3.1.1, contamos con todos los datos para el programa ELLN:

```
L0 = 200 kg/h
XA0 = 0.3
XB0 = 0.7
XAn = 0.04
V1 = 672.339187129 kg/h
YA1 = 8.16456463285E-2
YB1 = 2.91658499479E-2
```

Tenemos además los datos de equilibrio, que deben estar almacenadas en las variables globales x_A , x_B , y_A y y_B . Entonces resolvemos el problema escribiendo los siguientes datos en la pila:

```
200
0.3
0.7
0.04
672.339187129
8.16456463285E-2
2.91658499479E-2
« yA yB ROT INTLP »
« yA xA ROT INTLP »
« yA xB ROT INTLP »
ELLN
```

Con lo que se obtiene:

```
3.65861512238
```

Que es el número de etapas de equilibrio necesarias para lograr la separación requerida.

2. Repita el ejemplo anterior pero para un flujo de 450 kg/h de éter isopropílico puro, un flujo de solución igual a 150 kg/h y una fracción final de ácido acético en la fase refinado igual a 10%.

Solución

Los datos con los que contamos son:

```
L0 = 150 kg/h
XA0 = 0.3
XB0 = 0.7
XAn = 0.1
```

$V_0 = 450$
 $Y_{A0} = 0$
 $Y_{B0} = 0$

Además de los datos de equilibrio que deben estar almacenados en las variables globales x_A , x_B , y_A y y_B .

Resolvemos primero el sistema A escribiendo los siguientes datos en la pila:

```

150
0.3
0.7
0.1
450
0
0
« xA xB ROT INTLP »
« yA yB ROT INTLP »
ELLN1

```

Con lo que obtenemos:

```

LN: 105.075955966
XBN: .879014534884
V1: 494.924044034
YA1: 6.96923190926E-2
YB1: .025532619785

```

Con estos resultados podemos resolver el sistema B y calcular así el número de etapas:

```

150
0.3
0.7
0.1
494.924044034
6.96923190926E-2
0.025532619785
« yA yB ROT INTLP »
« yA xA ROT INTLP »
« yA xB ROT INTLP »
ELLN

```

Resultando:

1.72267591078

Que es el número de etapas requeridas para lograr la separación.

- En un equipo de múltiples etapas en contracorriente, se empleará querosén para extraer nicotina de 100 kg/h de una solución acuosa que contiene 0.01 en fracción en peso de nicotina. El querosén contiene 0.0005 de fracción en peso de nicotina y su flujo es de 200 kg/h. Se desea reducirá la concentración de nicotina en el agua a 0.001 de fracción en peso. Los datos de equilibrio son los siguientes:

<i>Fracción de soluto en la fase acuosa</i>	<i>Fracción de soluto en la fase querosén</i>
0.00101	0.000806
0.00246	0.001959
0.00500	0.004540
0.00746	0.006820

0.00988	0.009040
0.02020	0.018500

Calcule el número de etapas teóricas necesarias para lograr la separación requerida.

Solución

Este problema corresponde también al caso más frecuente. Los datos con los que contamos son:

$L_0 = 100$ kg/h
 $X_{A0} = 0.01$
 $X_{B0} = 0.99$
 $X_{An} = 0.001$
 $V_0 = 200$ kg/h
 $Y_{A0} = 0.0005$
 $Y_{B0} = 0$

Como sólo existen dos columnas en la tabla de equilibrio debemos asumir que los solventes de las dos fases son completamente inmiscibles y que en consecuencia sólo existen dos componentes en cualquier instante. Por lo tanto solo se requiere una de las fracciones pues la suma de las dos debe ser igual a 1.

Almacenamos los datos de equilibrio en las variables globales x_A , x_B , y_A y y_B :

```
[ 0.00101 0.00246 0.00500 0.00746 0.00988 0.02020 ] 'xA' STO
xA AXL 1 SWAP - AXL 'xB' STO
[ 0.000806 0.001959 0.004540 0.006820 0.009040 0.018500 ] 'yA' STO
{ 6 } 0 CON 'yB' STO
```

Entonces resolvemos el sistema A:

```
100
0.01
0.99
0.001
200
0.0005
0
« xA xB ROT INTLP »
« yA yB ROT INTLP »
ELLN1
```

Resultando:

```
LN: 99.0990990992
XBN: .999
V1: 200.900901013
YA1: 4.98206277749E-3
YB1: 0.
```

Ahora resolvemos el sistema B:

```
100
0.01
0.99
0.001
200.900901013
4.98206277749E-3
0
« yA yB ROT INTLP »
```

« yA xA ROT INTLP »
 « yA xB ROT INTLP »
 ELLN

Resultando:

4.53828703704

Que es el número de etapas teóricas necesarias para lograr la separación requerida.

6.3.2.2. Flujo mínimo de la fase extracto

A medida que disminuye el flujo de la fase extracto aumenta el número de etapas requeridas para lograr una determinada separación, este incremento continúa hasta que para un flujo inicial dado el número de etapas requeridas se hace infinito. El flujo para el cual el número de etapas se hace infinito es el flujo mínimo de la fase extracto.

Por supuesto en un equipo real se debe emplear un flujo inicial superior al mínimo, por lo general al menos 1.5 veces este flujo.

Cuando no se conoce el flujo inicial de extracto, es importante calcular el flujo mínimo para fijar un valor por encima de este. El flujo mínimo puede ser calculado por prueba y error con los programas *ELLN1* y *ELLN*, disminuyendo el flujo inicial de extracto (L_0) hasta que no es posible calcular el número de etapas por ser un número infinito.

Alternativamente, como el número de etapas es infinito cuando la fracción final de soluto en la fase extracto (y_{A1}) está en equilibrio con la fracción inicial de soluto en la fase refinado (x_{A0}), el flujo mínimo de refinado puede ser calculado directamente con los datos de equilibrio y los balances de materia.

La fracción final de soluto en la fase extracto se calcula con la función de equilibrio dada por la ecuación (7):

$$y_{A1} = f_{xAyA}(x_{A0}) \quad (7)$$

Igualmente la fracción final de componente B (solvente de la fase extracto) se calcula con la función de equilibrio dada por la ecuación (8):

$$y_{B1} = f_{xAyB}(x_{A0}) \quad (8)$$

Entonces las ecuaciones del balance de materia (ecuaciones 9 conforman un sistema de 3 ecuaciones con 3 incógnitas (L_n, V_0, V_1):

$$L_n = L_0 + V_0 - V_1 \quad (9.a)$$

$$L_n x_{An} + V_1 y_{A1} - V_0 y_{A0} = L_0 x_{A0} \quad (10.a)$$

$$L_n x_{Bn} + V_1 y_{B1} - V_0 y_{B0} = L_0 x_{B0} \quad (11.a)$$

Reemplazando la ecuación (9.a) en las ecuaciones (10.a) y (11.a) se forma un sistema de 2 ecuaciones con 2 incógnitas:

$$\begin{aligned} (L_0 + V_0 - V_1)x_{An} + V_1 y_{A1} - V_0 y_{A0} &= L_0 x_{A0} \\ (L_0 + V_0 - V_1)x_{Bn} + V_1 y_{B1} - V_0 y_{B0} &= L_0 x_{B0} \\ V_1(y_{A1} - x_{An}) + V_0(x_{An} - y_{A0}) &= L_0(x_{A0} - x_{An}) \\ V_1(y_{B1} - x_{Bn}) + V_0(x_{Bn} - y_{B0}) &= L_0(x_{B0} - x_{Bn}) \end{aligned} \quad (19)$$

De donde obtenemos el flujo mínimo de extracto (V_0):

$$V_{min} = \frac{(y_{A1} - x_{An})L_0(x_{B0} - x_{Bn}) - (y_{B1} - x_{Bn})L_0(x_{A0} - x_{An})}{(y_{A1} - x_{An})(x_{Bn} - y_{B0}) - (y_{B1} - x_{Bn})(x_{An} - y_{A0})} \quad (20)$$

Un programa que automatiza el cálculo de esta ecuación es el siguiente:

```

« 0 0 0
□ L0 XA0 XB0 XAN YA0 YB0 FXAXB FXAYA FXAYB YA1 YB1 XBN
« XAN FXAXB EVAL 'XBN' STO
  XA0 FXAYA EVAL 'YA1' STO
  XA0 FXAYB EVAL 'YB1' STO
  YA1 XAN - L0 XB0 XBN - * * YB1 XBN - L0 XA0 XAN - * * -
  YA1 XAN - XBN YB0 - * YB1 XBN - XAN YA0 - * - /
  "Vmin" □TAG
»
»
»

```

En lo sucesivo se asumirá que este programa ha sido guardado con el nombre "ELMIN". Los datos que requiere este programa y en ese orden son: el flujo inicial de refinado (L_0); las fracciones iniciales de soluto y solvente B en la fase refinado (x_{A0} , x_{B0}); la fracción final de soluto en la fase refinado (x_{An}); las fracciones iniciales de soluto y solvente B en la fase refinado (y_{A0} , y_{B0}); la función de equilibrio que devuelve el valor de x_B para un valor dado de x_A (F_{xAxB}); la función de equilibrio que devuelve el valor de y_A para un valor dado de x_A (F_{xAyA}) y la función de equilibrio que devuelve el valor de y_B para un valor dado de x_A (F_{xAyB}). Devuelve el flujo mínimo de extracto que se puede emplear para lograr la separación deseada.

Empleando este programa podemos calcular por ejemplo el flujo mínimo de extracto que se puede emplear en el ejemplo 3 del acápite 6.3.2.1:

```

100
0.01
0.99
0.001
0.0005
0.0
« xA xB ROT INTLP »
« xA yA ROT INTLP »
« xA yB ROT INTLP »
ELMIN

```

Resultando:

Vmin: 103.19741707

Que es el flujo mínimo inicial de refinado en kg/h que se puede emplear en el equipo. Como se dijo, en la práctica el flujo deberá ser al menos 1.5 veces este valor, es decir unos 155 kg/h.

Los equipos de múltiples etapas en contracorriente pueden estar constituidos por una serie de equipos provistos de un mezclador y un decantador. En estos casos cada equipo suele ser equivalentes a una etapa de equilibrio.

Otros equipos están conformados por torres de platos o rellenas donde el fluido más denso se alimenta por la cabeza de la columna y el menos denso por el fondo. La mezcla se consigue con paletas o mediante platos perforados por donde pasa el fluido menos denso. Una descripción detallada de estos equipos puede ser consultada en el Manual del Ingeniero Químico de J. H. Perry.

6.3.2.3. Preguntas y ejercicios

1. Escriba las ecuaciones del balance de materia cuando se toma como sistema B las últimas etapas del equipo.
2. ¿Qué es el flujo mínimo inicial de extracto?
3. A un equipo de múltiples etapas en contracorriente se alimentan 1000 kg/h de una solución acuosa que contiene 23.5% en peso de acetona y 76.5% en peso de agua. Como extractor se emplean metil-isobutil-cetona pura a 298-299 K. Se desea que el porcentaje final de acetona en el refinado sea del 2.5%. Los datos de equilibrio son los del ejercicio 12 del acápite 6.2.2.
 - a) Calcule el flujo mínimo inicial de refinado que podría emplearse.
 - b) Calcule el número de etapas teóricas requeridas si el flujo inicial de refinado es 1.5 veces el flujo mínimo.
4. Se empleará un equipo de múltiples etapas en contracorriente para extraer con éter isopropílico ácido acético de una solución acuosa que contiene 30% en peso de ácido acético. Al equipo se alimenta 1000 kg/h de la solución acuosa y se desea que su concentración sea reducida hasta un 2% en peso. Los datos de equilibrio son los del ejemplo 6.2.1.
 - a) Calcule el flujo mínimo inicial de refinado que podría emplearse.
 - b) Calcule el número de etapas teóricas si el flujo inicial de refinado es de 2500 kg/h.
5. Repita el ejercicio 4 si la concentración final de ácido acético en el refinado debe ser del 4% en peso.
6. Se empleará un equipo de múltiples etapas en contracorriente para extraer con querosén nicotina de una solución acuosa que contiene 1.5% en peso de nicotina. Al equipo se alimentan 2000 kg/h de querosén que contiene 0.05% de nicotina y 1000 kg/h de la solución acuosa. En el proceso se debe extraer el 90% de la nicotina contenida en la solución acuosa. Los datos de equilibrio son los del ejemplo 3 del acápite 6.3.2.1.
7. Determine el flujo mínimo de querosén que se podría emplear en el ejercicio 6 y calcule el número de etapas teóricas si se emplea 1.25 veces el flujo mínimo.
8. Se empleará un equipo de múltiples etapas en contracorriente para extraer con agua pura el 90% de la nicotina contenida en una solución de querosén que contiene 1.4% de nicotina. Se empleará un flujo de agua igual a 1.5 veces el flujo mínimo. Calcule el número de etapas teóricas necesarias para lograr la separación requerida.

7. LIXIVIACIÓN

En esta operación, conocida también como extracción sólido - líquido, se ponen en contacto una fase sólida con una fase líquida. La fase sólida contiene uno o más compuestos que son solubles en la fase líquida. Como resultado de esta operación parte del compuesto o compuestos solubles (el soluto) pasan a la fase líquida lográndose así extraer estos compuestos de la fase sólida.

Esta operación es ampliamente empleada en la extracción de aceites naturales así como en la extracción de minerales. Es por ejemplo la operación más importante en las fábricas de aceites vegetales, donde se extrae por lixiviación el aceite contenido en las semillas de algodón, girasol, maíz y de otros vegetales. Es igualmente la operación más importante en la extracción de oro cuando dicho mineral se encuentra mezclado con otros sólidos en forma de partículas microscópicas.

La Lixiviación se emplea exhaustivamente también en la industria farmacéutica, donde la mayor parte de los principios activos de las plantas y animales son extraídos mediante esta operación.

Para esta operación existe una gran variedad de equipos. En todos ellos se busca la manera más eficiente de lograr el mejor contacto posible entre las dos fases y la separación posterior de las mismas. En general el contacto se consigue de dos maneras: a) empleando agitadores y b) haciendo pasar el solvente a través de un lecho fijo del sólido. La separación de las dos fases se lleva a cabo por decantación, filtración y/o centrifugación. Una descripción detallada de algunos de estos equipos puede ser encontrada en el Manual del Ingeniero Químico de J. H. Perry.

7.1. Datos de equilibrio

Al igual que en la operación de extracción líquido - líquido, en la lixiviación, se asumen que sólo existen tres componentes: *A*: el soluto, *B*: el sólido inerte y *C*: el solvente. Estos tres componentes pueden corresponder cada uno a tres compuestos simples o como ocurre con mayor frecuencia a mezclas multicomponentes.

Una vez más para que los diseños puedan ser empleados en la práctica es necesario que los datos de equilibrio sean determinados experimentalmente. Para ello se ponen en contacto cantidades conocidas de las dos fases, empleando un mecanismo similar al que se empleará en el equipo industrial (agitación o paso del solvente a través del lecho relleno). El contacto debe ser mantenido por un tiempo aproximadamente igual al del equipo industrial.

Una vez realizado el contacto se separan las dos fases quedando un sólido húmedo y la solución. Entonces se analizan las composiciones de la solución, la del sólido y la de la solución que humedece al sólido. Se determina también la cantidad de solución que humedece al sólido. Los resultados de estos análisis constituyen una fila en la tabla de equilibrio (o dos puntos en la gráfica de equilibrio), siendo necesario en consecuencia realizar varias corridas experimentales para construir la tabla o gráfica de equilibrio.

Tradicionalmente en lixiviación los datos de equilibrio se expresan considerando la masa de sólido inerte con relación a la masa de solución (libre de sólido inerte). Con este fin se define "*N*" como la razón de la masa de sólido inerte a la masa de solución (sin sólido inerte):

$$N = \frac{\text{kg de sólido inerte}}{\text{kg de solución}} = \frac{\text{kg de B}}{\text{kg de A} + \text{kg de C}} \quad (1)$$

Al igual que sucede con las fracciones en peso, existe un valor de "N", al cual denominaremos "N_y", para la fase sólida (sólido húmedo):

$$N_y = \frac{\text{kg de sólido inerte}}{\text{kg de solución}} = \frac{\text{kg de B}}{\text{kg de A} + \text{kg de C}} \quad (\text{fase sólida}) \quad (2)$$

Y otro, al cual denominaremos "N_x", para la fase líquida:

$$N_x = \frac{\text{kg de sólido inerte}}{\text{kg de solución}} = \frac{\text{kg de B}}{\text{kg de A} + \text{kg de C}} \quad (\text{fase líquida}) \quad (3)$$

En la mayoría de los casos la cantidad de sólido inerte retenida en la fase líquida es despreciable, de manera que el valor de N_x es con frecuencia cero. Esta es la razón por la cual la mayoría de los datos de equilibrio que se encuentran en bibliografía sólo tienen datos para un parámetro general "N" el cual corresponde a N_y.

Para las concentraciones de la solución, que en lixiviación está conformada únicamente por el soluto (A) y el solvente (C), se emplean fracciones en peso. Puesto que existen dos soluciones: la solución de la fase sólida y la solución de la fase líquida, existen dos fracciones.

A la fracción de soluto en la solución de la fase sólida se denomina y^{*}_A:

$$y_A^* = \frac{\text{kg de soluto}}{\text{kg de solución}} = \frac{\text{kg de A}}{\text{kg de A} + \text{kg de C}} \quad (\text{fase sólida}) \quad (4)$$

A la fracción de de soluto en la solución de la fase líquida se denomina x^{*}_A:

$$x_A^* = \frac{\text{kg de soluto}}{\text{kg de solución}} = \frac{\text{kg de A}}{\text{kg de A} + \text{kg de C}} \quad (\text{fase líquida}) \quad (5)$$

Cuando los datos de equilibrio se expresan de manera gráfica tienen la forma que se muestra en la figura...

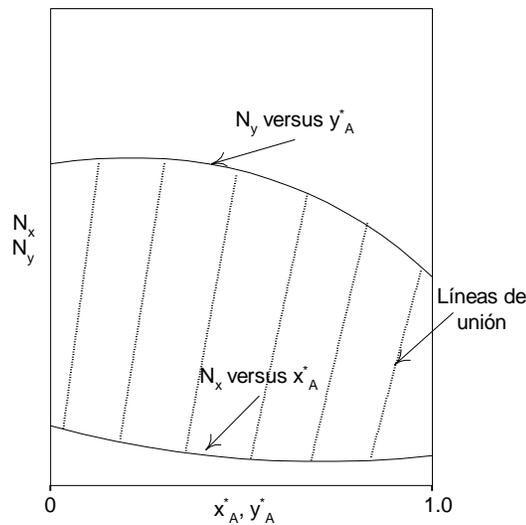


Figura 5.1. Representación de gráfica de los datos de equilibrio en la operación de Lixiviación

Como se puede observar en esta gráfica la curva superior corresponde a la fase sólida y la inferior a la fase líquida. Las líneas de unión unen los puntos de las fases sólida y líquida correspondientes a una corrida experimental y corresponden a una fila de la tabla de equilibrio.

Cuando los datos se expresan en forma tabular constan de las cuatro columnas que se muestran en la siguiente tabla:

Fase sólida		Fase líquida	
N_y	y_A^*	N_x	x_A^*
4	0.1	0.05	0.09
3	0.5	0.04	0.49
.

Como ya se dijo los datos de equilibrio disponibles generalmente no incluyen la columna N_x , pues los valores de la misma son cero o casi cero. Por otra parte, las composiciones de las soluciones de las fases sólida (y_A^*) y líquida (x_A^*) son diferentes sólo si la solución de la fase sólida ha tenido tiempo suficiente como para disolver más soluto. Puesto que esto sólo ocurre en raras ocasiones generalmente las composiciones de las dos soluciones son iguales.

Entonces para los casos más frecuentes las tablas de equilibrio solo cuentan con dos columnas: N que corresponde a N_y y y_A^* que también es el valor de x_A^* :

N (N_y)	y_A^* (x_A^*)
4	0.1
3	0.5
.

Igualmente en su forma gráfica los datos de equilibrio sólo presentan una curva, la correspondiente a la fase sólida y no existen líneas de unión pues las mismas son verticales:

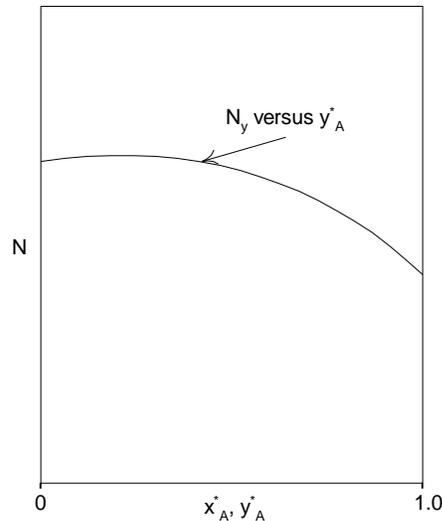


Figura 5.2. Representación gráfica más frecuente de los datos de equilibrio en la operación de lixiviación.

Puesto que los datos de equilibrio presentados en forma gráfica pueden ser transformados a su forma tabular con relativa facilidad (leyendo algunos valores de la gráfica), en lo sucesivo se asumirá que los datos de equilibrio se encuentran en la forma tabular. En los casos extremos donde sólo se conoce un valor de "N", se asume que la curva de equilibrio es una línea

recta y que los valores de y_A^* y x_A^* son iguales. Para estos casos la forma tabular tiene dos filas con el mismo valor de "N" y dos valores cualesquiera de y_A^* (comprendidos entre 0 y 1).

7.2. Lixiviación en una etapa de equilibrio

Con excepción de la forma en que se presentan los datos de equilibrio, los balances de materia y cálculos involucrados en la operación de lixiviación son las mismas que en la extracción líquido - líquido: ambas tienen tres componentes y dos fases. Por lo tanto en la lixiviación se pueden emplear las mismas ecuaciones y programas que en la extracción líquido - líquido, por supuesto es necesario convertir los datos de equilibrio de su forma tradicional a fracciones en peso.

Para llevar los datos de equilibrio de su forma tradicional a fracciones en peso tomamos como base 1 kg de solución, entonces la masa de sólido inerte será igual a N_y para la fase sólida y a N_x para la fase líquida:

$$N_y = \frac{\text{kg de B}}{\text{kg de A} + \text{kg de C}} = \frac{\text{kg de B}}{1 \text{ kg de solución}} \Rightarrow \text{kg de B} = N_y \quad (\text{fase sólida}) \quad (6)$$

$$N_x = \frac{\text{kg de B}}{\text{kg de A} + \text{kg de C}} = \frac{\text{kg de B}}{1 \text{ kg de solución}} \Rightarrow \text{kg de B} = N_x \quad (\text{fase líquida}) \quad (7)$$

Por lo tanto la fracción en peso de sólido inerte en la fase sólida, que en extracción líquido - líquido corresponde a la fase refinado (x_B), se calcula con:

$$x_B = \frac{\text{kg de B}}{\text{kg de A} + \text{kg de C} + \text{kg de B}} = \frac{N_y}{1 + N_y} \quad (8)$$

Y la fracción en peso de sólido inerte en la fase líquida, que en extracción líquido - líquido corresponde a la fase extracto (y_B) con:

$$y_B = \frac{\text{kg de B}}{\text{kg de A} + \text{kg de C} + \text{kg de B}} = \frac{N_x}{1 + N_x} \quad (9)$$

Igualmente, con la misma base (1 kg de solución) la masa de soluto en las fases sólida y líquida son respectivamente y_A^* y x_A^* :

$$y_A^* = \frac{\text{kg de A}}{\text{kg de A} + \text{kg de C}} = \frac{\text{kg de A}}{1 \text{ kg de solución}} \Rightarrow \text{kg de A} = y_A^* \quad (\text{fase sólida}) \quad (10)$$

$$x_A^* = \frac{\text{kg de A}}{\text{kg de A} + \text{kg de C}} = \frac{\text{kg de A}}{1 \text{ kg de solución}} \Rightarrow \text{kg de A} = x_A^* \quad (\text{fase líquida}) \quad (11)$$

Entonces la fracción en peso en la fase sólida, correspondiente a la fase refinado en la extracción líquido - líquido (x_A), se calcula con:

$$x_A = \frac{\text{kg de A}}{\text{kg de solución} + \text{kg de sólido inerte}} = \frac{y_A^*}{1 + N_y} \quad (12)$$

Y para la fase refinado con:

$$y_A = \frac{\text{kg de A}}{\text{kg de solución} + \text{kg de sólido inerte}} = \frac{x_A^*}{1 + N_x} \quad (13)$$

Un programa que realiza estos cálculos y guarda los resultados en las variables globales vxa , vxb , vya y vyb que son las variables empleadas en extracción líquido - líquido es el siguiente:

```

« □ NY YAA NX XAA
  « NY ARRY□ 1 GET □LIST DUP 1 ADD DUP
    3 ROLL / LIST□ □ARRY 'vxb' STO
    YAA ARRY□ 1 GET □LIST SWAP /
    LIST□ □ARRY 'vxa' STO
    NX ARRY□ 1 GET □LIST DUP 1 ADD DUP
    3 ROLL / LIST□ □ARRY 'vyb' STO
    XAA ARRY□ 1 GET □LIST SWAP /
    LIST□ □ARRY 'vya' STO
  »
»
»

```

En lo sucesivo se asumirá que este programa ha sido guardado en la variable global *LIX1*. Los datos que recibe este programa y en ese orden son: el vector con las fracciones de sólido inerte en la fase sólida (N_y); el vector con las fracciones de soluto en la solución de la fase sólida (y^*_A); el vector con las fracciones de sólido inerte en la fase líquida (N_y) y el vector con las fracciones de soluto en la solución de la fase líquida (y^*_B).

7.2.1. Ejemplo

En un equipo equivalente a una etapa de equilibrio se extraerá aceite, de hojuelas de soya, empleando hexano como disolvente. Al equipo se alimentan 100 kg de soya conteniendo un 20% de aceite y 100 kg de solvente puro. El valor de N para la fase sólida es esencialmente constante e igual a 1.5 kg de sólido insoluble/kg de solución. Calcule las cantidades y composiciones de las corrientes resultantes así como el % de aceite extraído.

Solución

Los datos con los que se cuentan son:

```

L0 = 100 kg
xA0 = 0.2
xB0 = 0.8
V0 = 100 kg
yA0 = 0.0
yB0 = 0.0

```

Siendo los datos de equilibrio:

```

Ny = [ 1.5 1.5 ]
y*_A = [ 0.1 0.9 ]
Nx = [ 0 0 ]
x*_A = [ 0.1 0.9 ]

```

Convertimos primero los datos de equilibrio en fracciones en peso empleando el programa *LIX1*:

```

[ 1.5 1.5 ]
[ 0.1 0.9 ]
[ 0 0 ]
[ 0.1 0.9 ]
LIX1

```

Con lo que los datos de equilibrio quedan guardados en las variables x_A , x_B , y_A y y_B . Entonces empleamos el programa *ELL1*:

```

0.2
0.8
100
0
0
« vxa vxb fun1 »
« vxa vya fun1 »
« vxa vyb fun1 »
ELL1

```

Obteniéndose así los resultados:

```

L1: 133.333333333
XA1: 6.66666666511E-2
XB1: .6
V1: 66.6666666947
YA1: .166666666628
YB1: 0.

```

Que son los flujos y composiciones de finales de las fases sólida (L) y líquida (V). Como se puede observar una buena parte del solvente (33.33 kg) se queda en la fase sólida. El porcentaje de soluto extraído es:

$$\% \text{ de aceite extraído} = \frac{V_1 Y_{A1}}{L_0 X_{A0}} * 100 = \frac{(66.6666666947)(0.166666666628)}{(100)(0.2)} * 100 = 55.56\%$$

7.2.2. Preguntas y ejercicios

1. ¿En qué consiste la operación de lixiviación?
2. ¿Qué operaciones podría emplear para la extracción del aceite esencial de las rosas?
3. En general ¿Cómo se logra el contacto entre las dos fases en la operación de lixiviación?
4. ¿Qué se asume para el diseño y dimensionamiento de los equipos de lixiviación?
5. ¿Cómo se determinan los datos de equilibrio en la operación de lixiviación?
6. Cuando los datos de equilibrio sólo cuentan con datos para un parámetro general "N" ¿A qué fase corresponde dicho parámetro?
7. Cuando los datos de equilibrio sólo cuentan con datos para un parámetro general "N" y una fracción en peso ¿Cuáles son los valores de N_y , N_x , y^*_A y x^*_A ?
8. Cuando los datos de equilibrio sólo constan de un valor para "N" ¿Cuál es la tabla de equilibrio?
9. ¿Qué se debe hacer para emplear las ecuaciones y programas correspondientes a la extracción líquido - líquido en lixiviación?
10. Repita el ejemplo pero considerando que los 100 kg de soya tienen un contenido de 22% de aceite y que se emplean 80 kg de solvente con un contenido de aceite igual al 3%.
11. La fase sólida (húmeda) resultante de una operación de lixiviación en un equipo equivalente a una etapa teórica pesa 100 kg y contiene 75 kg de sólido inerte y 25 kg de solución. La solución tiene un 10% en peso de aceite y un 90% de hexano. Este sólido húmedo es puesto en contacto con 100 kg de hexano puro en un equipo equivalente a una etapa de equi-

librio. El valor de N es esencialmente constante e igual a 1.5 kg de sólido inerte/kg de solución. Calcule los flujos y composiciones de las corrientes de salida así como el porcentaje de aceite extraído.

7.3. Lixiviación en múltiples etapas en contracorriente

Al igual que en una etapa de equilibrio, en múltiples etapas en contracorriente se pueden emplear las ecuaciones y programas correspondientes a la extracción líquido - líquido. Como antes se deben convertir previamente los datos de equilibrio de su forma tradicional a fracciones en peso (empleando el programa *LIX1*).

7.3.1. Ejemplo

En un equipo de lixiviación de múltiples etapas en contracorriente se lixiviará aceite de harina de soya empleando benceno como disolvente. Al equipo ingresan 2000 kg/h de sólido inerte con 800 kg/h de aceite y 50 kg/h de benceno. Por el lado del solvente ingresan 1310 kg/h de hexano y 20 kg/h de aceite. La corriente de sólido húmedo resultante deberá contener 120 kg/h de aceite. Los datos de equilibrio obtenidos en un equipo experimental que opera en condiciones similares a las del equipo industrial son los siguientes:

N	Y^*_A
2.00	0.0
1.98	0.1
1.94	0.2
1.89	0.3
1.82	0.4
1.75	0.5
1.68	0.6
1.61	0.7

Calcule los flujos y concentraciones de las corrientes resultantes así como el número de etapas requeridas.

Solución

Los datos con los que se cuenta son:

$$\begin{aligned} L_0 &= 2850 \text{ kg/h} \\ x_{A0} &= 800/2850 \\ x_{B0} &= 2000/2850 \\ L_n x_{An} &= 120 \text{ kg/h} \\ V_0 &= 1330 \\ y_{A0} &= 20/1330 \\ y_{B0} &= 0.0 \end{aligned}$$

Las ecuaciones del balance de materia (para el sistema A del proceso de extracción líquido - líquido) se reordenan tomando en cuenta que ahora el dato conocido es $L_n x_{An}$ y no x_{An} :

$$\begin{aligned} f(y_{A1}) &= L_n x_{Bn} + V_1 y_{B1} - C_{B0} = 0 \\ y_{B1} &= f_{y_{A1}}(y_{A1}) \\ L_n x_{An} + V_1 y_{A1} &= C_{A0} \Rightarrow V_1 = \frac{C_{A0} - L_n x_{An}}{y_{A1}} \\ L_n &= M - V_1 \end{aligned}$$

$$x_{An} = \frac{L_n x_{An}}{L_n}$$

$$x_{Bn} = f_{xAxB}(x_{An})$$

Donde:

$$M = L_0 + V_0$$

$$C_{A0} = L_0 x_{A0} + V_0 y_{A0}$$

$$C_{B0} = L_0 x_{B0} + V_0 y_{B0}$$

Entonces el programa ELLN1 debe ser modificado de la siguiente manera:

```

« 1 9 START 0 NEXT
□ L0 XA0 XB0 LNXAN V0 YA0 YB0 FXAXB FYAYB
  M CA0 CB0 YA1 YB1 XAN XBN Ln V1
« L0 V0 + 'M' STO
  L0 XA0 * V0 YA0 * + 'CA0' STO
  L0 XB0 * V0 YB0 * + 'CB0' STO
« 'YA1' STO
  YA1 FYAYB EVAL 'YB1' STO
  CA0 LNXAN - YA1 / 'V1' STO
  M V1 - 'Ln' STO
  LNXAN Ln / 'XAN' STO
  XAN FXAXB EVAL 'XBN' STO
  Ln XBN * V1 YB1 * + CB0 -
» CA0 M / Newt DROP
Ln "LN" □TAG XAN "XAN" □TAG XBN "XBN" □TAG
V1 "V1" □TAG YA1 "YA1" □TAG YB1 "YB1" □TAG
»
»

```

Que se asumirá ha sido guardado en la variable global *ELLN2*. Los datos de equilibrio se convierten a fracciones en peso empleando el programa *LIX1* y tomando en cuenta que como solo existen dos columnas *N* debe ser igual a N_y , que N_x es cero y que x^*_a es igual a y^*_a .

```

[ 2.00 1.98 1.94 1.89 1.82 1.75 1.68 1.61 ]
[ 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 ]
[ 0 0 0 0 0 0 0 0 ]
[ 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 ]
LIX1

```

Entonces resolvemos el sistema *A* escribiendo lo siguiente en la pila:

```

2850
800 2850 /
2000 2850 /
120
1330
20 1330 /
0.0
« xA xB ROT INTLP »
« yA yB ROT INTLP »
ELLN2

```

Obteniendo los resultados:

```

LN: 3013.86138613
XAN: 3.98160315376E-2
XBN: .663600525624
V1: 1166.13861387

```

YA1: .600271692984

YB1: 0.

Que son los flujos y composiciones de las corrientes resultantes. Con estos datos y el programa ELLN podemos calcular entonces el número de etapas teóricas requeridas:

```
2850
800 2850 /
2000 2850 /
3.98160315376E-2
1166.13861387
.600271692984
0
« yA yB ROT INTLP »
« yA xA ROT INTLP »
« yA xB ROT INTLP »
ELLN
```

Resultando:

3.98458914798

Que es el número de etapas de equilibrio necesarias para lograr la separación requerida.

7.3.2. Ejercicios

1. Con las mismas condiciones del ejemplo 7.3.1 pero asumiendo un de N constante e igual a 1.85 kg de sólido/kg de solución, calcule los flujos y composiciones de salida así como el número de etapas teóricas requeridas.
2. Con las mismas condiciones del ejemplo 7.3.1, pero disminuyendo el flujo de solvente y soluto en un 10% a 1179 kg de benceno y 18 kg de aceite, calcule los flujos y composiciones de salida así como el número de etapas de equilibrio requeridas.
3. Un mineral conteniendo sólido inerte y sulfato de cobre será lixiviado en un equipo de múltiples etapas en contracorriente empleando agua pura. El flujo por hora de sólido consta de 10000 kg de sólido inerte, 1200 kg de sulfato de cobre y 400 kg de agua. La solución resultante deberá contener 92% en peso de agua y 8% de sulfato de cobre. Se deberá recuperar un 95% del sulfato de cobre contenido en el mineral. La retención de solución por el sólido inerte (N) es constante e igual a 0.5 kg de sólido inerte/kg de solución acuosa. Calcule los flujos y composiciones de salida así como el número de etapas teóricas requeridas.
4. Hígados de halibut frescos conteniendo 25.7% de aceite serán lixiviados con éter etílico puro para remover el 95% del aceite empleando un equipo de múltiples etapas en contracorriente. La alimentación consta de 1000 kg/h de hígados frescos. La solución final deberá contener un 70% en peso de aceite. La retención de solución en el sólido inerte (N) varía como se muestra en la siguiente tabla (obtenida experimentalmente).

N	y_A
4.88	0.00
3.50	0.20
2.47	0.40
1.67	0.60
1.39	0.81

Calcule los flujos y composiciones de salida así como el número de etapas teóricas necesarias.

5. Hojuelas de soya conteniendo 22% en peso de aceite serán lixiviadas con hexano puro en equipo de múltiples etapas en contracorriente hasta que su contenido sea de 0.25 kg de aceite por cada 100 kg de sólido inerte. Al equipo se alimentan 1000 kg/h de soya y 473 kg/h de hexano. La retención de solución en el sólido inerte (N) varía como se muestra en la siguiente tabla obtenida experimentalmente.

N	Y_A
1.73	0.00
1.60	0.12
1.55	0.17
1.52	0.20
1.45	0.28
1.43	0.30

Elabore un programa que calcule los flujos y composiciones de salida así como el número de etapas teóricas necesarias.

8. ADSORCIÓN

En la operación de adsorción uno o más componentes de una fase fluida (gaseosa o líquida) son adsorbidos en la superficie de un sólido, lográndose así la separación. Los compuestos adsorbidos pueden ser luego recuperados o desechados, dependiendo de su valor económico, mediante la operación contraria: *desorción*. La *desorción* es generalmente un proceso térmico donde se proporciona calor al adsorbente para que libere los compuestos adsorbidos.

La adsorción se emplea principalmente para recuperar o remover cantidades relativamente pequeñas de soluto de la fase fluida, así por ejemplo se emplea para remover compuestos orgánicos de soluciones acuosas u orgánicas, remover o recuperar colorantes y pigmentos orgánicos de soluciones acuosas u orgánicas, remover o recuperar compuestos orgánicos resultantes de procesos fermentativos, remover agua de hidrocarburos gaseosos, remover compuestos de azufre del gas natural, remover o recuperar solventes del aire, remover sustancias olorosas del aire, etc.

8.1. Propiedades físicas de los adsorbentes

La característica física más importante de los adsorbentes es su área superficial, la cual suele estar comprendida entre 100 y 2000 m² por gramo de adsorbente. Esta gran área superficial es posible gracias a los poros microscópicos con los que cuentan, los cuales pueden ocupar hasta un 50% del volumen del adsorbente. Los diámetros medios de estos poros están comprendidos entre 3 y 60 Å.

En cuanto a su forma física, normalmente el adsorbente tiene la forma de pelotitas o gránulos cuyos diámetros medios varían entre 0.1 y 12 mm. Las partículas más pequeñas se emplean en lechos fluidizados o procesos discontinuos, mientras que los tamaños más grandes se emplean en lechos rellenos.

Algunos de los adsorbentes más ampliamente empleados son los siguientes:

8.1.1. Carbón activado

El carbón activado es el producto de la descomposición térmica de la madera, cortezas vegetales, carbón mineral, etc. Su área superficial varía entre 600 y 800 m²/g, con diámetros medios de sus poros comprendidos entre 10 y 60 Å.

El carbón activado se emplea para adsorber compuestos orgánicos, es uno de los adsorbentes de mayor disponibilidad y puede ser fabricado fácilmente en nuestro medio a partir del carbón vegetal.

Para fabricar carbón activado a partir de carbón vegetal, se coloca el carbón vegetal en un recipiente, se le hecha agua recién hervida, se agita y se deja decantar, entonces se elimina el agua (que arrastra consigo las impurezas). Este proceso se repite unas cuatro veces, con lo cual el carbón queda activado.

Para incrementar aún más el poder adsorbente del carbón, es conveniente colocar el carbón resultante del anterior proceso en un lecho relleno, en un tubo de unas 2 pulgadas de diámetro y hacer pasar vapor de agua a través del mismo. Con este procedimiento se incrementa considerablemente el poder adsorbente del carbón activado.

Una vez que el carbón activado queda saturado, puede volver a ser utilizado sometiéndolo una vez más al procedimiento antes descrito.

8.1.2. Silica gel

La silica gel se fabrica tratando con ácido una solución de silicato de sodio que luego se deja secar. Se emplea principalmente como desecante, pues adsorbe agua tanto de sólidos como de líquidos.

Así por ejemplo es ampliamente utilizado para embalar productos que contienen partes de hierro o acero que pueden oxidarse, como sucede con la mayoría de los artefactos electrónicos. La silica gel impide la oxidación pues adsorbe la humedad del aire, con lo que no permite que el agua actúe sobre las partes de hierro.

Se emplea también en la fabricación de pañales desechables, es probablemente la parte más importante de los mismos pues es la encargada de adsorber la humedad, impidiendo así que los líquidos escurran.

La silica gel tiene áreas superficiales comprendidas entre 600 y 800 m²/g y el diámetro medio de sus poros varía entre 20 y 50 Å.

8.1.3. Alúmina activada

Este adsorbente se fabrica sometiendo a un proceso de secado el óxido de aluminio hidratado. Al igual que la silica gel se emplea principalmente como desecante pues adsorbe agua tanto de sólidos como de líquidos.

La alúmina activada tiene áreas superficiales comprendidas entre 200 y 500 m²/g y el diámetro medio de sus poros varía entre 20 y 140 Å.

8.1.4. Zeolitas

Las zeolitas son cristales naturales de silicato de aluminio. Los átomos de estos cristales se enlazan entre sí formando una red cristalina abierta. Es esta red abierta la que constituye los poros del adsorbente. Como los poros están conformados por los enlaces atómicos del compuesto, son todos del mismo tamaño, a diferencia de los adsorbentes antes descritos que tienen poros de tamaño variable.

Dependiendo del silicato de aluminio que constituye la zeolita, se tienen zeolitas con diámetros de poro que van desde los 3 Å hasta los 10 Å, pero un mismo tipo de zeolita siempre tiene el mismo diámetro de poro.

Las zeolitas se emplean como desecantes, en los procesos de intercambio iónico y debido a la uniformidad de sus poros son empleadas también como tamices moleculares para separar hidrocarburos, en esta última aplicación las moléculas con un diámetro menor o igual al de los poros pasan a través de la zeolita, siendo retenidas las moléculas con un tamaño mayor, lográndose así una separación efectiva.

Puesto que las zeolitas son adsorbentes naturales (no fabricados), es cuestión de ubicar yacimientos de las mismas para explotarlos. Con seguridad que en nuestro país debemos contar con este tipo de yacimientos y dada la amplia gama de aplicaciones que tienen se justifica una exploración de las mismas.

8.1.5. Resinas o polímeros sintéticos

Estos adsorbentes son polímeros fabricados a partir de dos tipos principales de monómeros: aromáticos y ésteres acrílicos. Los polímeros fabricados a partir de monómeros aromáticos como el estireno y el divinilbenceno son empleados principalmente para adsorber compuestos orgánicos no polares de soluciones acuosas, mientras que los polímeros fabricados a partir de éste-

res acrílicos son empleados para adsorber compuestos orgánicos polares de soluciones acuosas.

De los adsorbentes que hemos estudiado las resinas sintéticas son las menos accesibles en nuestro medio pues en nuestro país no se fabrica ningún tipo de polímero.

8.2. Relaciones de equilibrio

En el proceso de adsorción ocurren varias etapas: el soluto primero se difunde de la masa de fluido a la superficie exterior del adsorbente, luego se difunde al interior de los poros, en el interior de los poros se difunde a la superficie de los poros donde finalmente es adsorbido. A su vez la adsorción puede ocurrir en una sola capa o en varias capas, en cuyo caso cada una de ellas tiene un diferente poder de adsorción.

Debido a lo complejo de las interacciones que tienen lugar durante este proceso resulta prácticamente imposible predecir, con un margen de error aceptable, las relaciones de equilibrio. Por esta razón todas las relaciones de equilibrio que se emplean para dimensionar o diseñar equipos de adsorción provienen de datos experimentales.

En general se ha observado que los datos experimentales se ajustan normalmente a una de tres formas generales: la ecuación lineal, la ecuación de Freundlich y la ecuación de Langmuir (ver Figura 3.1)

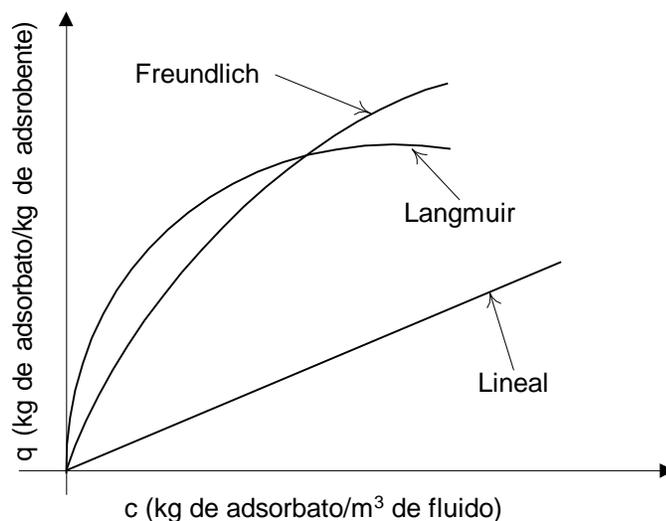


Figura 3.1. Formas comunes de la curva de equilibrio

Observe en la figura que al soluto se le ha dado el nombre de *adsorbato*. Esta es una denominación que se emplea frecuentemente en bibliografía, aunque nosotros emplearemos indistintamente ambos términos: *soluto* y *adsorbato* para hacer referencia al compuesto o compuestos a adsorber.

8.2.1. Forma lineal

La forma lineal se presenta muy rara vez en la práctica, sin embargo puede ser empleada en las regiones diluidas cuando el número de datos experimentales con los que se cuenta es reducido. La forma de la ecuación de equilibrio para este caso es:

$$q = Kc \tag{1}$$

Donde K es la constantes de equilibrio que debe ser determinada experimentalmente y que corresponde a la pendiente de la curva (pues como se puede ver en la Figura 3.1 la ordenada en el origen es cero).

Para calcular el valor de esta constante se efectúa una regresión lineal de los datos experimentales de equilibrio, siendo la pendiente de la ecuación ajustada el valor de K .

Para ajustar los datos en la calculadora HP se ingresa al menú *STAT*, se elige la opción *Fit data* (*ajuste de datos*), en el campo Σ *DAT* se introducen (con *EDIT*) los datos de la variable " c " (variable independiente) en la primera columna y los datos de la variable " q " (variable dependiente) en la segunda, se comprueba que el campo *X-Col* (variable independiente) esté en 1 y el campo *Y-Col* (variable dependiente) en 2 (sino es así deben ser colocados en 1 y 2 respectivamente). En el campo *Model* (modelo) se elige (con *CHOOS*) *Linear Fit* (*Ajuste Lineal*) y se presiona *OK*, entonces la calculadora realiza el ajuste y devuelve los resultados en la pila.

De los resultados del ajuste son de interés el coeficiente de correlación (*Correlation*) (que es devuelto en el segundo nivel de la pila) y la ecuación ajustada (que es devuelto en el tercer nivel de la pila).

El coeficiente de correlación nos permite juzgar cuan bien se ajustan los datos al modelo elegido. Un valor igual a 1 indica un ajuste perfecto, es decir que los valores calculados con la ecuación ajustada son los valores correctos. Por el contrario, un valor alejado de uno (por ejemplo 0.5) indica un mal ajuste, es decir que los resultados calculados con la ecuación ajustada no son correctos. Mientras más alejado sea el valor del coeficiente de correlación de uno peor es el ajuste y en consecuencia más alejados de los valores correctos los resultados calculados con la ecuación ajustada.

La ecuación ajustada (devuelta en el tercer nivel de la pila) tiene la forma $a+bx$, donde " a " es la ordenada en el origen (que en este caso deberá ser cero o cercano a cero) y " b " la pendiente de la línea recta el mismo que corresponde al valor de K .

8.2.2. Forma de Freundlich

La forma de *Freundlich* es probablemente la que más frecuentemente se presenta en la práctica. A esta forma se ajustan la mayoría de los sistemas de adsorción donde el fluido es líquido, también se ajustan a esta forma varios sistemas gaseosos, sobre todo cuando se adsorben hidrocarburos en carbón activado. La forma de esta ecuación es:

$$q = Kc^n \quad (2)$$

Donde K y n son constantes que deben ser determinadas experimentalmente. Para calcular estos valores se lleva a cabo una regresión de potencia de los datos experimentales de equilibrio.

Esta forma se considera favorable porque como se puede observar en la Figura 3.1, la adsorción es relativamente alta para concentraciones bajas de soluto en el fluido. Ello es importante en la adsorción pues esta operación se emplea principalmente para separar o recuperar soluto que se encuentra en concentraciones relativamente bajas.

Para ajustar los datos en la calculadora HP, se sigue el mismo procedimiento explicado en el ajuste lineal en el anterior acápite, excepto que ahora en lugar de elegir el modelo lineal (en el campo *model*), se elige *Power Fit* (*ajuste de potencia*).

En este caso la ecuación ajustada tiene la forma: bx^n , por lo que corresponde directamente a la ecuación (2), siendo b el valor de K . Al igual que en el ajuste lineal es importante analizar el coeficiente de correlación para determinar cuan bien se ajustan los datos experimentales a la ecuación de *Freundlich*.

8.2.3. Forma de Langmuir

Esta forma tiene una base teórica, ha sido deducida asumiendo que el adsorbente tiene un número fijo de puntos de adsorción y que la adsorción ocurre en una sola capa. Sin embargo son pocos los sistemas de adsorción que obedecen a esta ecuación por lo que su utilidad práctica es limitada. La forma de *Langmuir* es:

$$q = \frac{q_0 c}{K + c} \quad (3)$$

Donde q_0 y K son constantes que deben ser determinadas experimentalmente mediante una regresión de los datos de equilibrio.

La ecuación de *Langmuir* es considerada altamente favorable porque como se puede observar en la Figura 3.1, la adsorción es elevada aún para concentraciones muy bajas de soluto. Es por consiguiente la forma ideal para cualquier proceso de adsorción, sin embargo y como se dijo antes son muy pocos los sistemas que obedecen a esta ecuación.

Para ajustar los datos experimentales la ecuación (3), la misma debe ser escrita en la forma:

$$\frac{1}{q} = \frac{K + c}{q_0 c} = \frac{K}{q_0} \left(\frac{1}{c} \right) + \frac{c}{q_0 c} = \frac{1}{q_0} + \frac{K}{q_0} \left(\frac{1}{c} \right) \quad (3.a)$$

Que es la ecuación de una línea recta ($y=a+bx$), donde la variable independiente (x) es $1/c$, la variable dependiente (y) $1/q$, la ordenada en el origen (a) $1/q_0$ y la pendiente (b) K/q_0 .

Entonces para ajustar los datos en la calculadora *HP*, se procede igual que en el ajuste lineal explicado en el acápite 8.2.1, excepto que ahora se deben introducir en la primera columna la inversa de c ($1/c$) y en la segunda la inversa de q ($1/q$).

Una vez obtenida la ecuación ajustada (que como vimos tiene la forma $a+bx$) se calcula el valor de q_0 , sacando la inversa de de la ordenada ($1/a$) y el valor de K se calcula multiplicando el valor de la pendiente por q_0 ($b \cdot q_0$).

Como en las anteriores formas, es importante analizar el coeficiente de correlación para determinar cuan bien se ajustan los datos a la ecuación de *Langmuir*.

Casi siempre, sin importar cual se la forma que siga la curva de equilibrio, la adsorción disminuye rápidamente a medida que aumenta la temperatura. Este hecho es aprovechado para regenerar el adsorbente sometándolo a un tratamiento térmico (dando lugar así a la desorción).

8.2.4. Ejemplo

Los siguientes datos de equilibrio fueron obtenidos experimentalmente, utilizando soluciones de fenol en agua empleando como adsorbente carbón activado en forma de granos. Las pruebas fueron realizadas a temperatura ambiente y la misma se mantuvo aproximadamente constante a 20°C. Determine a que forma se ajustan mejor los datos.

c (kg de fenol/m ³ de solución)	q (kg de fenol/kg de carbón)
0.322	0.150
0.117	0.122
0.039	0.094
0.0061	0.059
0.0011	0.045

Solución

Primero realizamos el ajuste lineal siguiendo el procedimiento descrito en el acápite 8.2.1. Los datos que se introducen en el campo Σ DAT, en las columnas 1 y 2 son los de la tabla.

Los resultados del ajuste son los siguientes:

```
' .065582195297+.292846297434*X'
```

```
Correlation: .903629766008
```

```
Covariance: .00526015
```

Que tiene un coeficiente de correlación aceptable, con un error de +/- 10%.

Con los mismos datos probamos ahora la ecuación de Freundlich, por lo tanto hacemos un ajuste de potencia (es decir elegimos el modelo de potencia: *Power Fit*) con lo que obtenemos los siguientes resultados:

```
' .191276587893*X^.21841471793'
```

```
Correlation: .997026271678
```

```
Covariance: 1.1388904035
```

Que tiene un coeficiente de correlación muy cercano a uno, por lo que podemos concluir que es un muy buen ajuste, mucho mejor por su puesto que el ajuste lineal.

Finalmente hacemos la prueba con la ecuación de *Langmuir*. Para ajustar a la ecuación de *Langmuir* tenemos que introducir la inversa de los datos empleados en los dos ajustes anteriores, como estos datos están almacenados en la variable global Σ DAT, podemos calcular la inversa de los mismos evitando así el volver a introducirlos.

Para ello en (la HP49) colocamos en la pila la variable Σ DAT, convertimos el array en una lista con *AXL*, calculamos la inversa de las listas con « *INV* » *DOLIST*, convertimos el resultado nuevamente en array con *AXL* y guardamos el resultado en Σ DAT. O podemos escribir directamente lo siguiente:

```
 $\Sigma$ DAT AXL « INV » DOLIST AXL ' $\Sigma$ DAT' STO
```

Entonces realizamos el ajuste lineal (*Linera Fit*) con lo que obtenemos los siguientes resultados:

```
'9.65114824893+1.47861290027E-2*X'
```

```
Correlation: .885505230829
```

```
Covariance: 2245.64074341
```

Por consiguiente las constantes de la ecuación de *Langmuir* (q_0 y K) son:

$$q_0 = \frac{1}{a} = \frac{1}{9.65114824893} = .103614613951$$

$$K = b \cdot q_0 = 1.47861290027E-2 \cdot .103614613951 = 1.53205904844E-3$$

Sin embargo el coeficiente de correlación de este ajuste es el peor de todos (el más alejado de 1). Por consiguiente la ecuación a la que mejor se ajustan los datos de equilibrio es la de *Freundlich*:

$$q = Kc^n = 0.191276587893c^{0.21841471793} \cong 0.1913c^{0.2184}$$

8.2.5. Preguntas y ejercicios

1. ¿Qué ocurre en la operación de adsorción?
2. ¿Cuál es la operación contraria a la adsorción?
3. ¿En qué situaciones es conveniente aplicar la operación de adsorción?
4. ¿Cuál es la característica física más importante de los adsorbentes?
5. ¿De donde resulta el carbón activado?
6. ¿Para qué se emplea el carbón activado?
7. ¿Cuál es la principal aplicación de la sílica gel?
8. ¿Cuál es la principal aplicación de la alúmina activa?
9. ¿Cómo se fabrican las zeolitas?
10. ¿Cuáles son las principales aplicaciones de las zeolitas?
11. ¿En la adsorción es posible predecir las curvas de equilibrio con un margen de error aceptable? ¿Por qué?
12. ¿Cuáles son las tres formas a las que normalmente se ajustan los datos de equilibrio?
13. De las tres formas a las que normalmente se ajustan los datos de equilibrio ¿cuál es la que más frecuentemente se presenta en la práctica?
14. Cuando se ajustan los datos a una ecuación, ¿Para qué sirve el coeficiente de correlación?
15. Entre la ecuación de *Freundlich* y la de *Langmuir*, ¿Cuál tiene la forma más favorable y por qué?
16. Los siguientes datos isotérmicos de equilibrio corresponden a la adsorción de glucosa de soluciones acuosas empleando como adsorbente alúmina activada. Determine la ecuación que mejor se ajusta a los datos y escriba las constantes de la misma tomando en cuenta las unidades.

c (g /cm ³)	0.0040	0.0087	0.019	0.027	0.094	0.195
q (g soluto/g alúmina)	0.026	0.053	0.075	0.082	0.123	0.129

8.3. Adsorción por lotes

La adsorción por lotes se emplea principalmente a escala de planta piloto y en la industria farmacéutica para adsorber soluto de soluciones líquidas. En este proceso el adsorbente y la solución son puestos en contacto íntimo con la ayuda de un agitador.

Una vez alcanzado el equilibrio se deja decantar la mezcla y se separa la solución del adsorbente con la ayuda de un filtro (pues el adsorbente se encuentra normalmente en forma de pequeñas partículas).

Si llamamos M a la masa de adsorbente (en kg), S el volumen de solución (en m^3), q la concentración de soluto en el adsorbente (en kg de soluto/kg de adsorbente) y c la concentración de soluto en la solución (en kg/m^3), entonces el balance de materia para el soluto resulta:

$$q_F M + c_F S = qM + cS \tag{4}$$

Donde los subíndices F denotan las concentraciones en la alimentación (*Feed*). Debido a que generalmente las soluciones son diluidas, se asume generalmente que la masa de adsorbente (M) es el peso total de adsorbente (incluido el soluto) y que el volumen de la solución (S) es constante. Esa es la razón por la cual estos parámetros no tienen subíndices en la ecuación (4).

Los datos que usualmente se conoce son: la masa de adsorbente (M), el volumen de solución (S), la concentración inicial de soluto en el adsorbente (q_F) y la concentración inicial de soluto en la solución (c_F). Son incógnitas la concentración final de soluto en el adsorbente (q) y la concentración final de soluto en la solución (c).

Puesto que se tiene dos incógnitas y una ecuación, hace falta una ecuación más para resolver el sistema. Esta ecuación es la ecuación de equilibrio que toma una de las tres formas estudiadas en el acápite 8.2. Entonces con la ecuación (4) y la ecuación de equilibrio se puede resolver el sistema calculando así las dos incógnitas (q y c).

Si la ecuación de equilibrio toma la forma de la ecuación de *Freundlich* o la de *Langmuir*, entonces se tiene un sistema de dos ecuaciones, una de ellas no lineal, por lo que la solución no puede ser calculada analíticamente. En estos casos (que son los más frecuentes) se puede encontrar la solución gráfica o numéricamente

Para encontrar la solución gráficamente se dibuja la curva de equilibrio en una gráfica c versus q y se traza la línea del balance de materia de manera que atraviese la curva de equilibrio (ver Figura 3.2). La intersección entre la curva de equilibrio y la línea del balance de materia constituye la solución buscada.

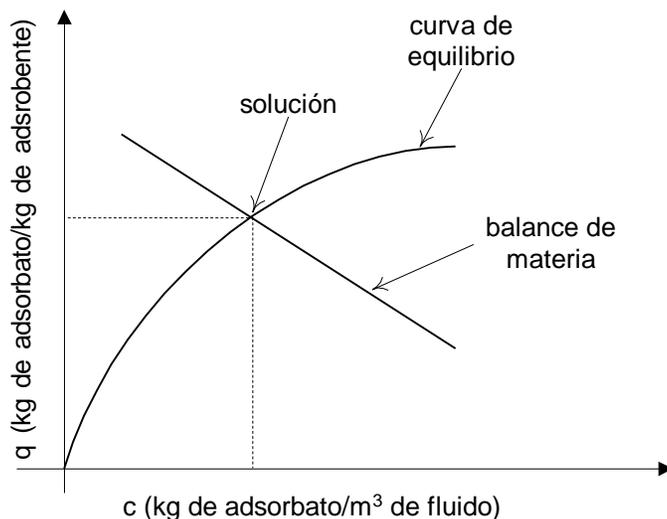


Figura 3.2. Adsorción por lotes. Solución gráfica

Numéricamente se trata el problema como una función de una sola variable y se la resuelve con un método iterativo (como el de Newton - Raphson).

En nuestro caso la función a resolver será la siguiente:

$$f(c) = qM + cS - (q_F M - c_F S) = qM + cS - c_0 = 0$$

$$q = f_c(c)$$
(5)

Donde f_c es la función de equilibrio (*Lineal, Freundlich* o *Langmuir*), que nos permite calcular un valor de q conociendo el valor de c . Resolveremos la ecuación (5) con el método de Newton - Raphson (*NEWT*), trabajando con 9 dígitos de precisión y tomando como valor inicial la mitad del valor de c_F . El programa que resuelve la ecuación (5) es el siguiente:

```

« 0 0 0 → cf qf S M fc c0 c q
« qf M * cf S * + 'c0' STO
« 'c' STO
  c fc EVAL 'q' STO
  q M * c S * + c0 -
»
cf 2 / 9 30 NEWT DROP
c "c" →TAG q "q" →TAG
»
»

```

En lo futuro se asumirá que este programa ha sido guardado en al variable 'Adb1'.

Los datos que requiere este programa y en ese orden son: la concentración inicial de soluto en la solución (cf), la concentración inicial de soluto en el adsorbente (qf), el volumen de solución a tratar (S), la masa de adsorbente empleada (M) y la función de equilibrio (fc). Devuelve como resultados la concentración final de soluto en la solución (c) y la concentración final de soluto en el adsorbente (q).

8.3.1. Ejemplo

1 m³ de agua residual conteniendo 0.21 kg de fenol, será tratada con 1.40 kg de carbón activado granular fresco. La mezcla será agitada hasta que alcance el equilibrio. Empleando los datos de equilibrio del ejemplo 8.2.4, calcule las composiciones finales así como el porcentaje de fenol extraído.

Solución

Los datos con los que contamos son los siguientes: concentración inicial de soluto en la solución: $c_F = 0.21 \text{ kg/m}^3$; concentración inicial de soluto en el adsorbente: $q_F = 0 \text{ kg/kg}$; volumen de solución: $S = 1 \text{ m}^3$; masa de adsorbente: $M = 1.4 \text{ kg}$ y la ecuación de equilibrio del ejemplo 8.2.4: $q = f_c = 0.191276587893c^{0.21841471793}$.

Contamos entonces con todos los datos que se requieren para resolver el problema con el programa *Abd1*. En consecuencia escribimos lo siguiente en la pila:

```
0.21 0 1 1.4 « 0.21841471793 ^ 0.191276587893 * » Adb1
```

Con lo que obtenemos los resultados:

c: 6.33967378826E-2

q: .104716615798

Que son las concentraciones finales de soluto en la solución y en el adsorbente respectivamente.

Entonces el porcentaje de fenol extraído es:

$$\% \text{ de fenol extraído} = \frac{0.21 - 6.33967378826E-2}{0.21} \times 100 = 69.8110771986$$

8.3.2. Preguntas y ejercicios

1. ¿Cuáles son los principales campos de aplicación de la adsorción por lotes?
2. ¿Qué tipo de fluido es adecuado para la adsorción por lotes?
3. ¿En el balance de materia qué se asume con relación a la masa de adsorbente y el volumen de solución?
4. Si el volumen de solución (S) no fuera considerado constante ¿Qué otro dato o datos serían necesarios en el balance de materia?
5. ¿Por qué en la mayoría de los casos no es posible resolver analíticamente el sistema de ecuaciones que se forma en la adsorción por lotes?
6. En la solución numérica propuesta se ha colocado el sistema como una función de c ¿Es posible colocar el sistema como una función de q ?
7. ¿Numéricamente es posible resolver la función con el método de Regula - Falsi? De ser así ¿Qué parte del programa tendría que ser modificado?
8. Si en lugar de emplear como valor inicial igual a la mitad de c_F , se emplea el promedio de c_F y q_F , ¿Qué cambio debería hacerse en el programa?
9. En el programa ¿Para qué se utiliza el comando *DROP* después de llamar al método de Newton?
10. Calcule las concentraciones finales y el porcentaje de fenol extraído si en el ejemplo el contenido de fenol es de 0.5 kg por m^3 y se emplea 1 kg de carbón activado.
11. Calcule las concentraciones finales y el porcentaje de fenol extraído si en el ejemplo el volumen de solución es de 2.5 m^3 , en los cuales existe 0.625 kg de fenol, siendo la masa de carbón activado empleada 3.0 kg.
12. 3 m^3 de una solución acuosa de glucosa contiendo 0.145 kg de glucosa/ m^3 , será mezclada con 4.2 kg de alúmina activada hasta alcanzar el equilibrio. Calcule las concentraciones finales y el porcentaje de glucosa extraído. Emplee los datos de equilibrio del ejercicio 16 del acápite 8.2.5.

8.4. Diseño de columnas de adsorción de lecho fijo

Las columnas de adsorción de lecho fijo constituyen el método de adsorción más ampliamente empleado a nivel industrial. Puesto que estas columnas son normalmente pequeñas y fáciles de montar, son también de uso frecuente a nivel de planta piloto.

Una columna de adsorción de lecho fijo básicamente es un tubo con una malla o filtro en la parte inferior para impedir la salida del adsorbente (ver Figura 3.3). El adsorbente, que constituye el relleno de la columna, se encuentra generalmente en forma de gránulos con diámetros medios comprendidos entre 5 y 12 mm. El fluido a ser tratado se hace pasar a través de la colum-

na a un flujo constante. En el caso de los líquidos fluye casi siempre hacia abajo, mientras que en el caso de los gases el flujo puede ser indistintamente hacia abajo o hacia arriba de la columna.

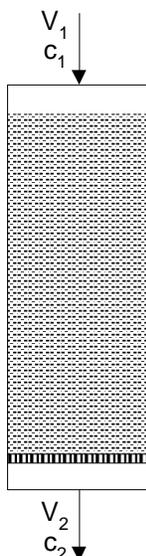


Figura 3.3. Esquema de una columna de adsorción de lecho fijo

Aun cuando las columnas de adsorción de lecho fijo tienen la apariencia de un equipo continuo estacionario, son en realidad equipos discontinuos donde el flujo y concentración de entrada son estacionarios, pero los de salida no. Estos equipos se hacen funcionar mientras el flujo y concentración de salida permanecen esencialmente constantes y se detiene su funcionamiento tan pronto como la concentración comienza a ascender.

Se ha comprobado experimentalmente que la saturación en estos equipos ocurre uniforme y progresivamente. Primero se satura el adsorbente que se encuentra al principio de la columna, mientras que el adsorbente restante permanece prácticamente inalterado. A medida que transcurre el tiempo se va saturando un altura mayor de adsorbente, pero el adsorbente restante sigue inalterado, finalmente al cabo de cierto tiempo se satura la mayor parte del adsorbente y entonces parte del soluto no es adsorbido, como consecuencia la concentración de salida comienza a ascender. Es en este momento, conocido como el tiempo de ruptura, donde se detiene el funcionamiento del equipo y se pasa el flujo a otro equipo similar pero con adsorbente fresco.

El perfil de concentraciones en una torre de adsorción, en función de la altura de relleno (medido deteniendo el proceso de adsorción para diferentes tiempos: t_1 a t_6) tiene la apariencia que se muestra en la Figura 3.4.

Como se puede observar en esta figura, el perfil de concentraciones se desplaza a lo largo de la columna pero permanece constante, es decir en todo momento, el proceso de adsorción ocurre sólo en una altura fija del relleno: H_{UNB} . Las alturas de relleno anterior y posterior a H_{UNB} permanecen inalteradas. Esta altura de adsorción está ubicada inicialmente al principio de la torre y se va desplazando hacia abajo hasta que en un tiempo dado (t_5) alcanza el final de la torre. Para cualquier tiempo posterior a este (t_6), el perfil sigue siendo constante, pero como no existe suficiente altura de relleno no se completa y como consecuencia la concentración final de soluto comienza a ascender rápidamente.

En el diseño de las torres de adsorción de lecho fijo se aprovecha esta altura constante (H_{UNB}). Con ese fin se llevan a cabo pruebas experimentales

a nivel de laboratorio que permiten determinar esta longitud, entonces los resultados pueden ser extrapolados al equipo industrial sumando a la altura requerida la altura (H_{UNB}) determinada experimentalmente. Esta altura extra de relleno (conocida como altura no utilizable de la torre) asegura que el equipo sea utilizado solamente mientras se adsorbe la totalidad (o casi la totalidad) del soluto.

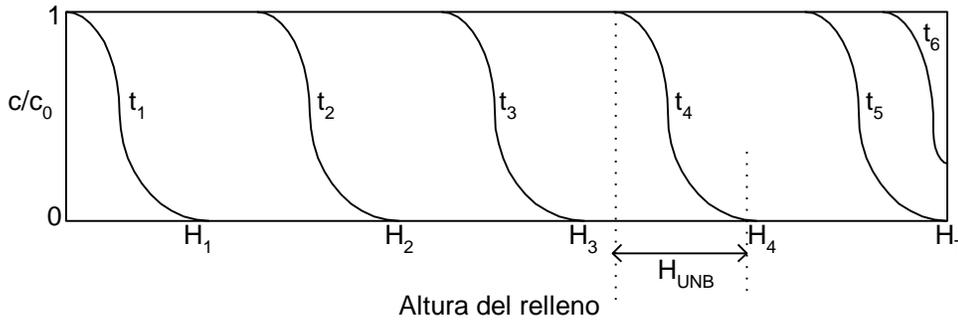


Figura 3.4. Perfil de concentraciones en función de la altura del lecho relleno

Para determinar experimentalmente la altura no utilizable H_{UNB} , se lleva a cabo una corrida experimental en un equipo de adsorción pequeño, usualmente un tubo de 1 o 2 pulgadas de diámetro y entre 10 y 50 cm de longitud. Al efectuar esta prueba se debe tener cuidado de emplear condiciones de temperatura, presión, relación diámetro de partícula a diámetro de la columna, concentración y flujo másico similares a los del equipo industrial.

En esta prueba se miden las concentraciones de soluto a la salida del equipo, inicialmente a intervalos regulares de 1 o 2 horas (mientras la concentración permanece constante) y una vez que comienza a variar con mayor frecuencia.

Con los datos de la corrida experimental se construye una gráfica de la relación de concentración c/c_0 versus el tiempo, la cual tiene una forma similar al de la Figura 3.5. (donde c_0 es la concentración inicial de soluto).

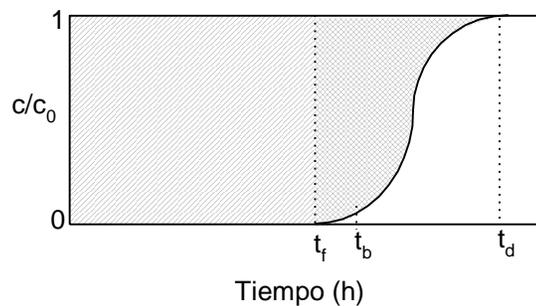


Figura 3.5. Resultados de la prueba experimental que permite calcular la altura no utilizada de relleno.

Como se puede observar, el perfil final de concentraciones (desde t_f hasta t_d) es la imagen del perfil de concentraciones mostrada en la Figura 3.4. En esta gráfica t_f es el tiempo a partir del cual comienza a escapar soluto y t_d es el tiempo para el cual el adsorbente queda totalmente saturado.

El área total sobre la curva (entre 0 y t_d) es equivalente a la altura total de la torre, mientras que la comprendida entre 0 y t_f es equivalente a

la altura útil de la torre. Por lo tanto el área comprendida entre t_f y t_d es el equivalente a la altura no utilizada de la torre H_{UNB} .

Como en la práctica es difícil determinar con exactitud el valor de t_f , se suele utilizar en su lugar t_b , que es el tiempo para el cual la relación c/c_0 está comprendida entre 0.01 y 0.05. Este tiempo se conoce como el tiempo del punto de ruptura.

El tiempo total (t_t), equivalente a la altura total de la torre, puede ser calculado invirtiendo la Figura 3.5 y calculando el área bajo la curva, es decir resolviendo la integral:

$$t_t = \int_0^{t_d} \left(1 - \frac{c}{c_0}\right) dt \quad (6)$$

De manera similar, el tiempo equivalente a la altura útil de relleno (t_u) puede ser calculada con:

$$t_u = \int_0^{t_b} \left(1 - \frac{c}{c_0}\right) dt \quad (7)$$

Donde como se puede ver se emplea el tiempo del punto de ruptura t_b en lugar de t_f .

Entonces se cumple aproximadamente que:

$$\frac{H_U}{H_T} = \frac{t_u}{t_t} \quad (8)$$

Donde H_T es la altura total de la torre la cual se conoce pues es la altura de relleno empleada en la prueba experimental. Entonces la única incógnita H_U (la altura útil de relleno) puede ser despejada de esta ecuación:

$$H_U = H_T \frac{t_u}{t_t} \quad (9)$$

Finalmente, la altura no utilizable del equipo (H_{UNB}) que es el valor buscado, puede ser calculada restando a la altura total la altura útil:

$$H_{UNB} = H_T - H_U = H_T - H_T \frac{t_u}{t_t} = H_T \left(1 - \frac{t_u}{t_t}\right) \quad (10)$$

Las integrales pueden ser calculadas directamente a partir de los datos experimentales empleando el método de Simpson en su forma tabular (programa *SIMPT*). No obstante, dado que en este caso la mayor parte del área se encuentra debajo de una línea recta y el método de Simpson emplea una parábola, se obtienen resultados un tanto más exactos empleando el método del trapecio (que utiliza una línea recta). No obstante la integral correspondiente al sector comprendido entre t_f y t_d (que es una curva) se calcula con mayor exactitud empleando el método de Simpson. En general no obstante se calcula el área (la integral) empleando sólo uno de los dos métodos. En los ejemplos de este acápite emplearemos el método del Trapecio.

Al igual que el método de Simpson, el método del trapecio se implementa en dos formas: a) Para integrar funciones analíticas y b) para integrar datos tabulados.

Aun cuando casi siempre el método de Simpson calcula resultados más exactos cuando se integran funciones analíticas, presentamos el programa para el método del Trapecio, para aquellas situaciones donde este método represente alguna ventaja:

```

« 0 0 → F A B N H S
« B A - N / 'H' STO
  A H + B H - FOR X
    X F EVAL 'S' STO+
  H STEP
  A F EVAL 2 S * + B F EVAL + H * 2 /
»
»

```

En lo sucesivo se asumirá que este programa ha sido guardado en la variable *TRAPP*. Al igual que el método de Simpson, este programa requiere como datos la función a integrar (función de una variable), los límites inferior y superior de integración y el número de segmentos.

Por ejemplo para integrar la siguiente ecuación analítica:

$$\int_{5.1}^{8.4} (3 - 6x^2 + 1.1x^{3.1}) dx$$

Se escribe en la pila:

```
« → X « 3 6 X SQ - 1.1 X 3.1 ^ * + » » 5.1 8.4 20 TRAPP
```

Con lo que se obtiene el resultado: 529.06770079, que es el valor de la integral y que es menos exacto que el valor calculado con el método de Simpson.

El programa que integra una serie de datos tabulados es:

```

« 0 0 → X Y N S
« X SIZE 1 GET 1 - 'N' STO
  1 N FOR I
    X I 1 + GET
    X I GET -
    Y I 1 + GET
    Y I GET +
    2 / * 'S' STO+
  NEXT
  S
»
»

```

En lo sucesivo se asumirá que este programa ha sido guardado con el nombre "*TRAPT*".

Así por ejemplo para calcular el valor de la siguiente integral:

$$\int_{2.5}^{7.2} f(x) dx = \int_{2.5}^{7.2} y dx$$

En base a los siguientes datos tabulados:

x	y
2.0	4.800
2.5	6.625
2.7	7.509
3.0	9.000
3.3	10.689
3.8	13.944
4.2	16.944
4.5	19.425

5.1	24.981
5.7	31.329
6.0	34.800
6.4	39.736
6.9	46.401
7.2	50.664
7.5	55.125

Guardamos los datos tabulados, entre los límites de integración, en dos variables:

```
[2.5 2.7 3.0 3.3 3.8 4.2 4.5 5.1 5.7 6.0 6.4 6.9 7.2] 'Vx' STO
[6.625 7.509 9.0 10.689 13.944 16.944 19.425 24.981 31.329 34.8 39.736
46.401 50.664] 'Vy' STO
```

Y escribimos lo siguiente en la pila:

```
Vx Vy TRAPT
```

Con lo que obtenemos el resultado: 115.76965, que es el valor de la integral.

Puesto que utilizaremos con relativa frecuencia el comando *AXL* existente en la calculadora HP49, presentamos el siguiente programa que hace lo mismo en la calculador HP48:

```
<< DUP TYPE
  IF 5 ==
  THEN LIST→ OVER TYPE
    IF 5 ==
    THEN → N
      << 1 N
        START LIST→ →ARRY N ROLLD
        NEXT N ROW→
      >>
    ELSE →ARRY
    END
  ELSE ARRY→ LIST→
    IF 2 ==
    THEN DUP2 * → N M NE
      << 1 N
        FOR I M →LIST 'NE' M STO- NE I + ROLLD
        NEXT N →LIST
      >>
    ELSE →LIST
    END
  END
END
>>
```

En lo sucesivo asumiremos que este programa ha sido guardado con el nombre "AXL".

8.4.1. Ejemplo

Una corriente residual de aire contiene alcohol, el mismo que será adsorbido en una torre de lecho fijo con carbón activado. Para determinar la longitud del equipo industrial se ha realizado una corrida experimental empleando un lecho relleno con 79.2 g de carbón activado en un tubo de 4 cm de diámetro y 14 cm de altura. El gas tiene una concentración inicial de 600 ppm, una densidad de 0.00115 g/cm³ y un flujo de 754 cm³/s. Los resultados del experimento son los siguientes:

$t (h)$	c/c_0
0.0	0.000
3.0	0.000
3.5	0.002
4.0	0.030
4.5	0.155
5.0	0.396
5.5	0.658
6.0	0.903
6.2	0.933
6.5	0.975
6.8	0.993

Calcule:

- a) La longitud no utilizada del lecho (H_{UNB}) considerando que el punto de ruptura ocurre en $c/c_0 = 0.01$.
- b) La longitud del equipo industrial si tiene que funcionar durante 6 horas continuas.
- c) La capacidad de saturación del carbón activado.

Solución:

Inciso a:

Puesto que el último dato sólo llega hasta 0.993, realizamos una extrapolación lineal para calcular el valor de t_d :

[0.975 0.993] [6.5 6.8] 1.0 INTLP

Que nos devuelve el valor de t_d : 6.9166666667 \approx 6.92 h.

De manera similar determinamos el valor del tiempo de ruptura (t_b):

[0.002 0.030] [3.5 4.0] 0.01 INTLP

Que nos devuelve el valor t_b : 3.64285714286 \approx 3.64 h.

Ahora calculamos el tiempo total (t_t):

[0.0 3.0 3.5 4.0 4.5 5.0 5.5 6.0 6.2 6.5 6.8 6.9] [0.000 0.000 0.002 0.030 0.155 0.396 0.658 0.903 0.933 0.975 0.993 1.0] AXL 1 SWAP - AXL TRAPT

Que nos devuelve el valor de t_t : 5.1891. (El método de Simpson devuelve: 5.20465952381)

Entonces calculamos el valor de t_u :

[0.0 3.0 3.5 3.64] [0.000 0.000 0.002 0.01] AXL 1 SWAP - AXL TRAPT

Que nos devuelve el valor de t_u : 3.63866 \approx 3.6387 h. (El método de Simpson devuelve 3.64557074285).

Entonces la altura de relleno utilizada (H_U) puede ser calculada con la ecuación (9):

$$H_U = H_T \frac{t_u}{t_i} = 14.0 \frac{3.6387}{5.1891} = 9.81707810602 \approx 9.82 \text{ cm.}$$

Y la altura no utilizable (H_{UNB}) con la ecuación (10):

$$H_{UNB} = H_T \left(1 - \frac{t_u}{t_i}\right) = 14.0 \left(1 - \frac{3.6387}{5.1891}\right) = 4.18292189398 \approx 4.18 \text{ cm.}$$

Por supuesto este valor podía ser calculado también restando a la longitud total H_T la longitud útil H_U ($14.0 - 9.82 = 4.18$).

Inciso b:

Puesto que en las torres de adsorción el perfil de concentraciones es constante y sólo se desplaza hacia el final de la torre a medida que transcurre el tiempo, la altura requerida para que el equipo funcione continuamente durante un cierto intervalo de tiempo se calcula con una regla de tres simple. Por supuesto es necesario conocer algún valor de la altura de relleno y el tiempo de ruptura para esa altura. Estos valores resultan de la prueba experimental y son los que se han determinado en el anterior inciso: $H_U = 9.82 \text{ cm}$ y $t_b = 3.64 \text{ h}$.

Por consiguiente la altura de relleno que deberá tener el equipo industrial es:

$$6 \text{ horas de funcionamiento continuo} * \frac{9.82 \text{ cm}}{3.64 \text{ horas de funcionamiento continuo}} = 16.1868131868 \text{ cm} \approx 16.17 \text{ cm.}$$

A esta altura se le debe sumar la longitud no utilizada H_{UNB} (que es constante) pues de lo contrario al final del intervalo de 6 horas el equipo ya estaría saturado y no adsorbería nada de soluto. Entonces la altura del relleno en el equipo industrial deberá ser:

$$\text{Altura del relleno en el equipo industrial} = 16.17 + 4.18 = 20.35 \text{ cm.}$$

Como se puede observar el equipo "industrial" puede ser considerado por su tamaño un equipo de laboratorio o de planta piloto.

Inciso c

La capacidad de saturación del carbón activado es la cantidad de soluto que puede adsorber el carbón activado hasta quedar totalmente saturado. Este valor se calcula con el tiempo total t_t , que como se dijo es el tiempo equivalente a la altura total del equipo, es decir el tiempo equivalente a la utilización total del relleno existente en el equipo.

Para calcular la cantidad de soluto que se adsorbe hasta saturar el relleno se multiplica el flujo de gas por su concentración y por el tiempo total (t_t).

El flujo del gas en este ejemplo es:

$$\left(\frac{754 \text{ cm}^3 \text{ de gas}}{s}\right) * \left(\frac{0.0015 \text{ g de gas}}{1 \text{ cm}^3 \text{ de gas}}\right) * \left(\frac{3600 \text{ s}}{1 \text{ h}}\right) = \left(\frac{3121.56 \text{ g de gas}}{h}\right)$$

La concentración inicial del gas es $600 \text{ ppm} = 600 \text{ g de soluto} / (1 \text{ millón de gramos de gas}) = 0.0006 \text{ g de soluto} / \text{g de gas}$.

El tiempo total, calculado en el inciso (a) es: 5.1891 h .

Por lo tanto la cantidad de soluto que se adsorbe hasta saturar el relleno es:

$$\left(\frac{3121.56 \text{ g de gas}}{h}\right) * \left(\frac{0.0006 \text{ g de soluto}}{\text{g de gas}}\right) * 5.1891 \text{ h} = 9.7188521976 \approx 9.72 \text{ g de soluto}$$

Entonces la capacidad de saturación del carbón activado es:

$$\text{Capacidad de saturación del carbón activado} = \left(\frac{9.72 \text{ g de soluto}}{79.2 \text{ g de carbón activado}}\right) \approx \frac{0.123 \text{ g de soluto}}{\text{g de carbón activado}}$$

Al extrapolar los datos de laboratorio a escala industrial se debe mantener constante el flujo superficial del gas (distancia/tiempo), entonces con frecuencia no sólo se debe calcular la altura de la torre, sino también el diámetro de manera que se cumpla esta condición. Así si en el anterior ejemplo el flujo de gas que se emplea en la industria es 3000 cm³/s (en lugar de los 754 cm³/s del experimento) el diámetro del equipo industrial debe ser calculado como se explica a continuación.

El área transversal del equipo experimental es: $\pi * 4^2 / 4 \approx 12.57 \text{ cm}^2$, entonces la velocidad superficial del equipo es:

$$\text{Velocidad superficial del equipo experimental} = \left(\frac{754 \frac{\text{cm}^3}{\text{s}}}{12.57 \text{ cm}^2}\right) \approx 59.98 \frac{\text{cm}}{\text{s}}$$

Como esta velocidad debe permanecer constante, el área transversal del equipo industrial será:

$$\text{Área transversal del equipo industrial} = \left(\frac{3000 \frac{\text{cm}^3}{\text{s}}}{59.98 \frac{\text{cm}}{\text{s}}}\right) \approx 50.02 \text{ cm}^2$$

Por lo tanto el diámetro del equipo industrial será:

$$\text{Diámetro del equipo industrial} = \sqrt{\frac{4 * 50.02}{\pi}} \approx 7.98 \text{ cm}$$

8.4.2. Preguntas y ejercicios

1. ¿Básicamente de qué consta una torre de adsorción de lecho fijo?
2. ¿Las torres de adsorción de lecho fijo son equipos de funcionamiento continuo?
3. ¿Cómo cambia el perfil de concentraciones a lo largo de la columna de adsorción?
4. ¿Por qué al equipo industrial se le suma la altura H_{UNB} ?
5. ¿Cómo se determina el valor de H_{UNB} ?
6. ¿Qué parámetro se mide en la prueba experimental que se realiza en el diseño de torres de adsorción?
7. ¿A qué es equivalente el área total sobre la curva de adsorción?
8. ¿Qué es el tiempo de ruptura y por qué se utiliza en el diseño?
9. ¿A qué es equivalente el área sobre la curva de adsorción comprendida entre 0 y t_b ?
10. ¿Por qué el área comprendida entre 0 y t_b se calcula con mayor exactitud empleando el método del Trapecio?

11. ¿Por qué la altura útil del equipo industrial se calcula con una regla de tres simple?
12. ¿Qué es la capacidad de saturación de un adsorbente?
13. Al extrapolar los datos de laboratorio a nivel industrial ¿Qué parámetro debe permanecer constante?
14. En el diseño de una torre de adsorción, por extrapolación de los datos experimentales ¿Sólo es necesario calcular la altura de relleno?
15. Empleando los datos y resultados del ejemplo calcule lo siguiente: a) La altura de una torre de adsorción industrial que debe operar de manera continua durante 24 horas; b) El diámetro de la torre industrial si el flujo que debe tratar es de 7500 cm³/s.
16. Se llevó a cabo una prueba experimental a 23 °C, para remover vapor de agua de una corriente de gas nitrógeno empleando una torre de lecho fijo rellena con zeolitas.

La columna tenía una altura de 0.268 m y un diámetro de 5.5 cm. El flujo másico del gas fue de 10.5 kg/h y la concentración inicial de vapor 926×10^{-6} kg de agua/kg de nitrógeno. La densidad aparente del relleno empleado fue de 712.8 kg/m³ y la densidad del nitrógeno 0.00109 g/cm³. Los resultados obtenidos en esta prueba fueron los siguientes:

t (h)	c (kg de H ₂ O/kg N ₂) x 10 ⁶
0.00	0.0
9.00	0.6
9.20	2.6
9.60	21.0
10.00	91.0
10.40	235.0
10.80	418.0
11.25	630.0
11.50	717.0
12.00	855.0
12.50	906.0
12.80	926.0

Considerando un tiempo de ruptura equivalente a $c/c_0 = 0.02$, calcule:

- a) El tiempo total, el tiempo útil, la altura útil y la altura no útil del equipo experimental.
- b) Si el equipo industrial debe funcionar durante 16 horas continuas tratando un flujo de gas igual a 60.2 kg/h, calcule la altura y diámetro que deberá tener el equipo industrial.
- c) Repita el cálculo del inciso anterior pero considerando que el flujo es tratado por dos torres de adsorción funcionando en paralelo en lugar de una.
- d) Si actualmente existe en la industria una torre de adsorción con el diámetro adecuado y una altura de 1.5 m, calcule el tiempo que podría funcionar este equipo en forma continua.

- e) Si el flujo que debe tratar la industria es de 1920 kg/día, empleando durante ese tiempo dos torres de adsorción: una durante las primeras 12 horas y la segunda durante las siguientes 12 (mientras se regenera la primera), calcule la altura y diámetro que deberán tener estas torres.
- f) Calcule la capacidad de saturación de la zeolita empleada.

9. PROCESOS DE INTERCAMBIO IÓNICO

Los procesos de intercambio iónico son básicamente reacciones químicas entre los iones de una solución y los iones de un sólido insoluble.

Aun cuando la adsorción y los procesos de intercambio iónico son operaciones diferentes (la primera es básicamente un proceso físico, mientras que la segunda es un proceso químico), los procedimientos de diseño y/o dimensionamiento son los mismos. Esto se debe a que en ambas operaciones los perfiles de concentración permanecen constantes y sólo se desplazan a lo largo del relleno hacia el final de la columna.

Por lo tanto en ambas operaciones, en todo momento, el proceso sólo tiene lugar una altura fija de relleno (H_{UNB}) y en ambas esa altura se va desplazando a medida que transcurre el tiempo desde el principio hasta el final de la torre. Estas características en común hacen posible el tratar los procesos de intercambio iónico como un caso especial de los procesos de adsorción.

En los procesos de intercambio iónico, un cierto ión es removido de una solución y es reemplazado por otro procedente del sólido, de manera que la neutralidad electrónica se mantiene.

Los primeros sólidos de intercambio iónico empleados en la industria fueron las zeolitas, que como sabemos son cristales naturales y que desde el punto de vista de los procesos de intercambio iónico, tienen la característica de ser intercambiadores de cationes.

Probablemente la aplicación más importante de las zeolitas en la industria sea el ablandamiento del agua empleada en los calderos e intercambiadores de calor. Como se sabe la dureza del agua es causada por la presencia de iones calcio (Ca^{++}), los mismos que con la acción del calor forman depósitos de carbonato de calcio. El carbonato de calcio se incrustan en los equipos disminuyendo su rendimiento por dos razones: a) porque el carbonato de calcio actúa como un aislante y b) por una disminución del volumen disponible para los fluidos.

Los iones calcio del agua dura pueden ser removidos empleando zeolitas cargadas con iones sodio (Na^+). En el proceso los iones calcio son reemplazados por los iones sodio (que como sabemos no causan dureza) de acuerdo a la siguiente reacción:



Donde R representa a la zeolita inerte. Una vez que la zeolita está saturada con iones calcio, se regenerada poniéndola en contacto con una solución concentrada de $ClNa$ (produciéndose así la reacción inversa).

La mayoría de los sólidos de intercambio iónico empleados en la actualidad son polímeros sintéticos. Aquellos que contienen grupos sulfónicos, carboxílicos o fenólicos pueden intercambiar cationes:



Mientras que los polímeros que contienen grupos amina pueden ser empleados para intercambiar aniones:



9.1. Relaciones de equilibrio en los procesos de intercambio iónico

Las relaciones de equilibrio para los procesos de intercambio iónico, son las relaciones de equilibrio químico de la reacción que tiene lugar, por ejemplo para la reacción (2), el equilibrio estará dado por:

$$K = \frac{[NaR][H^+]}{[Na^+][HR]} \quad (4)$$

Se asume que todos los sitios del sólido disponibles para los cationes están llenos ya sea con iones sodio (Na^+) o con cationes hidrógeno (H^+). Por consiguiente la suma de las concentraciones de los dos cationes en el sólido será constante:

$$[\bar{R}] = [NaR] + [HR] = \text{constante} \quad (5)$$

Reemplazando la ecuación (5) en la ecuación (4) resulta:

$$\begin{aligned} K &= \frac{[NaR][H^+]}{[Na^+](\bar{R} - [NaR])} = \frac{[NaR][H^+]}{[Na^+]\bar{R} - [Na^+][NaR]} \\ \Rightarrow K[Na^+][\bar{R}] - K[Na^+][NaR] &= [NaR][H^+] \\ \Rightarrow K[Na^+][\bar{R}] &= [NaR][H^+] + K[Na^+][NaR] = [NaR]([H^+] + K[Na^+]) \\ \Rightarrow [NaR] &= \frac{K[\bar{R}][Na^+]}{[H^+] + K[Na^+]} \quad (6) \end{aligned}$$

o

$$[NaR] = \frac{[\bar{R}][Na^+]}{\frac{[H^+]}{K} + [Na^+]} \quad (6.a)$$

Si además se trata de una solución tampón, donde la concentración de iones H^+ permanece esencialmente constante, entonces la ecuación (6) tiene la forma de la ecuación de Langmuir:

$$q = \frac{q_0 c}{K + c}$$

Esta es una razón más para considerar los procesos de intercambio iónico como un caso particular de los procesos de adsorción.

9.2. Ejemplo

Se realizó un experimento para remover iones cobre de una solución acuosa empleando un polímero sintético. La prueba se efectuó en un tubo de 1 pulgada de diámetro y 1.2 pies de altura, el flujo de solución fue de 1.5 gph. En la prueba se determinó que el punto de ruptura ocurría a los 7 minutos y al integrar los datos obtenidos se determinó que la fracción utilizada hasta el punto de ruptura fue 0.6. Calcule el diámetro y punto de ruptura de una nueva torre que opere en condiciones similares a las del experimento, pero que tenga una altura de 3 pies y trate un flujo de 4.5 gph.

Solución

Como se dijo, el diseño de las torres de intercambio iónico es el mismo que el de las torres de adsorción.

La altura utilizada (HU) del equipo experimental es:

$$H_{UNB} = 0.6 * 1.2 \text{ pies} = 0.72 \text{ pies}$$

Por lo tanto la altura no utilizada (H_{UNB}) es:

$$H_{UNB} = 1.2 - 0.72 = 0.48 \text{ pies}$$

En el nuevo equipo la altura no utilizada seguirá siendo la misma, por lo tanto la altura útil del nuevo equipo será:

$$H_U = 3 - 0.48 = 2.52 \text{ pies}$$

En consecuencia el nuevo punto de ruptura ocurrirá a los:

$$t_b = 2.52 \text{ pies} \frac{7 \text{ min}}{0.72 \text{ pies}} = 24.5 \text{ min}$$

En otras palabras el equipo de 3 pies de altura podrá funcionar de manera continua durante 24.5 minutos.

Para que los datos experimentales puedan ser extrapolados a otro equipo, el flujo superficial debe mantenerse constante. El flujo superficial del equipo experimental es:

$$\text{Flujo superficial} = \frac{1.5 \frac{\text{gal}}{h}}{\frac{\pi}{4} (1 \text{ plg})^2} = 1.90986 \frac{\text{gal}}{h \cdot \text{plg}^2}$$

Entonces la sección transversal del nuevo equipo deberá ser:

$$\text{Area transversal} = \frac{4.5 \frac{\text{gal}}{h}}{1.90986 \frac{\text{gal}}{h \cdot \text{plg}^2}} = 2.35619 \text{ plg}^2$$

Y el diámetro:

$$\text{Diámetro del nuevo equipo} = \sqrt{\frac{4 * 2.35619}{\pi}} = 1.73205 \text{ plg}$$

9.3. Preguntas y ejercicios

1. ¿Qué son los procesos de intercambio iónico?
2. ¿Por qué se dice que la adsorción y los procesos de intercambio iónico son operaciones diferentes?
3. ¿Por qué se emplean los mismos procedimientos de diseño para la adsorción y los procesos de intercambio iónico?
4. ¿Cuál es la principal aplicación industrial de las zeolitas en los procesos de intercambio iónico?
5. ¿Por qué causan los iones calcio la dureza del agua?
6. ¿Qué polímeros sintéticos son adecuados para el intercambio iónico para cationes?
7. ¿Qué polímeros sintéticos son adecuados para el intercambio iónico de aniones?

- 8. ¿A qué ecuación se asemejan las ecuaciones de equilibrio de los procesos de intercambio iónico?
- 9. En una prueba experimental se determinó que el tiempo de ruptura para un proceso de intercambio iónico, en una columna de 0.4 m de altura y 1.5 pulgadas de diámetro, fue de 8 min. En la prueba se empleó un flujo de solución igual 0.2 m³/h y se llegó a utilizar el 65% de la altura total. Calcule la altura y diámetro de un equipo industrial que debe operar de manera continua durante 2 horas tratando un flujo de solución igual a 1.35 m³/h.
- 10. Se realizaron pruebas experimentales en una columna de intercambio iónico que empleaba 99.3 g de amberlita para remover iones cobre (Cu⁺⁺) de una solución 0.18 M de sulfato de cobre (CuSO₄). La torre tenía una altura de 30.5 cm, un diámetro de 2.59 cm y el flujo de solución empleado fue de 1.37 cm³/s. Los datos obtenidos en la prueba son los siguientes:

t (s)	c (g mol/l)
420	0.0000
480	0.0033
510	0.0075
540	0.0157
600	0.0527
660	0.1063
720	0.1433
780	0.1634
810	0.1722
870	0.1763
900	0.1800

La concentración deseada en el punto de ruptura es $c/c_0=0.010$.

- a) Calcule el tiempo de ruptura, el tiempo total equivalente a la altura total del equipo, la altura útil, la altura no útil y la fracción utilizada del equipo.
- b) Si el equipo industrial debe operar de manera continua durante 4 horas y el flujo que debe tratar es de 1 litro por segundo, calcule la altura y diámetro que deberá tener el equipo industrial.
- c) Si la industria cuenta con una torre de intercambio iónico de 1.5 m de altura y 0.5 m de diámetro, calcule el tiempo que podría funcionar de manera continua y el flujo que podría tratar.
- d) Calcule la capacidad de saturación de la amberlita.

10. SECADO DE MATERIALES

El secado de materiales hace referencia a la eliminación de cantidades relativamente pequeñas de humedad de un sólido. Con frecuencia se confunden los procesos de secado y el de evaporación, sin embargo, en el proceso de evaporación se eliminan cantidades mas bien grandes de líquido y además, dicha eliminación ocurre casi siempre por ebullición del líquido, mientras que en el secado de materiales, la humedad se elimina normalmente a temperaturas menores a las del punto de ebullición con la intervención de un gas (usualmente aire) que arrastra consigo la humedad del sólido.

El secado de materiales puede presentarse en una industria tanto en el tratamiento de las materias primas como en los productos terminados. En el primer caso se elimina la humedad de los materiales para optimizar el rendimiento de los procesos y equipos, por ejemplo, en muchas operaciones de molienda, es necesario secar el material pues la molienda sólo es posible si el contenido de humedad se encuentra por debajo de cierto límite, de manera similar, en la extracción de aceites esenciales de plantas, las plantas deben ser previamente trituradas a fin de romper las paredes celulares, pero esto sólo es posible en plantas secadas, pues en las húmedas, lo único que se consigue es extraer líquidos. Por otra parte, muchos productos acabados tienen que ser secados para poder ser comercializados, por una parte, porque las normas así lo exigen y por otra porque de esa manera se facilita la conservación y transporte de los productos.

En las industrias de los alimentos con frecuencia se emplea el secado para conservar los alimentos durante periodos prolongados. Se sabe que la mayoría de las bacterias no pueden proliferar si el contenido de humedad del material es menor o igual al 10%, sin embargo, para prevenir la acción de otras enzimas y algunas bacterias resistentes normalmente se secan los productos hasta contenidos de humedad menores al 5%.

10.1. EQUIPOS PARA EL SECADO

Los equipos de secado, en función a la forma en la que operan suelen clasificarse en discontinuos o por lotes y en continuos. Por la forma en que se transfiere el calor hacia el líquido los equipos se clasifican en directos, cuando el calor es transmitido por conducción e indirectos, cuando el calor es transmitido por convección a través de un gas.

Los equipos continuos básicamente consisten en secadores de bandejas, donde se hace circular el aire a través de las bandejas en algunos casos en paralelo, en otras transversalmente y en otras una combinación de ambas. El aire que circula a través de las bandejas es calentado por un intercambiador de calor, calentado generalmente con tubos de vapor de agua. Puesto que el aire se satura a medida que recorre las bandejas, es necesario alimentar continuamente aire fresco y eliminar la misma cantidad de aire saturado. El porcentaje de aire fresco que se alimenta al equipo varía entre el 10 y el 20% del total que pasa por los intercambiadores de calor. El material que se coloca en las bandejas tiene espesores que varían entre 1 y 10 cm. Desde el punto de vista de la transferencia de calor estos equipos son principalmente indirectos, aunque parte del calor es transmitido también por conducción y por radiación.

Existen multitud de variante en los secadores de bandeja, probablemente la más interesante sea aquella que emplea carretillas de bandejas, esto permite introducir al equipo todas las bandejas al mismo tiempo y cargar una carretilla mientras se seca el contenido de otra. De esta manera se consigue

un considerable ahorro de tiempo, pero también de energía, pues el equipo no sufre un gran pérdida de energía en la alimentación de las bandejas.

Otra variación de los secadores de bandejas ocurre cuando es necesario secar al vacío debido a que el material a secar es muy sensible a la temperatura. En estos casos los secadores de bandejas son equipos herméticos, donde se produce el vacío y la transferencia de calor se produce directamente por conducción a través de las bandejas aunque cierta cantidad de calor se transfiere también por radiación.

En cuanto a los secadores continuos existen varios tipos de equipos entre los que podemos mencionar los de túnel, los de banda transportadora, los secadores de granos, los rotatorios, los de tambor y los de aspersion.

Los secadores de túnel son básicamente secadores de bandejas, donde las mismas se encuentran en carretillas que pasan a través de un túnel, en el que se hace circular aire caliente en contracorriente. Estos equipos son diseñados de forma tal que al llegar la carretilla al final del túnel, el material de las bandejas se encuentre ya seco.

En los secadores de banda transportadora el material a secar se alimenta a la banda, la cual transporta el material a través del equipo de secado. La banda transportadora generalmente consiste de una malla en la cual el material a secar tiene un espesor que varía entre 2.5 y 15 cm. y donde el aire de secado se hace circular a través de la banda, en contracorriente o mediante una combinación de ambas. El equipo está diseñado de tal manera que cuando el material alcanza el final de la banda está ya seco.

Los secadores rotatorios consisten de un cilindro rotatorio ligeramente inclinado. El material a secar se alimenta por uno de los extremos y a medida que atraviesa el equipo el material es elevado, debido al movimiento de rotación y gracias a la ayuda de unas aspas que se encuentran en el interior del cilindro, hasta la parte superior del mismo de donde caen y al caer atraviesan la corriente de aire caliente que circula por el interior del cilindro. El calor es transferido principalmente de forma indirecta, por convección a través del aire caliente, aunque puede ser transmitido también por conducción cuando las paredes del cilindro son paredes enchaquetadas en cuyo interior circula vapor de agua. Al igual que los anteriores equipos, los secadores rotatorios son diseñados de forma tal que el material queda seco cuando alcanza el otro extremo del cilindro.

Los secadores de tambor consisten de cilindros metálicos en cuyo interior circula vapor de agua. Estos tambores son giratorios y a medida que giran recogen por adhesión una solución o pasta en la que se encuentran parcialmente sumergidos. Es esta solución o pasta que se seca a medida que el tambor gira. Estos equipo se diseñan de forma tal que el material que se adhiere al mismo quede seco cuando el tambor termina de dar un giro. El material seco es removido por unos raspadores antes de que esa parte del tambor vuelva a sumergirse en la solución o pasta. Si es importante que el material secado tenga un espesor uniforme se utilizan rodillos aplanadores, los que le dan uniformidad al material, en ocasiones se pueden utilizar también tambores dobles y al pasar la pasta o solución a través de los mismos adquiere un espesor uniforme. En estos equipos la alimentación también puede ser realizada por la parte superior.

Los secadores de granos son equipos que se emplean en la industria agrícola para secar los granos cosechados hasta aproximadamente un 13% de manera que puedan ser almacenados sin problemas durante un periodo de aproximadamente un año. Constan de un cilindro hueco que termina en un cono que se encuentra dentro de un cilindro cuyas paredes son casi siempre una malla de alambre. Los granos se alimentan por la parte superior (por el cono) y son

distribuidos alrededor del cilindro hueco, a medida que van descendiendo por gravedad, se hace circular aire caliente, que sale del cilindro, a través de los granos, con lo que se logra el secado requerido, posteriormente, se hace circular aire frío, que también sale del cilindro para enfriar los granos hasta una temperatura adecuada para su almacenamiento. El espacio entre el cilindro hueco interior y la malla es de aproximadamente medio metro.

10.2. CONTENIDO DE HUMEDAD EN EL EQUILIBRIO

Si un sólido húmedo se deja en un cuarto en contacto con una gran cantidad de aire no saturado a temperatura constante, irá perdiendo humedad hasta alcanzar un contenido de humedad que permanece constante. Entonces se dice que el sólido ha alcanzado una humedad que está en equilibrio con el aire a esa temperatura y humedad.

Por otra parte si un sólido completamente seco se deja en el mismo cuarto, irá adsorbiendo humedad hasta que alcanza un contenido de humedad constante. Entonces, una vez más, el sólido tiene una humedad que está en equilibrio con el aire a esa temperatura y humedad. Sin embargo, en el proceso de secado suele suceder que la humedad de equilibrio que se alcanza cuando el sólido pierde humedad no es igual a la humedad de equilibrio que se alcanza cuando el sólido la gana, ha este fenómeno se conoce como histéresis y por lo general, la humedad de equilibrio del sólido que pierde humedad es mayor a la humedad de equilibrio cuando del sólido que la gana.

10.2.1. Datos de equilibrio para materiales orgánicos e inorgánicos