

79
29



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE QUIMICA

**CALCULO DEL EQUILIBRIO
EN SISTEMAS DE VARIAS REACCIONES
USANDO ECUACIONES DE ESTADO**



TESIS
para obtener el título de **Ingeniero Químico**
presenta

EXAMENES PROFESIONALES
FAC. DE QUIMICA

IGNACIO QUESADA MORALES

1989

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

I N D I C E

CAPITULO 1:	Introducción	pg 1
CAPITULO 2:	Planteamiento general del problema de equilibrio.	pg 3
CAPITULO 3:	Solución por el método de minimización de la energía de Gibbs	pg 6
CAPITULO 4:	Solución por el método de constantes de equilibrio	pg 17
CAPITULO 5:	Aplicaciones	pg 23
CAPITULO 6:	Análisis de resultados	pg 25
CAPITULO 7:	Conclusiones	pg 31
APENDICE 1:		pg 33
APENDICE 2:		pg 37
APENDICE 3:		pg 39
APENDICE 4:		pg 43
APENDICE 5:		pg 63

CAPITULO 1

Introducción

La transformación de materias primas en productos de un mayor valor por medio de reacciones químicas es algo de vital importancia en la industria química. El cálculo de equilibrio químico es un elemento necesario para el diseño y operación de reactores en los cuales se llevan a cabo estas transformaciones. Si bien en la realidad existen sistemas en los cuales el factor dominante es la cinética de las reacciones, el cálculo del equilibrio químico sigue siendo necesario en el desarrollo comercial de procesos químicos, ya que indica en que sentido se dará la transformación y además la limitante para obtener la máxima conversión posible. La temperatura, presión y composición afectan tanto la velocidad de reacción como la conversión al equilibrio. Es por esto que al hacer un análisis de las mejores condiciones de operación, algunas veces se encuentra que no es posible operar a las condiciones en las cuales se favorezca la cinética de las reacciones, ya que la conversión al equilibrio sería pequeña. Un ejemplo de esto es el caso de la oxidación de SO_2 a SO_3 en presencia de V_2O_5 , en el cual con un aumento de temperatura se incrementa la velocidad de reacción pero la conversión al equilibrio disminuye.

Por estas razones el cálculo de equilibrio químico es necesario y ha sido ampliamente tratado por diversos métodos. Si bien existen soluciones particulares, como los métodos gráficos, que son sencillos y prácticos cuando el número de reacciones es pequeño, el cálculo se vuelve tedioso cuando el número de reacciones y especies químicas presentes es grande.

En la tesis se plantean dos métodos diferentes de solución y se desarrollaron los programas. Uno de los métodos consiste en el cálculo de constantes de equilibrio y su resolución para una serie de reacciones. El otro método se basa en encontrar el mínimo de la

energía de Gibbs. De estos dos métodos se desarrollaron los programas para su aplicación en computadora. Estos programas pueden ser una herramienta muy útil para los cursos de Termodinámica Química, ya que le darán al estudiante la posibilidad de analizar un mayor número de ejemplos. Además podrá hacer una serie de cálculos y consideraciones al resolver los ejemplos que en otras condiciones sería difícil realizar por el gran número de operaciones necesarias.

En este trabajo las aplicaciones se limitan a una fase reaccionante multicomponente con la presencia de fases sólidas puras. Los procedimientos de solución de este trabajo pueden ser fácilmente extendidos a varias fases multicomponentes y agregarles el cálculo de coeficientes de actividad para fases líquidas.

Se incluye el uso de ecuaciones de estado cúbicas (Peng-Robinson, Soave) para el cálculo de coeficientes de fugacidad. Una de las ventajas de este tipo de ecuaciones es que pueden ser aplicadas a fase gaseosa y también a fase líquida. Además se usa la modificación de Mathias y la de Stryjek-Vera para el caso de componentes polares.

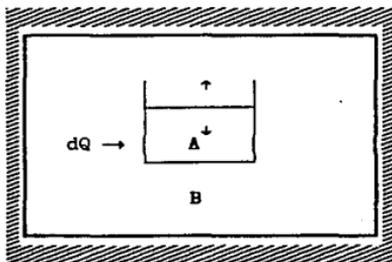
Se hace la comparación de los dos métodos para ver su velocidad de convergencia y comprobar que se llega a soluciones iguales.

CAPITULO 2
Planteamiento general del problema
del equilibrio

Equilibrio es algo que denota una condición estática, ausencia de movimiento, de cualquier tendencia a un cambio macroscópico en un sistema. Como todo cambio es debido a una fuerza, en el equilibrio no existe una fuerza que varíe las condiciones del sistema, es decir que las fuerzas existentes están en equilibrio. Con el tiempo los sistemas aislados tienden al equilibrio.

De la segunda ley de la Termodinámica se sabe que en un sistema aislado, el estado de equilibrio es aquel para el cual la entropía total del sistema es máxima.

El caso que nos interesa considerar es aquel en el cual se tiene un sistema que está a temperatura, presión y número de átomos-mol de cada elemento constantes. Si se considera que el sistema A en estudio se encuentra inmerso en un macrosistema B que está aislado del exterior, el estado de equilibrio del sistema A+B corresponderá a las condiciones para las cuales el sistema A por separado se encuentra en equilibrio.



Por la segunda ley de la Termodinámica se sabe:

$$dS \geq 0$$

2.1

$$dS_T = dS_A + dS_B \geq 0$$

2.2

Para el macrosistema B

$$dS_B = -dQ/T \quad 2.3$$

De la primera ley de la termodinámica, se tiene para el sistema A que:

$$dU_A = dQ - PdV_A \quad 2.4$$

despejando dQ:

$$dQ = dU_A + PdV_A \quad 2.5$$

Sustituyendo esta ecuación (2.5) en la ecuación (2.3) se obtiene

$$dS_B = - (dU_A + PdV_A)/T \quad 2.6$$

y sustituyendo a su vez este resultado en la ecuación (2.2), se obtiene:

$$dS_A - (dU_A + PdV_A)/T \geq 0 \quad 2.7$$

Como se considera que la temperatura y la presión son constantes

$$d(TS_A - U_A - PV_A)/T \geq 0 \quad 2.8$$

Cambiando el signo del lado izquierdo de la expresión se invierte el signo de la desigualdad y como la temperatura es siempre positiva, se puede multiplicar por su valor sin alterar la desigualdad, dando como resultado la siguiente expresión:

$$d(U_A + PV_A - TS_A) \leq 0 \quad 2.9$$

La definición de la energía de Gibbs, G, es:

$$G = U + PV - TS \quad 2.10$$

donde G es una propiedad de estado ya que es originada por propiedades de estado. La condición de equilibrio para el sistema A a temperatura y presión constantes se expresa entonces por la siguiente relación:

$$dG \leq 0 \quad 2.11$$

o integrando

$$\Delta G \leq 0 \quad 2.12$$

En la ecuación (2.11) el signo de igualdad nos da la condición de equilibrio, de este modo el equilibrio está dada por un mínimo de la energía de Gibbs. De los diversos criterios existentes de equilibrio

los más empleados son aquellos que tienen que ver con la energía de Gibbs. Esto se debe a que la energía de Gibbs está basada en temperatura y presión, que son propiedades tangibles del sistema. Además existe la facilidad que una vez calculada la energía de Gibbs el signo algebraico de ésta, nos indica el sentido de la transformación. En caso de ser negativa nos indica que la transformación puede ocurrir espontánea o naturalmente en el sentido planteado. Si es positiva indica que la transformación no es espontánea, es decir que la dirección natural del proceso es en sentido contrario. En caso de ser igual a cero el sistema está en equilibrio con respecto a la transformación planteada.

CAPITULO 3

Solución por el método de minimización de la energía de Gibbs

Como anteriormente se explicó, el equilibrio se obtiene cuando $dG=0$, es decir, el sistema se encuentra en el mínimo de la energía de Gibbs.

En un sistema multicomponente la energía de Gibbs estará dada por la sumatoria de los potenciales químicos μ_i de cada componente multiplicados por el número de moles N_i presentes.

$$G = \sum N_i \mu_i \quad 3.1$$

Considerando que el sistema reaccionante está formado por una fase multicomponente y varias fases sólidas puras se tiene que:

$$\text{fase multicomponente} \quad \mu_i = g^i + RT \ln a^i \quad i=1, 2, \dots, C \quad 3.2$$

$$\text{fase sólida pura} \quad \mu_i = g^i(T) \quad i=C+1, C+2, \dots, C+S \quad 3.3$$

donde C es el número de especies químicas presentes en la fase multicomponente, S es el número de especies químicas en estado sólido puro y que además no están presentes en la fase multicomponente, g^i es la energía de Gibbs de formación del componente puro i a la temperatura T del sistema y en su estado de referencia, y a^i es la actividad del componente i en la fase multicomponente. Para las fases sólidas el estado de referencia es el sólido puro a 1 atm y para los componentes en la fase multicomponente es el gas ideal a 1 atm. En el apéndice 2 se presenta la deducción de las ecuaciones (3.2 y 3.3) y se presenta el cálculo de g^i en función de T.

La energía de Gibbs total está dada por:

$$G = \sum_{i=1}^C N_i (g^i + RT \ln a^i) + \sum_{i=C+1}^{C+S} N_i g^i \quad 3.4$$

La actividad a^i en el caso de la fase multicomponente estará dada en

términos generales por:

$$a_i = f_i / f_i^* \quad 3.5$$

donde f_i es la fugacidad del componente i en la fase multicomponente y f_i^* es la fugacidad del componente puro en su estado de referencia. Si se tiene como estado de referencia gas ideal a 1 atm entonces $f_i^* = 1$ atm. La fugacidad f_i se calcula mediante:

$$f_i = P y_i \phi_i \quad 3.6$$

donde P es la presión total del sistema, y_i es la fracción mol del componente i en la fase multicomponente y ϕ_i es el coeficiente de fugacidad del componente i . Si la fase multicomponente se considera como gas ideal entonces $\phi_i = 1$. Para el caso de gases reales o líquidos en este trabajo se incluye el cálculo de ϕ_i por medio de una ecuación de estado.

Como sabemos que $y_i = N_i / N$, y que ϕ_i es función de T , P y y_i , entonces la energía de Gibbs es función exclusivamente de los moles de cada especie química:

$$G = G(N_1, N_2, \dots, N_{c+s}) \quad 3.7$$

De modo que el problema se reduce a encontrar el conjunto de N_i que minimice la energía de Gibbs G a la temperatura y presión indicadas. La solución está limitada ya que no todas las combinaciones de N_i son aceptables. Estas N_i están restringidas por los balances de materia. Aunque en un sistema reaccionante el número de moles de las especies químicas no se conserva, si se conserva el número total de átomos de cada elemento químico. Entonces deben cumplirse las siguientes relaciones:

$$\sum_{i=1}^{c+s} A_{ji} N_i - B_j = 0; \quad j = 1, 2, \dots, M \quad 3.8$$

donde A_{ji} es el número de átomos del elemento j en la especie química i , M es el número de elementos químicos presentes y B_j es el número

total de átomos-mol del elemento j en la mezcla reaccionante. Cuando hay elementos inertes presentes en la mezcla de reacción la suma anterior es extendida a incluir todos los componentes presentes. Cada material inerte es tratado como un nuevo elemento que no aparece en ninguna otra especie química.

Quando se tiene que encontrar el mínimo o el máximo de la función G que está sujeta a la serie de M condiciones laterales, como los dados por las ecuaciones (3.8), existen varias formas de proceder. Una manera es considerar las condiciones laterales como M ecuaciones simultaneas a partir de las cuales, al menos teóricamente, M de las variables pueden expresarse en términos de las restantes. Entonces por sustitución pueden eliminarse estas M variables de la función a minimizar. Por último puede determinarse el mínimo o el máximo de G como función de las variables restantes. Esto se conoce como el método explícito.

Otro método es el conocido como método implícito. En este método se calculan las diferenciales de las M condiciones laterales. Entonces se resuelven simultaneamente estas M ecuaciones y por sustitución se eliminan de dG estas M diferenciales. Por último, se iguala a cero el coeficiente de cada una de las diferenciales restantes que quedan en dG , y estas ecuaciones, junto con las M condiciones laterales, nos proporcionan un sistema de ecuaciones a partir del cual pueden hallarse todas las posibilidades para los valores máximos o mínimos de G .

Un tercer procedimiento, menos obvio pero generalmente más conveniente que los dos anteriores, es el uso de los multiplicadores de Lagrange.

Este método requiere que se establezca la función auxiliar:

$$\Phi(N_i) = G(N_i) + \sum_{j=1}^M \Pi_j \Psi_j(N_i) \quad 3.9$$

Donde las Ψ_j son las expresiones que definen las M condiciones laterales:

$$\Psi_j = B_j - \sum A_{ji} N_i; \quad j=1,2,\dots,M \quad 3.10$$

y las Π_j (los multiplicadores de Lagrange) son parámetros desconocidos que deberán determinarse. Entonces se calculan las derivadas parciales de Φ con respecto a las N_i y se igualan a cero. Finalmente se resuelven estas ecuaciones, junto con las M ecuaciones de restricción. A primera vista, el uso de los multiplicadores de Lagrange puede parecer ineficaz, puesto que se tiene que resolver un sistema de ecuaciones más grande. Sin embargo, generalmente este es el procedimiento más conveniente, debido a que no se tiene que decidir cuales variables deben considerarse como independientes y cuales se deben eliminar, y además a menudo nos permite sacar ventaja de la simetría que se logra en el sistema de ecuaciones.

Para encontrar el mínimo de G usamos el método de descenso precipitado. La idea básica de este método es la siguientes: dada una aproximación N al mínimo N^* de G , uno busca el mínimo de G más cercano a N sobre una línea recta, desde N sobre la dirección de $\nabla G(N)$. Con esto obtenemos una nueva aproximación y continuamos hasta que el grado de convergencia deseado sea alcanzado.

Comenzamos suponiendo que tenemos alguna solución aproximada (después de k iteraciones) para el número de moles N_1, N_2, \dots, N_{c+1} , que satisface la ecuaciones de conservación de masa. De este modo la función de la energía de Gibbs está dada por:

$$G/RT = F = \sum_{i=1}^k N_i (g_i^0/RT + \ln a_i) + \sum_{l=c+1}^k N_l g_l^0/RT \quad 3.11$$

donde el supraíndice k representa el valor de variable en la iteración k .

Si se toma como estado de referencia al gas ideal a 1 atm. entonces

$$a_i = \phi_i^k P^k / N^k \quad 3.12$$

donde N es el número total de moles en la fase multicomponente en la iteración k:

$$N^k = \sum_{i=1}^c N_i^k \quad 3.13$$

Definiendo L_i de la siguiente manera:

$$\text{fase multicomponente} \quad L_i = g_i^k / RT + \ln P^k + \ln \phi_i^k \quad 3.14$$

$$\text{fases solidas} \quad L_i = g_i^k / RT \quad 3.15$$

la ecuación 3.11 queda expresada en la siguiente forma:

$$G^k / RT = F^k = \sum_{i=1}^k N_i^k (L_i + \ln N_i^k / N^k) + \sum_{i=c+1}^k N_i^k L_i \quad 3.16$$

Expandiendo por series de Taylor la función de la energía de Gibbs alrededor de N_i^k de la iteración k, y teniendo hasta términos de segundo orden se obtiene:

$$F^k = F^k + \sum_{i=1}^{k+1} (N_i^k - N_i^k) \partial F^k / \partial N_i^k + 1/2 \sum_{i=1}^{k+1} \sum_{l=1}^{k+1} (N_i^k - N_i^k)(N_l^k - N_l^k) \partial^2 F^k / \partial N_i^k \partial N_l^k \quad 3.17$$

Realizando las derivadas necesarias y después de algunas simplificaciones se tiene que:

para $1 \leq i \leq C$

$$\partial F^k / \partial N_i^k = [L_i + \ln(N_i^k / N^k)] \quad 3.18$$

$$\partial^2 F^k / \partial N_i^k \partial N_l^k = 1/N_i^k - 1/N^k \quad 3.19$$

$$0 \quad \text{para } C+1 \leq i \leq C+S \quad 3.20$$

$$\partial^2 F^k / \partial N_i^k \partial N_l^k =$$

$$-1/N^k \quad \text{para } 1 \leq i \leq C \text{ y } l \neq i \quad 3.21$$

En las ecuaciones anteriores se ha considerado que ϕ_i^k en la iteración k no es función de la composición, esto es, una constante, y por lo tanto L_i de la iteración k es constante al momento de obtener la derivada con respecto a N_i^k . Esta aproximación es válida cuando la

fase multicomponente se aproxima a una solución ideal (de gases reales). En caso de que ϕ_i sea fuertemente dependiente de la composición, entonces la convergencia de este método será lenta y habrá que formularlo incluyendo la variación de ϕ_i con la composición.

para $C+1 \leq C+S$

$$\frac{\partial^k F}{\partial N_i^k} = L_i^k \quad 3.22$$

$$\frac{\partial^2 F}{\partial N_i \partial N_j} = 0 \quad \text{para } 1 \leq i \leq C+S \quad 3.23$$

Sustituyendo estas derivadas en la serie de Taylor (3.17) se obtiene la siguiente ecuación:

$$F^{k+1} = \sum_{i=1}^c N_i^k (L_i + \ln(N_i/N)) + \sum_{i=c+1}^k N_i^k L_i + \sum_{i=1}^k (N_i^k - N_i^k) (L_i + \ln(N_i/N)) + \sum_{i=c+1}^k (N_i^k - N_i^k) L_i + \frac{1}{2} \sum_{i=1}^k N_i^k ((N_i^k - N_i^k)/N_i - (N_i^k - N_i^k)/N) \quad 3.24$$

Ahora utilizamos los multiplicadores de Lagrange para introducir los balances de masa como restricciones a la solución:

$$\Phi^{k+1} = F + \sum_{j=1}^{k+1} \pi_j (B_j - \sum_{i=1}^{c+s} A_{ij} N_i) \quad 3.25$$

Mientras los balances de masa se cumplan $\Phi = F$, y entonces el mínimo de Φ es también el mínimo de F . La condición para encontrar el mínimo es que todas las derivadas parciales de Φ con respecto a las N_i de la iteración $k+1$ sean iguales a cero.

$$\text{para } 1 \leq i \leq C+S \quad \frac{\partial \Phi^{k+1}}{\partial N_i} = 0 \quad 3.26$$

Diferenciando la expresión de Φ de la iteración $k+1$ con respecto a N_i de la iteración $k+1$ obtenemos:

para $1 \leq i \leq C$

$$\frac{\partial \Phi^{k+1}}{\partial N_i} = L_i + \ln(N_i/N) + (N_i^k - N_i^k)/N_i - (N_i^k - N_i^k)/N - \sum_{j=1}^m \pi_j A_{ij} = 0 \quad 3.27$$

para $C+1 \leq i \leq C+S$

$$\frac{\partial \Phi}{\partial N_i} = L_i - \sum_{j=1}^{k+m} \pi_j A_{ji} = 0 \quad 3.28$$

Además de la C+S ecuaciones anteriores, se tienen los balances de masa para cada elemento:

$$\text{para } 1 \leq j \leq m \quad B_j - \sum_{i=1}^{c+s} A_{ji} N_i = 0 \quad 3.29$$

Por último, la definición de N

$$N = \sum_{i=1}^{k+s} N_i \quad 3.30$$

Las ecuaciones anteriores forman un sistema de C+S+M+1 ecuaciones donde las incógnitas son los C+S moles de las especies presentes N_i de la iteración $k+1$, los M multiplicadores de Lagrange π_j y el número total de moles de la fase multicomponente N de la iteración $k+1$.

Este sistema de ecuaciones a ser resuelto se reduce eliminando algunas de las incógnitas. Para lograr esto se procede como sigue. De las derivadas parciales de Φ de la iteración $k+1$ con respecto a N_i , conservamos las ecuaciones para $C+1 \leq i \leq C+S$. En las ecuaciones para $1 \leq i \leq C$ despejamos N_i de la iteración $k+1$, obteniendo:

$$N_i = N_i \left[\frac{N}{N} + \sum_{j=1}^{k+m} \pi_j A_{ji} - (L_i + \ln(N_i/N)) \right] \quad 3.31$$

haciendo la suma sobre todas las $1 \leq i \leq C$, y recordando que $N = \sum N_i$

$$N = \sum_{i=1}^{k+s} N_i \left[\frac{N}{N} + \sum_{j=1}^{k+m} \pi_j A_{ji} - \sum_{i=1}^{k+s} N_i (L_i + \ln(N_i/N)) \right] \quad 3.32$$

simplificando

$$\sum_{i=1}^{c} N_i \sum_{j=1}^{k+m} \pi_j A_{ji} = \sum_{i=1}^{c} N_i (L_i + \ln(N_i/N)) \quad 3.33$$

se sabe que para $C+1 \leq i \leq C+S$

$$\sum_{j=1}^m \pi_j A_{ji} = L_i \quad 3.34$$

multiplicando esta ecuación (3.34) por N_i de la iteración k , y sumando sobre i , obtenemos:

$$\sum_{i=1}^{c+s} N_i \sum_{j=1}^m \pi_j A_{ji} = \sum_{i=1}^k N_i L_i \quad 3.35$$

l=0+1 j=1 l=0+1

sumando esta última ecuación (3.35) con la ecuación (3.33) obtenemos

$$\sum_{l=1}^{c+m} N_l \sum_{j=1}^m \Pi_j A_{jl} = \sum_{l=1}^c N_l (L_l + \ln(N_l/N)) + \sum_{l=c+1}^m N_l A_{ll} \quad 3.36$$

se cambia el orden de las sumas del lado izquierdo de la última ecuación (3.36) se tiene

$$\sum_{l=1}^{c+m} N_l \sum_{j=1}^m \Pi_j A_{jl} = \sum_{j=1}^m \Pi_j \sum_{l=1}^{c+m} N_l A_{jl} \quad 3.37$$

como las N_l de cualquier iteración satisfacen los balances de masa, tenemos que

$$\sum_{l=1}^{c+m} N_l \sum_{j=1}^m \Pi_j A_{jl} = \sum_{j=1}^m \Pi_j B_j \quad 3.38$$

y sustituyendo en la ecuación original (3.36), obtenemos

$$\sum_{j=1}^m \Pi_j B_j = \sum_{l=1}^m N_l (L_l + \ln(N_l/N)) + \sum_{l=c+1}^m N_l L_l \quad 3.39$$

En los balances de masa (3.29) sustituimos las ecuaciones (3.31) para N_l de la nueva iteración de las especies presentes en la fase multicomponente, y conservamos las N_l para las fases sólidas

$$\sum_{l=1}^c A_{jl} [N_l + N_l/N + N_l \sum_{j=1}^m \Pi_j A_{jl} - N_l (L_l + \ln(N_l/N))] + \sum_{l=c+1}^{k+1} A_{jl} N_l = B_j \quad 3.40$$

se despejan las incógnitas en el lado derecho de las ecuación (3.40)

$$\sum_{l=1}^c A_{jl} N_l \sum_{l=1}^m \Pi_j A_{ll} + \sum_{l=c+1}^{k+1} A_{jl} N_l + N_l \sum_{l=1}^c A_{jl} N_l / N = B_j + \sum_{l=1}^m N_l (L_l + \ln(N_l/N)) \quad 3.41$$

Cambiando el orden de la sumatoria del primer término de la derecha de la última ecuación (3.41) tenemos:

$$\sum_{l=1}^c A_{jl} N_l \sum_{l=1}^m \Pi_j A_{ll} = \sum_{l=1}^m \sum_{l=1}^c A_{jl} A_{ll} N_l \Pi_j \quad 3.42$$

Definimos

$$R_{jl} = \sum_{l=1}^c A_{jl} A_{ll} N_l \quad 3.43$$

sustituyendo (3.43) y (3.42) en la ecuación (3.41) obtenemos

para $1 \leq j \leq m$

$$\sum_{l=1}^m R_{jl} \Pi_j + \sum_{l=1}^{c+m} A_{jl} N_l + N_l \sum_{l=1}^c A_{jl} N_l / N = B_j + \sum_{l=1}^m N_l (L_l + \ln(N_l/N)) \quad 3.44$$

que con las ecuaciones (3.34)

para $C+1 \leq i \leq C+S$

$$\sum_{j=1}^m \pi_j A_{ij} = L_i^k$$

y por último la ecuación (3.39)

$$\sum_{j=1}^m \Pi_j B_j = \sum_{i=1}^c (N_i^k (L_i^k + \ln(N_i^k / N^k))) + \sum_{i=C+1}^{C+S} N_i^k L_i^k$$

forman un sistema de $M+S+1$ ecuaciones, que es resuelto para las $M+S+1$ incógnitas que son los M multiplicadores de Lagrange, los S números de moles de los sólidos puros y el número total de moles de la fase multicomponente. Las N_i de la iteración $k+1$ para la fase multicomponente son obtenidos a partir del número total de moles de la iteración, por medio de la ecuación (3.31):

$$N_i^{k+1} = N_i^k \left[\frac{N^k}{N^{k+1}} + \sum_{j=1}^m \Pi_j A_{ij} - (L_i^k + \ln(N_i^k / N^k)) \right]$$

En esta ecuación se observa que no se puede dar cero como suposición inicial del número de moles N_i , ya que esto ocasionaría que en todas las subsecuentes iteraciones no existan moles de este componente.

Un problema que se presenta durante la solución, consiste en que en ocasiones se obtienen valores negativos de N_i para la nueva iteración y como no podemos admitir estos valores negativos del número de moles para ningún componente, es necesario limitar el cambio del número de moles de una iteración a otra. Esto se logra mediante un procedimiento sencillo que además nos ayuda a disminuir el número de iteraciones necesarias para obtener la convergencia deseada.

Definamos los ΔN_i como la diferencia de N_i en la nueva iteración y el valor anterior

$$\Delta N_i = N_i^{k+1} - N_i^k; \quad i=1, 2, \dots, C+S \quad 3.45$$

Estos ΔN_i son números que nos indican la mejor dirección de cambio, pero no necesariamente el mejor tamaño de este cambio. El tamaño del

cambio en las moles de cada compuesto se obtiene mediante un parámetro λ de la siguiente manera:

$$N_i^{k+1} = N_i^k + \lambda \Delta N_i \quad 3.46$$

El valor de λ será escogido como el número mas grande posible que satisfaga las siguientes dos condiciones: (1) que todas las N_i de la nueva iteración $k+1$ sean positivas, y (2) que la derivada de G/RT con respecto a λ no se vuelva positiva (esto es que no nos pasemos del mínimo). Esta última derivada es facilmente calculada y es:

$$d(G/RT)/d\lambda = \sum_{i=1}^c \Delta N_i (L_i^k + \ln((N_i^k + \Delta N_i)/(N_i^k + \Delta N_i))) + \sum_{i=c+1}^k \Delta N_i L_i^k \quad 3.48$$

donde

$$\Delta N = N^{k+1} - N^k$$

El cálculo de λ se hace por medio del procedimiento de Newton-Raphson, limitado por el valor máximo posible que no haga negativa ninguna N_i de la nueva iteración $k+1$. En caso de que el valor de λ que minimiza G/RT se salga del límite, se toma como solución una fracción de dicho límite. El valor de λ puede llegar a ser mayor de uno. No es necesario una precisión muy grande en el cálculo de λ , ya que al principio, cuando los valores de N_i están distantes de la solución es mas importante proseguir con la siguiente iteración que encontrar una λ muy exacta. Conforme se acercan los valores de N_i a la solución, el valor de λ tiende a ser uno. Hay que recordar que los balances de masa son satisfechos por cualquier valor de λ .

Existe otro punto a considerar en los cálculos y es que la cantidad de alguna especie en la fase multicomponente puede llegar a ser tan pequeña que el término logarítmico en la ecuación (3.44) cause problemas, además de que el número de iteraciones será muy grande. En este caso el componente en cuestión debe ser eliminado de la mezcla de reacción (sin dejar de considerar los efectos que como reactivo

inicial tenga en las B_j). Una manera de lograr esto es tomando un valor de λ que haga que la composición de este componente sea cero. Los resultados representarán adecuadamente la composición al equilibrio de las especies presentes en una cantidad significativa. Para estimar la composición de las especies que son trazas en la fase multicomponente, se considera que el número total de moles y el número de moles de cada especie al equilibrio no va a variar significativamente con la inclusión de estos componentes traza, y los multiplicadores de Lagrange permanecerán constantes. De este modo la energía de Gibbs está dada por

$$G/RT = F = \sum_{i=1}^c N_i (L_i + \ln(N_i/N)) + \sum_{i=c+1}^{c+a} N_i L_i + \sum_{j=1}^m \Pi_j (B_j - \sum_{i=1}^{c+a} A_{ji} N_i) \quad 3.49$$

Considerando que ahora las sumatorias incluyen también las especies trazas y que los multiplicadores de Lagrange permanecen constantes, se toman las derivadas parciales de la energía de Gibbs con respecto a las N_i y se igualan a cero, que es la condición de equilibrio, obteniéndose las siguientes relaciones:

para $1 \leq i \leq c$

$$(L_i + \ln(N_i/N)) + N_i \partial \ln N_i / \partial N_i + \sum_{j=1}^m \Pi_j (-\partial \ln N_i / \partial N_i) - \sum_{j=1}^m \Pi_j A_{ji} = 0$$

que se simplifica a

$$L_i + \ln(N_i/N) - \sum_{j=1}^m \Pi_j A_{ji} = 0$$

y despejando N_i se obtiene un valor aproximado de la composición del componente traza en la fase multicomponente:

$$N_i = N \exp \left(\sum_{j=1}^m \Pi_j A_{ji} - L_i \right) \quad 1 \leq i \leq c$$

En el apéndice 1 se muestra el cálculo de π_i utilizando ecuaciones de estado. En el apéndice 2 se presenta el procedimiento de cálculo de las cantidades g^i para la temperatura de reacción T . El algoritmo descrito en este capítulo se aplica en el desarrollo del programa

CAPITULO 4
Resolución por el método de las
constantes de equilibrio

En este método es necesario definir una serie de reacciones químicas independientes



En una reacción química las especies de la izquierda son llamadas reactivos y las especies químicas de la derecha son los productos, sin importar en que sentido realmente se lleve a cabo la reacción. En la reacción anterior A, B, L y M representan especies químicas y las cantidades a, b, l y m son llamadas coeficientes estequiométricos, números positivos que preceden las fórmulas químicas y que balancean la reacción

Una manera alternativa y mas general de expresar algebraicamente una reacción química es



donde ahora las A_i representan las fórmulas químicas. Las cantidades α_i son llamados números estequiométricos y son numéricamente iguales a los coeficientes estequiométricos. Sin embargo, se adopta la convención de signos que establece que α_i sea negativo para un reactivo y positivo para un producto.

Cuando una reacción como la anterior se lleva a cabo, los cambios en el número de moles de las especies están directamente relacionados con los números estequiométricos

$$\Delta N_{A1} / \Delta N_{A2} = \alpha_1 / \alpha_2 \quad 4.2$$

aplicando este principio a una cantidad diferencial obtenemos

$$dN_1 / dN_2 = \alpha_1 / \alpha_2 \quad 4.3$$

Si se define X como el grado de avance de la reacción, el cambio en el número de moles de cada una de las especies químicas se cuantifica mediante la siguiente relación:

$$dN_1/\alpha_1 = dN_2/\alpha_2 = dN_3/\alpha_3 = \dots = dX \quad 4.4$$

En el caso de existir simultáneamente más de una reacción en el sistema, se define una X_j para cada reacción. El cambio en el número de moles de cada especie estará dado por

$$dN_i = \sum_j \alpha_{ij} dX_j \quad 4.5$$

La energía de Gibbs total de un sistema multicomponente está dada por

$$dG = -SdT + VdP + \sum_i \mu_i dN_i \quad 4.6$$

Sustituyendo dN_i en función de dX por medio de la ecuación (4.5)

$$dG = -SdT + VdP + \sum_j \sum_i \mu_i \alpha_{ij} dX_j \quad 4.7$$

Como dG es una diferencial exacta, entonces se cumple que

$$\sum_j \mu_i \alpha_{ij} = (\partial G / \partial X_j)_{T,P}$$

La cantidad $\sum \mu_i \alpha_{ij}$ representa la variación de la energía de Gibbs con respecto al avance de reacción. El equilibrio se alcanza cuando la energía de Gibbs es mínima con respecto a todas las posibles variables. De este modo el criterio de equilibrio para cada una de las reacciones químicas es el siguiente

$$\sum_i \mu_i \alpha_{ij} = 0 \quad j=1,2,\dots,R \quad 4.8$$

donde R es el número de reacciones presentes en el sistema.

Substituyendo la ecuación que permite calcular el potencial químico

$$\mu_i = g_i + RT \ln a_i \quad 4.9$$

se tiene la siguiente ecuación

$$\sum_i \alpha_{ij} g_i + RT \sum_i \ln(a_i^{\alpha_{ij}}) = 0 \quad 4.10$$

o bien

$$\ln \Pi_i (a_i^{\alpha_{ij}}) = - \sum_i \alpha_{ij} g_i / RT \quad 4.11$$

Si se define el producto de las actividades a_i de todas las especies químicas elevadas a su número estequiométrico como la constante de equilibrio de una reacción K_j de la reacción j ésima

$$K_j = \prod_{i=1}^{c+r} a_i^{\alpha_{ij}} \quad 4.12$$

donde c es el número de especies químicas presentes en la fase multicomponente y r es el número de especies sólidas puras. Con esta definición tenemos

$$-RT \ln(K) = - \sum_{i=1}^{c+r} \alpha_{ij} g_i \quad 4.13$$

El término de la derecha de la ecuación se llama el cambio de energía de Gibbs de la reacción.

En la definición de K las actividades de las especies químicas están dadas por la relación entre la fugacidad a las condiciones de equilibrio, f_i y la fugacidad del estado de referencia, f_i° . El estado de referencia es arbitrario y puede ser diferente para cada especie química, lo que es indispensable es que para cada especie química el estado de referencia para el cálculo de f_i° sea el mismo que para g_i° .

Para un gas el estado de referencia se toma generalmente como el gas ideal puro a 1 atm. De esta forma $f_i^\circ = 1 \text{ atm}$ y $f_i = \phi_i y_i P$.

Para una fase sólida pura se puede tomar como referencia el sólido puro a 1 atm. Para el cálculo de la fugacidad del sólido a las condiciones de equilibrio se tiene que

$$d \ln(\phi_i) = V_i dP / RT \quad 4.14$$

Integrando desde 1 atm hasta P y considerando que para una fase condensada el volumen varía muy poco con la presión, tenemos

$$\phi_i / \phi_i^\circ = \exp(V_i(P-1) / RT) \quad 4.15$$

Excepto para presiones muy altas, este término se aproxima a uno y puede ser omitido de las expresiones de K .

De este modo la expresión de K_j para un sistema formado por una fase multicomponente en presencia de sólidos puros queda

$$K_j = \prod_{i=1}^{c+r} (P \phi_i y_i)^{-\alpha_{ij}}; \quad j=1, 2, \dots, R \quad 4.16$$

En esta ecuación se pueden agrupar dos términos

$$K_j = K_{yj} K_{\phi j} \quad 4.17$$

donde

$$K_{y_j} = \prod_{i=1}^{c+a} (P Y_i)^{-\alpha_{ij}} \quad 4.18$$

y

$$K_{p_j} = \prod_{i=1}^{c+a} (P_i)^{-\alpha_{ij}} \quad 4.19$$

este último término, en el caso de estar trabajando con gas ideal o una solución líquida ideal, es igual a uno.

Para encontrar la composición al equilibrio en este planteamiento, las variables a considerar son el grado avance de reacción X_j de las R reacciones, teniendo R ecuaciones de las constantes de equilibrio, una por cada reacción. Además se tienen las ecuaciones auxiliares de los balances de materia

$$N_i = N^0 - i + \sum_{j=1}^R \alpha_{ij} X_j; \quad i=1,2,\dots,c+a \quad 4.20$$

Los grados de avance de las reacciones estará limitado por la restricción de que el número de moles N_i de cada una de las especies químicas no podrá ser negativo en ningún momento.

Para calcular las composiciones (fracción mol) de cada compuesto químico en la fase multicomponente se procede de igual forma que en el capítulo anterior

$$N = \sum_{i=1}^c N_i$$
$$Y_i = N_i / N$$

donde N es el número de moles en la fase multicomponente.

El algoritmo de solución del problema de equilibrio planteado en este capítulo, se establece de la siguiente manera:

- 1.-Calcular las constantes de equilibrio K_j a la temperatura de reacción. (Apéndice 2)
- 2.-Ordenar las reacciones químicas en un orden descendiente del valor de K_j .
- 3.-Resolver una sola reacción química a la vez para obtener el grado

de avance de dicha reacción, considerando que no hay variación en las demás reacciones. Empezar en el orden escogido hasta terminar con todas las reacciones.

4.-Repetir el paso 3 hasta que entre dos vueltas consecutivas los grados de avance de las reacciones químicas cumplan con el criterio de convergencia.

Para la solución de una sola reacción química se propuso el siguiente procedimiento. Primero se escribe la ecuación de equilibrio de la reacción en la siguiente forma

$$(K/K\phi)_j = \prod_{\text{prod}} Y_i^{-\alpha_{ij}} / \prod_{\text{react}} Y_i^{-\alpha_{ij}} \quad 4.21$$

o bien, reorganizando la ecuación anterior

$$F(X_j) = \prod Y_i^{-\alpha_{ij}} - (K/K\phi)_j \prod Y_i^{-\alpha_{ij}} = 0. \quad 4.22$$

Posteriormente se establecen los límites que acotan la solución de X_j de modo que no se llegue a tener moles negativas de ninguna especie química.

Para encontrar el valor máximo de la X_j , se calculan para cada uno de los reactivos de la reacción, el grado de avance que se tendría si fueran reactivos limitantes

$$X_j' = 1 / -\alpha_{ij} [N + i + \sum_{k=1}^R \alpha_{ik} X_k] \quad 4.23$$

El menor de estos valores será el grado máximo de avance que se tendría sin agotar alguno de los reactivos

$$(X_j)_{\text{max}} = \min[X_j']$$

El valor mínimo se encuentra calculando para los productos

$$X_j'' = -1 / \alpha_{ij} [N + i + \sum_{k=1}^R \alpha_{ik} X_k] \quad 4.24$$

De estos valores se conserva el máximo

$$(X_j)_{\text{min}} = \max[X_j'']$$

Ahora aplicamos el método de Newton-Raphson para encontrar el valor de X_j que se encuentre entre los límites establecidos y que haga $F(X_j) = 0$. Como se sabe que

$$F[(X_i)_{\min}] < 0$$

$$F[(X_i)_{\max}] > 0$$

existe una raíz entre dichos límites. Para asegurar la convergencia, se mueven los límites conforme valoremos la función, para mantener siempre acotada la solución.

El cálculo de los coeficientes de fugacidad se hace con la composición que se tiene en el momento de comenzar la solución de cada una de las reacciones. Posteriormente, mientras se resuelve una reacción independientemente de las otras se considera que los coeficientes de fugacidad son constantes. Una vez terminado el cálculo de esta reacción se vuelven a calcular los coeficientes de fugacidad para la nueva composición y se resuelve la siguiente reacción.

Este algoritmo de cálculo se aplica al desarrollo del programa de cómputo presentado en el Apéndice 5.

CAPITULO 5

Aplicaciones

Reformacion de gas natural

El primer caso que se presenta es el ejemplo propuesto por Smith y Van Ness (14), en el cual se tiene una mezcla gaseosa reaccionante compuesta de metano, monóxido de carbono, dióxido de carbono, vapor de agua e hidrógeno a una temperatura de 1000 K y una presión de 1 atmósfera. Las reacciones que se presentan en este sistema son:



La composición inicial de la mezcla propuesta por Smith y van Ness es de 2 moles de CH_4 y 3 moles de H_2O . Esta composición es equivalente a tener 1.98 moles de CH_4 , 2.97 moles de H_2O , 0.07 moles de H_2 y 0.01 moles tanto de monóxido de carbono como de dióxido de carbono. La razón de usar esta última composición es que no se debe fijar como suposición inicial cero moles para algún componente, ya que en las subsecuentes iteraciones no aparecerán valores para los moles de este componente. Esta composición cumple con los mismos balances de masa que el ejemplo de Smith y Van Ness.

Para este mismo sistema se probó el ejemplo presentado por White y Seider (20), en el cual se tiene presente nitrógeno como una especie inerte. Las condiciones de operación son de 1067 K y 12.3 atmósferas.

En ambos casos se supone un comportamiento de gas ideal, debido a estar trabajando a temperaturas altas y presiones bajas. Para encontrar la composición del sistema en el equilibrio se utilizaron los dos métodos propuestos en los capítulos 3 y 4, con la intención de comparar su velocidad de convergencia, ver que se obtenga los mismos resultados y comparar éstos con los datos publicados.

Producción de Metanol

Otro ejemplo que fue probado es la producción de metanol a partir de

una mezcla de monóxido de carbono y dióxido de carbono con hidrógeno en proporciones estequiométricas. Para este caso se utilizó el método de minimización de la energía de Gibbs. Inicialmente se propuso que las especies presentes en el equilibrio fueran monóxido de carbono, dióxido de carbono, hidrógeno, vapor de agua, metanol y oxígeno. Las condiciones de reacción fueron una temperatura de 600 K y una presión de 330 atmósferas. Como en este caso se tiene condiciones en las cuales el comportamiento de los gases se aleja de la idealidad, se observó las diferencias entre la solución encontrada con la suposición de comportamiento ideal y la obtenida con el uso de ecuaciones de estado (Peng-Robinson). Además, como tenemos la presencia de compuestos polares (vapor de agua y metanol) se utilizó también la modificación de Stryjek y Vera a Peng-Robinson para especies químicas polares. La composición inicial de la mezcla gaseosa es de 1.8 moles de monóxido de carbono, 3.7 moles de hidrógeno y 0.1 moles para todas las especies restantes. La temperatura es de 1067 K y una presión de 12 atm.

Reducción del óxido de hierro

El caso de una fase reaccionante multicomponente con la presencia de sólidos puros se ejemplificó con la reducción de óxido de hierro para obtener hierro, en la presencia de coque. Inicialmente se resolvió para una presión de 1 atmósfera y una temperatura de 700 K, a la cual el óxido estable de hierro es el Fe_3O_4 . Posteriormente se probó a una presión de una atmósfera y una temperatura de 1100 K en la cual el óxido estable de hierro es el FeO . En ambos casos debido a la baja presión y altas temperaturas se supuso un comportamiento de gas ideal.

CAPITULO 6

Análisis de Resultados

Reformación de gas natural

En el caso propuesto por Smith y Van Ness, el método de minimización de la energía de Gibbs converge en 4 iteraciones, mientras el método de las constantes de equilibrio requiere de 19 iteraciones. En el apéndice 4 se muestran los resultados para todos los ejemplos de este trabajo. La solución que se obtiene por ambos métodos es la misma y se compara con la obtenida por Smith y Van Ness en la siguiente tabla:

Smith y Van Ness		Minimización.
0.0199	yCH ₄	0.02363
0.0995	yH ₂ O	0.1034
0.1753	yCO	0.1725
0.0359	yCO ₂	0.0365
0.6694	yH ₂ O	0.6638

Al comparar los resultados se observa que hay una gran similitud. Las diferencias no se deben a un error numérico o a la necesidad de una convergencia mayor, ya que los resultados obtenidos por los dos métodos son muy similares. Las diferencias existentes se deben a que los valores de las energías de Gibbs de cada compuesto, g^i , a la temperatura del sistema utilizadas por Smith y Van Ness son diferentes a las calculadas en el programa. La validez del valor que se obtenga va a depender de la precisión que se tenga en el C_p , ya que se están calculando integrales que van de los 298 K a los 1000 K. Los valores de la energía de Gibbs entre la temperatura y la constante universal de los gases son:

Smith y Van Ness		Minimización
2.32	CH ₄	3.22
-23.16	H ₂ O	-23.73
-24.12	CO	-23.53

El caso propuesto por White y Seider es similar al anterior, la diferencia está en que se tiene la presencia de un elemento inerte. En este caso el método de minimización converge en 4 iteraciones, mientras el método de las constantes de equilibrio requiere de 22 iteraciones. Las soluciones obtenidas son las siguientes:

White y Seider		Minimización.
0.875	nCH ₄	1.059.
61.821	nH ₂ O	62.057.
0.59	nN ₂	0.59
50.980	nH ₂	50.374.
6.282	nCO	6.158
7.983	nCO ₂	7.942

Las diferencias se siguen debiendo a los valores que se tienen de la energía de Gibbs para cada compuesto.

En el método de las constantes de equilibrio el número de iteraciones aumenta de un ejemplo a otro. Esto es lógico ya que el valor de las constantes aumenta, como se muestra a continuación:

Ejemplo de Smith y Van Ness		Ejemplo de White y Seider
20.53	K ₁	107.77
27.90	K ₂	112.90

El número de iteraciones del método de las constantes de equilibrio es mucho más grande que el de minimización. Esto se debe a que en el primero estamos usando un método que es muy sensible a la suposición inicial. En ambos métodos se dio la misma composición inicial. Otro punto a considerar para la diferencia en el número de iteraciones, es que en el método de minimización se cuenta con el parámetro λ que permite acelerar la convergencia. Un valor de λ mayor de la unidad acelera la convergencia. Un valor de λ entre cero y uno amortigua la convergencia. Los valores de λ durante la convergencia de los dos

ejemplos probados son los siguientes:

Ejemplo Smith y van Ness

1.582	Iteración 1
0.955	Iteración 2
1.005	Iteración 3

Ejemplo White y Seider

5.535
0.992
0.960

Para el segundo ejemplo en el cual las diferencias de la energía de Gibbs son mayores (mayores valores de las K) los valores iniciales de λ son mayores permitiendo una rápida convergencia.

Producción de Metanol

El segundo ejemplo es el caso de la producción de metanol a partir de monóxido de carbono e hidrógeno. Para este caso se consideró una alimentación de monóxido de carbono con hidrógeno en proporciones estequiométricas. La temperatura fue de 600 K y una presión de 330 atmósferas. El primer caso se hizo suponiendo que al equilibrio tenemos presentes monóxido de carbono, dióxido de carbono, hidrógeno, vapor de agua, metanol y oxígeno. Para este ejemplo se utilizó únicamente el método de la minimización de la energía de Gibbs. Después de 12 iteraciones el sistema no había convergido. En la tabla siguiente se muestran valores de λ en algunas iteraciones:

Iteración	λ
1	0.0157
2	0.0085
3	0.0067
4	0.0060
5	0.0057
6	0.0055
10	0.0055

Si se observa el valor de λ se ve que este es pequeño y va disminuyendo de iteración en iteración, debido a que los moles de

oxígeno que tienden a cero, están limitando el valor máximo posible de λ . De este modo no es posible llegar a la convergencia deseada.

Se volvió a calcular el equilibrio quitando de la mezcla de reacción el oxígeno. Después de 5 iteraciones se logra la convergencia. Se tienen los valores de los multiplicadores de Lagrange y se puede calcular una aproximación del número de moles del oxígeno en el equilibrio, considerando que es un componente traza. Esta composición es del orden de $2 \cdot 10^{-41}$, con lo cual se ve que el oxígeno sí es un componente traza.

Como estamos trabajando a altas presiones, la suposición de gas ideal no es válida. Por esto se hicieron los cálculos usando la ecuación de Peng-Robinson para calcular los coeficientes de fugacidad. El número de iteraciones se incrementa de 5 a 9. Si se observa los valores de λ a partir de la iteración 4, se ve que los valores de λ son cercanos a uno. Además las variaciones que se tienen en los números no son muy grandes. Esto se explica por que en estas últimas iteraciones las variaciones se están dando en los coeficientes de fugacidad y hasta que no se estabilizan, no se puede lograr la convergencia deseada. Lo que ocurre es que estamos variando los coeficientes de la función sobre la cual buscamos convergencia, y para lograr esta convergencia es necesario tener un buen valor de la función y de su derivada. Esto es consecuencia de haber considerado que los coeficientes de fugacidad eran constantes al evaluar las derivadas durante cada iteración. Esta suposición ocasiona que el número de iteraciones sea mayor, pero si se relaja esta suposición el número de cálculos en cada iteración sería mucho más grande. Los resultados obtenidos considerando gas real en lugar de gas ideal son diferentes como se aprecia en la siguiente tabla:

gas ideal		gas real
0.5665	nCO	0.3159

0.1822	nCO ₂	0.1681.
1.4797	nH ₂	0.9362.
0.0177	nH ₂ O	0.0318
1.2512	nCH ₄ O	1.5159.

Se observa que tenemos un aumento en la conversión, es decir un menor número de moles de hidrógeno y un mayor número de moles de metanol, cuando consideramos que es gas real. Esto se debe a que el coeficiente de fugacidad para el hidrógeno es mayor de uno (1.32), mientras que para el metanol el coeficiente de fugacidad es menor de uno (0.60).

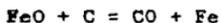
Como tenemos especies polares presentes como el vapor de agua y el metanol, también se usó la modificación de Stryjek y Vera a Peng-Robinson. Los resultados se obtuvieron después de 9 iteraciones. No existe ninguna diferencia significativa en el número de moles de las especies presentes, con respecto a los valores obtenidos anteriormente.

Reducción de hierro

El último ejemplo que se propuso fue la reducción del óxido de hierro en presencia de coque, para obtener hierro. Con este ejemplo se probó el caso en el cual se tiene una fase multicomponente y la presencia de fases sólidas puras. El primer caso que se considera es a una temperatura de 600 K y a presión atmosférica. A esta temperatura el óxido de hierro estable es el Fe₃O₄. Se propuso que las especies presentes al equilibrio son monóxido de carbono, dióxido de carbono, hidrógeno, vapor de agua, carbono, hierro y el óxido de hierro (Fe₃O₄). La solución se obtuvo después de 15 iteraciones. El número de iteraciones es alto debido a que al principio el valor de λ se ve limitado por los valores de moles negativos del monóxido de carbono y el dióxido de carbono.

Luego se probó un caso en que la temperatura sea de 1100 K. A esta

temperatura el óxido de fierro estable es el FeO. Después de 8 iteraciones no se ha encontrado una solución y se detienen los cálculos debido a que el valor de λ es cada vez menor (3×10^{-4}) lo que nos indica que no va a converger el método. Si se observan los resultados se ve que las especies que están limitando el valor de λ son dos especies sólidas, C y FeO. En estos casos lo más probable es que alguna de las fases sólidas no esté presente en el equilibrio. El siguiente análisis permite determinar cual de las fases sólidas es posible que desaparezca, y por lo tanto, no se debe incluir en el sistema. Si se observa que la reacción



tiene una constante de equilibrio igual a

$$K = a_{\text{FeO}} a_{\text{CO}} / a_{\text{Fe}} a_{\text{C}}$$

y como

$$a_{\text{FeO}} = a_{\text{C}} = a_{\text{Fe}} = 1$$

$$K = a_{\text{CO}}$$

Como la presión es de una atmósfera y se tiene altas temperaturas, se puede considerar comportamiento de gas ideal y la expresión de K queda

$$K = Y_{\text{CO}}$$

El valor de K es de 16.77, y como no podemos tener nunca una fracción mol mayor de uno, no se puede alcanzar el equilibrio. Es decir que por más óxido de fierro, FeO, y carbono se tenga el equilibrio estará totalmente desplazado a la derecha.

Se repiten los cálculos considerando que no tenemos óxido de fierro presente en el equilibrio. Para esto mantenemos el balance de materia, distribuyendo los moles iniciales de FeO en los otros compuestos. Para este caso se obtiene la solución después de 4 iteraciones y se tiene carbono presente al equilibrio.

CAPITULO 7

Conclusiones

El objetivo de la tesis consistió en elaborar un programa de cómputo que permita calcular el estado de equilibrio en un sistema multicomponente con la presencia de varias fases sólidas puras.

El programa de cómputo desarrollado cumple con este objetivo y permite el cálculo del estado de equilibrio en un sistema con varias reacciones químicas.

Al hacer una comparación entre los dos métodos planteados (minimización de la energía de Gibbs y constantes de equilibrio) se observa que si bien el método de las constantes de equilibrio es un método más sencillo de plantear, tiene una serie de desventajas con respecto al método de la minimización de la energía de Gibbs. Una de estas desventajas es la necesidad de plantear un sistema de reacciones independientes que nos representen a la mezcla reaccionante. Además el método de las constantes de equilibrio requiere de un mayor número de iteraciones y tiempo de cómputo para lograr la misma convergencia que el método de minimización de la energía de Gibbs. Una ventaja adicional de este último método es que fácilmente puede ser extendido para sistemas de varias fases multicomponentes (Apéndice 3).

Los programas de esta tesis pueden ser utilizados como apoyo a los cursos de termodinámica. En este caso los problemas que se le planteen a los estudiantes deberán contener la toma de decisiones sobre que compuestos están presentes en el equilibrio, como fueron los casos del oxígeno en la producción de metanol y del FeO en la reducción del óxido de hierro a 1100 K presentados en esta tesis.

Entre los aspectos que faltan por desarrollar está el hecho de que no se contó con parámetros de interacción binarios para el cálculo de los coeficientes de fugacidad ni se incluyen métodos para el cálculo de coeficientes de actividad para fases líquidas. Además se puede

mejorar el programa incluyendo ciertas tomas de decisiones sobre los compuestos presentes en el equilibrio. Esto tiene la ventaja de hacer el programa mas eficiente, aunque con fines didácticos es preferible que sea el estudiante el que tome la decisiones.

APENDICE 1

Cálculo de fugacidades por medio de ecuaciones de estado

Una ecuación de estado nos sirve para calcular la variación de las propiedades termodinámicas con respecto a los valores obtenidos para el gas ideal. En el caso de la fugacidad tenemos que:

$$\text{si } v=v(T,P) \quad \ln(f_i/PX_i) = \int_1^P [\frac{v_i}{RT} - 1/P] dP \quad \text{c.1}$$

$$\text{si } P=P(T,v) \quad RT \ln(f_i/PX_i) = -\ln(PV/RT) + \int_v^\infty [\frac{\partial P}{\partial N_i} - RT/V] dV \quad \text{c.2}$$

donde definimos

$$\phi_i = f_i/PX_i \quad \text{c.3}$$

siendo ϕ_i el coeficiente fugacidad del componente i .

La integral se calcula sustituyendo la ecuación de estado que se aplique. Para el caso de gas ideal el coeficiente de fugacidad es igual a uno.

Las ecuaciones de Soave y la de Peng-Robinson pertenecen a las ecuaciones de estado cúbicas.

$$P = RT/(v-b) - a/(v^2 + uv + wv^2) \quad \text{c.4}$$

siendo para Soave $u=1$ y $w=0$, y para Peng-Robinson $u=2$ y $w=1$.

El factor de compresibilidad Z que corrige la ecuación de gas ideal es:

$$Z = Pv/RT \quad \text{c.5}$$

y se puede obtener de la ecuación (c.4) por manipulación algebraica, llegando a una ecuación cúbica en Z :

$$Z^3 + (uB - B - 1)Z^2 + (A + wB - uB^2 - uB)Z + (-wB^3 - wB^2 - AB) = 0 \quad \text{c.6}$$

donde

$$A = aP/(RT)^2 \quad \text{c.7}$$

$$B = bP/RT \quad \text{c.8}$$

Los parámetros a y b están relacionados con propiedades termodinámicas de las sustancias puras de modo que:

$$b = d_1 RT_c / P_c \quad \text{c.9}$$

$$a = d_2 (RT_c)^2 \alpha / P_c \quad \text{c.10}$$

siendo

$$\alpha = [1 + m(1 - (T/T_c)^{0.5})]^2 \quad \text{c.11}$$

$$m = d_3 + d_4 \omega + d_5 \omega^2 \quad \text{c.12}$$

donde ω es el factor acéntrico de Pitzer, R es la constante universal de los gases y T_c , P_c son las propiedades críticas de la sustancia pura. Los valores de las constantes d_1, \dots, d_5 son:

Soave		Peng-Robinson
0.08664	d_1	0.0778
0.42748	d_2	0.45724
0.480	d_3	0.36464
1.574	d_4	1.54226
-0.176	d_5	-0.26992

Para el caso de componentes polares existe la modificación de Mathias a la ecuación de Soave. Esta modificación consiste en agregar otro parámetro q que interviene en el cálculo de α

$$\text{para } T < T_c \quad \alpha = [1 + m(1 - (T/T_c)^{0.5} - q(1 - T/T_c)(0.7 - t/T_c))]^2 \quad \text{c.13}$$

$$\text{para } T > T_c \quad \alpha^{0.5} = \exp\{[c - 1](1 - (T/T_c)^c)/c\} \quad \text{c.14}$$

siendo

$$c = 1 + m/2 + 0.3q \quad \text{c.15}$$

Los parámetros reportados por Mathias fueron obtenidos utilizando diferentes valores de d_3 , d_4 y d_5 , siendo estos

$$d_3 = 0.48508 \quad d_4 = 1.55191 \quad d_5 = -0.15613$$

Para la ecuación de Peng-Robinson, Stryjek y Vera plantean una modificación similar para componentes polares. Esta modificación es

basicamente la misma que la de Mathias, salvo que en este caso se amplía la expresión de $m=f(\omega)$:

$$\alpha = [1 + k_1 - Tr \cdot 0.5]^{-2} \quad \text{c.16}$$

siendo

$$k = k_0 + k_1(1 + Tr \cdot 0.5)(0.7 - Tr) \quad \text{c.17}$$

$$k_0 = d_1 + d_2 \omega + d_3 \omega^2 + d_4 \omega^3 \quad \text{c.18}$$

y los valores de las constantes d_1, \dots, d_4 son:

$$d_1 = 0.378893 \quad d_2 = 1.4897153$$

$$d_3 = -0.1731848 \quad d_4 = 0.0196554$$

Estas relaciones pueden usarse par el caso de mezclas si definimos alguna regla de mezclado. La regla de mezclado usada para b es:

$$b = \sum_i X_i b_i \quad \text{c.19}$$

Para el caso de a existen diferentes reglas de mezclado. Estas dependen de los parametros de interacción binarios (K_{ij}) que se tienen. Soave propone la siguiente regla de mezclado:

$$a = \sum_i \sum_j X_i X_j (a_i a_j)^{-0.5} (1 - K_{ij}) \quad \text{c.20}$$

donde $K_{ii} = 0$ y $K_{ij} = K_{ji}$

También se proponen reglas de mezclado del tipo de Van Laar o de Margules, donde:

$$a = \sum_i \sum_j X_i X_j a_{ij} \quad \text{c.21}$$

siendo a_{ij} para Margules

$$a_{ij} = (a_i a_j)^{-0.5} (1 - X_i K_{ij} - X_j K_{ji}) \quad \text{c.22}$$

y para Van Laar

$$a_{ij} = (a_i a_j)^{-0.5} (1 - K_{ij} K_{ji} / (X_i K_{ij} + X_j K_{ji})) \quad \text{c.23}$$

ambas expresiones se reducen a la inicial cuando $K_{ij} = K_{ji}$

Usando la ecuación (c.2) para el coeficiente de fugacidad y la ecuación de estado (c.4) con la regla de mezclado usada, se tiene que:

$$\ln p_i = -\ln(Z-B) + (Z-1)(\partial b_{Ni}/\partial N_i)/b +$$

$$[(\partial a_{Ni}^{-2}/\partial N_i)/b N_i RT - (\partial b_{Ni}/\partial N_i)a/b^{-2} RT] A \quad c.24$$

donde

$$A = 1/(u^2 - 4w)^{0.5} \ln((Z+B(u^2 - 4w)^{0.5}/2)/(Z+B(u^2 - 4w)^{0.5}/2))$$

c.25

$$\partial b_{Ni}/\partial N_i = b_i \quad c.26$$

para una regla de mezclado con la de la ecuación (c.20)

$$\partial a_{Ni}^{-2}/\partial N_i = 2 \sum_j X_j (a_{ij})^{-0.5} (1 - K_{ij}) \quad c.27$$

si la regla de mezclado es como Margules (c.22)

$$\partial a_{Ni}^{-2}/\partial N_i = 2 \sum_j X_j a_{ij} + 2 \sum_{j \neq i} X_i X_j (a_{ij})^{-0.5} [X_i K_{ij} + X_j K_{ji} - K_{ij}] +$$

$$2 \sum_{j \neq i} \sum_{m \neq j, m \neq i} X_j X_m (a_{jam})^{-0.5} (X_i K_{jm} + X_m K_{mj}) \quad c.28$$

para el caso de van Laar

$$\partial a_{Ni}^{-2}/\partial N_i = 2 \sum_j X_j a_{ij} - 2 \sum_{j \neq i} X_i X_j (a_{ij})^{-0.5} K_{ij} K_{ji} +$$

$$(X_i K_{ij} + X_j K_{ji} - K_{ij}) / (X_i K_{ij} + X_j K_{ji})^{-2} +$$

$$2 \sum_{j \neq i} \sum_{m \neq j, m \neq i} X_j X_m (a_{jam})^{-0.5} K_{jm} K_{mj} / (X_j K_{jm} + X_m K_{mj}) \quad c.29$$

Una de las ventajas de este tipo de ecuaciones de estado, es que se pueden obtener valores de coeficientes de fugacidad para fases líquidas.

APENDICE 2

Cálculo de g

La energía de Gibbs g del componente puro debe ser calculada a la temperatura T del sistema reaccionante y a la presión de 1 atm.

Para su evaluación se parte de la energía de Gibbs de formación del componente puro, Ag^*f , y de la entalpía de formación, Δh^*f , a la temperatura de referencia $T_0=298.15^\circ\text{K}$ y a la presión de 1 atm. Estas cantidades se encuentran reportadas en numerosas tablas de propiedades, como el Apéndice del libro de Walas (19) o en el libro de Reid, Frausnitz y Sherwood (12).

La variación con la temperatura de la energía de Gibbs está dada por:

$$g(T)/T = Ag^*f/T_0 + \Delta h^*f(1/T - 1/T_0) + (\Delta h - T\Delta s)/T$$

donde el primer término de la ecuación es la energía de Gibbs de formación de la sustancia, el segundo término nos da la variación de la energía de Gibbs con la temperatura cuando se considera que el Δh^*f permanece constante con la temperatura, y el último término nos da la corrección debida al C_p . El Δh y el Δs están dados en términos del C_p por las siguientes ecuaciones:

$$\Delta h = \int_{T_0}^T C_p dT$$

$$\Delta s = \int_{T_0}^T C_p dT/T$$

donde T^* es la temperatura a la cual están reportadas Ag^*f y Δh^*f , y T es la temperatura a la cual se quiere g . Entre las formas más usadas del C_p están funciones del tipo

$$C_p = a + bT + cT^2 + dT^3 + e/T^2.$$

sustituyendo esta ecuación del C_p en las ecuaciones (b.17) y (b.18), e integrando se tiene que:

$$\Delta h = a(T - T^*) + b(T^2 - T^{*2})/2 + c(T^3 - T^{*3})/3 + d(T^4 - T^{*4})/4 - e(1/T - 1/T^*)$$

$$\Delta s = a \ln(T/T^*) + b(T - T^*) + c(T^2 - T^{*2})/2 + d(T^3 - T^{*3})/3 - e(1/T^2 - 1/T^{*2})/2$$

En los programas se manejan dos formas equivalentes de g^i : 1) es calculando la energía de Gibbs de la sustancia a la temperatura deseada como se acaba de describir y 2) es calculando una energía de formación de la sustancia a la temperatura deseada. Esta última forma se utilizó con el fin de comparar con datos reportados. La única variación en el cálculo es que en vez de usar el C_p de la sustancia se usa un ΔC_p que es:

$$\Delta C_p = \Delta a + \Delta b T + \Delta c T^2 + \Delta d T^3 + \Delta e / T^2$$

donde los coeficientes están en función de los coeficientes del C_p de la sustancia pura y los C_p de los elementos que forman la sustancia, de la siguiente forma:

$$\Delta a = a^* - \sum_{j=1}^e \nu_j a_j$$

donde e es el número de elementos puros que forman a la sustancia, ν_j es el coeficiente estequiométrico de los elementos en la reacción de formación de la sustancia, a_j es el coeficiente a del C_p del elemento j y a^* es el coeficiente a del C_p de la sustancia

APENDICE 3

Extensión del método de minimización

de la energía libre de Gibbs para multifases

En el caso de tener un sistema con varias fases multicomponentes presentes en el equilibrio, la energía de Gibbs está dada por

$$F = G/RT = \sum_{i=1}^S g^{\circ} N_i + \sum_{l=1}^P \sum_{i=s+1}^C N_{li} (g^{\circ} - \mu_i/RT + \ln a_{li}) \quad c.1$$

Donde S es el número de especies sólidas puras presentes, C es el número total de especies químicas presentes en la mezcla, P es el número de fases presentes en cuales se encuentran distribuidas las especies, a_{li} es la actividad del componente i en la fase l y generalmente se expresa:

para gases como $a_{li} = p_{li}/P$

para fases líquidas como $a_{li} = \gamma_{li} X_{li}/f_i$

definimos

$$B_i = \sum_{l=1}^P A_{li} N_l + \sum_{l=1}^P \sum_{i=s+1}^C A_{li} N_{li} \quad c.2$$

donde A_{ij} es el número de átomos del elemento j en la especie química i.

Se considera que se tiene una aproximación inicial, y se obtiene la siguiente aproximación expandiendo la función de energía de Gibbs por series de Taylor

$$F \approx F + \sum_{j=1}^{k+1} (N_j - N_j^0) \partial G / \partial N_j + \sum_{l=1}^P \sum_{j=s+1}^C (N_{lj} - N_{lj}^0) \partial G / \partial N_{lj} \quad c.3$$

$$+ \frac{1}{2} \sum_{j=1}^S (N_j - N_j^0)^2 \partial^2 G / \partial N_j^2 + \frac{1}{2} \sum_{l=1}^P \sum_{j=1}^C \sum_{j'=1}^C (N_{lj} - N_{lj}^0)(N_{lj'} - N_{lj'}^0) \partial^2 G / \partial N_{lj} \partial N_{lj'}$$

Se introducen los balances de materia a la función por medio de los multiplicadores de Lagrange y se tiene

$$Q = F + \sum_{k=1}^{k+1} \lambda_k [B_k - \sum_{j=1}^S A_{kj} N_j - \sum_{l=1}^P \sum_{j=s+1}^C A_{kl} N_{lj}] \quad c.4$$

Esta función se minimiza igualando a cero las derivadas parciales de la función con respecto al número de moles de cada especie en la

iteración $k+1$

$$\frac{\partial Q}{\partial N_i} = \frac{\partial Q}{\partial N_i} = 0 \quad / \quad \frac{\partial N_i}{\partial N_i} = 0 \quad \text{c.5}$$

Estas ecuaciones, junto con los balances de masa (c.2) y las relaciones.

$$N_i = \sum_{l=1}^c N_{il}; \quad l=1,2,\dots,P \quad \text{c.6}$$

Se tiene un sistema de $P+S+E+P*(C-S)$ ecuaciones en donde las incógnitas son los S número de moles de las especies sólidas, los E multiplicadores de Lagrange, los P número de moles totales de cada fase y el número de moles de cada especie en cada fase. Por un procedimiento similar al del capítulo 3 para una fase multicomponente, se reduce el sistema a un sistema de $S+P+E$ ecuaciones. Las incógnitas en este sistema serán los moles de las C especies sólidas puras, los E multiplicadores de Lagrange y los P moles de cada fase. Los moles para cada especie química en cada fase para la nueva aproximación, son calculadas en base al número total de moles de la fase

$$N_{il} = N_{il} \left[g^* + \frac{u_i}{RT} + \ln a_i + N_i \frac{k_{i+1}}{N_i + \sum_{k=1}^m N_{ik}} \right] \quad \text{c.7}$$

El sistema de ecuaciones se obtiene considerando que las derivadas de los coeficientes de fugacidad y las de los coeficientes de actividad con respecto a la composición son iguales a cero. En el diagrama se ve el sistema resultante.

Del mismo modo que se definió una λ que nos indica la mejor longitud de cambio, en el caso de una fase multicomponente, es posible definirla para varias fases multicomponentes. Este parámetro sigue siendo el mayor valor que cumpla con los criterios de (1) no tener moles negativos de ninguna especie en ninguna fase y (2) que la derivada $dG_t/d\lambda$ no se vuelva positiva, es decir no nos pasemos del mínimo. Esta derivada estará dada como:

$$dG/d\lambda = \sum_{l=1}^P G^* \Delta_l / RT + \sum_{l=1}^P \sum_{i=1}^c \sum_{k=1}^m G_{ik} \Delta_{ik} \quad \text{c.8}$$

En el caso de tener componentes traza, es decir que su composición

sea muy pequeña, los términos logarítmicos pueden causar problemas. En tal caso es necesario retirar este componente de la mezcla (manteniendo su efecto sobre el valor de las B_j) y su composición en el equilibrio estará dada por

$$N_{i1} = N_i \exp(-g^*i/RT + \ln \beta_{i1} + \sum_{j=1}^m \pi_j A_{ik}) \quad \text{c.9}$$

donde para líquidos

$$\beta_{i1} = \gamma_{i1} / f^*_{i1}$$

y el coeficiente de actividad es considerando dilución infinita.

Cuando se postula incorrectamente fases sólidas puras al equilibrio, no se puede obtener una solución. Esto es debido a que el valor de λ será pequeño. Un modo de saber si se debe incluir o no una fase sólida pura, es realizar los cálculos sin la presencia de fases sólidas puras. De este modo se obtienen los valores de los multiplicadores de Lagrange y se aplica el siguiente criterio a las fases sólidas:

$$g^*i/RT - \sum_{j=1}^m \pi_j A_{ik} < 0 \quad \text{c.10}$$

Si la especie sólida cumple con este criterio puede ser incluida en los cálculos.

Matriz de Coeficientes

	$N_i(k+1)/N_i$	N_j	π_m	Lado derecho ecuacion
$i=1 \leq P$		$i=1 \leq S$	$i=1 \leq E$	
$i=1 \leq E$	$\sum_{l=1}^C A_{il} N_{il}$	A_{jm}	R_{mm}	$B_m + \sum_{l=1}^C A_{il} N_{il} (G_{il}/RT)$
$i=1 \leq P$	0	0	$\sum_{l=1}^E A_{il} N_{il}$	$\sum_{l=1}^E N_{il} (G_{il}/RT)$
$i=1 \leq S$	0	0	A_{jm}	g_{0j}/RT

Donde P es el numero de fases presente, S el numero de fases solidos puras, E el numero de elementos quimicos, A_{ij} es el numero de atomos del elemento j en la especie quimica i, N_{il} es el numero de moles de la especie i en la fase l en la iteracion k, N_j es el numero de moles de la fase solida pura j, m son los multiplicadores de Lagrange y g_{0j} es la energia libre de Gibbs de la fase solida j.

B_m esta definido como

$$B_m = \sum_{l=1}^S A_{il} N_{il} + \sum_{i=1}^E A_{il} N_{il}$$

G_{il} es la energia libre de Gibbs de la especie i en la fase l.

R_{mm} esta definido como

$$R_{mm} = \sum_{m_1, m_2} \sum_{i=1}^E A_{im_1} A_{im_2} N_{il}$$

APENDICE 4

Resultados

A continuación se presentan los resultados de los ejemplos realizados. En cada ejemplo se dan al principio las condiciones de reacción.

Para los ejemplos en los que se emplea el método de las constantes de equilibrio, se dan los valores de los grados de avance de reacción para cada reacción en cada iteración.

Cuando se utiliza el método de la minimización de la energía de Gibbs se dan los valores del número de moles (N_i) en cada iteración.

También se dan los valores que se obtienen para la nueva suposición (Nueva). Por último se da el valor que se toma de λ para actualizar los números de moles de las diferentes especies químicas y así obtener el número de moles (N_i) de la siguiente iteración. Cuando se usa alguna ecuación de estado se dan los valores que se obtienen de los coeficientes de fugacidad en cada iteración.

REFORMACION DE GAS NATURAL
 EJEMPLO DE SMITH Y VAN NISS
 Calculo del equilibrio quimico por medio de constantes de equilibrio
 Temperatura= 1.0000000000E+03
 Presion= 1.0000000000E+00
 Moles iniciales de CO 1.0000000000E-02
 Moles iniciales de CO2 1.0000000000E-02
 Moles iniciales de H2 7.0000000000E-02
 Moles iniciales de H2O 2.9700000000E+00
 Moles iniciales de CH4 1.9800000000E+00
 Equilibrio Reaccion 1 20.52
 Equilibrio Reaccion 2 27.87
 Iteracion 1
 X reaccion 1= 4.6255552909E-01
 X reaccion 2= 1.1512204124E+00
 Iteracion 2
 X reaccion 1= 7.9416338016E-01
 X reaccion 2= 8.8710663450E-01
 Iteracion 3
 X reaccion 1= 1.0204589064E+00
 X reaccion 2= 6.9745187240E-01
 Iteracion 4
 X reaccion 1= 1.1723050317E+00
 X reaccion 2= 5.6676225672E-01
 Iteracion 5
 X reaccion 1= 1.2727380907E+00
 X reaccion 2= 4.7899193865E-01
 Iteracion 6
 X reaccion 1= 1.3402411569E+00
 X reaccion 2= 4.2065821381E-01
 Iteracion 7
 X reaccion 1= 1.3848923830E+00
 X reaccion 2= 3.8129995898E-01
 Iteracion 8
 X reaccion 1= 1.4144241565E+00
 X reaccion 2= 3.5518942443E-01
 Iteracion 9
 X reaccion 1= 1.4339595255E+00
 X reaccion 2= 3.3788356031E-01
 Iteracion 10
 X reaccion 1= 1.4468852973E+00
 X reaccion 2= 3.2641853958E-01
 Iteracion 11
 X reaccion 1= 1.4554395802E+00
 X reaccion 2= 3.1882475353E-01
 Iteracion 12
 X reaccion 1= 1.4611017489E+00
 X reaccion 2= 3.1379564362E-01
 Iteracion 13
 X reaccion 1= 1.4648500513E+00
 X reaccion 2= 3.1046524243E-01
 Iteracion 14
 X reaccion 1= 1.4673316039E+00
 X reaccion 2= 3.0825984592E-01
 Iteracion 15
 X reaccion 1= 1.4689746053E+00
 X reaccion 2= 3.0679945888E-01

Iteracion 16
X reaccion 1= 1.4700624567E+00
X reaccion 2= 3.0583242019E-01
Iteracion 17
X reaccion 1= 1.4707827563E+00
X reaccion 2= 3.0519207142E-01
Iteracion 18
X reaccion 1= 1.4712596967E+00
X reaccion 2= 3.0476805089E-01
Iteracion 19
X reaccion 1= 1.4715755029E+00
X reaccion 2= 3.0448727734E-01

R E S U L T A D O S

Iteraciones realizadas = 19

Reaccion 1: Xanterior = 1.4712596967E+00 Xnueva = 1.4715755029E+00

Reaccion 2: Xanterior = 3.0476805089E-01 Xnueva = 3.0448727734E-01

Composiciones finales :

Moles de CO 1.4815755029E+00

Moles de CO2 3.1448727734E-01

Moles de H2 5.7026756181E+00

Moles de H2O 8.8944994241E-01

Moles de CH4 2.0393721975E-01

Y (CO) = 1.7243410754E-01
Y (CO2) = 3.6601801862E-02
Y (H2) = 6.6370953007E-01
Y (H2O) = 1.0351919745E-01
Y (CH4) = 2.3735363075E-02

REFORMACION DE GAS NATURAL
EJEMPLO DE SMITH Y VAN NESS

Calculo de la composicion al equilibrio por medio del metodo de la minimizacion de la energia de Gibbs

Moles iniciales de CO= 1.0000000000E-02
Moles iniciales de CO2= 1.0000000000E-02
Moles iniciales de H2= 7.0000000000E-02
Moles iniciales de H2O= 2.9700000000E+00
Moles iniciales de CH4= 1.9800000000E+00
Temperatura (K) 1.0000000000E+03
Presion (atm) 1.0000000000E+00

Iteracion 1

NI CO=	1.0000000000E-02	Nueva=	6.5783687204E-02
NI CO2=	1.0000000000E-02	Nueva=	5.5765313045E-02
NI H2=	7.0000000000E-02	Nueva=	4.2041231361E-01
NI H2O=	2.9700000000E+00	Nueva=	2.6226856870E+00
NI CH4=	1.9800000000E+00	Nueva=	1.8784509998E+00
LAMBDA	1.5821548734E+01		

Iteracion 2

NI CO=	8.9258432565E-01	Nueva=	1.51267865204E+00
NI CO2=	7.3407813067E-01	Nueva=	2.9627604683E-01
NI H2=	5.6140654968E+00	Nueva=	5.7231397492E+00
NI H2O=	6.3925941816E-01	Nueva=	8.9476938561E-01
NI CH4=	3.7333754416E-01	Nueva=	1.9104543262E-01
LAMBDA	9.5538217853E-01		

Iteracion 3

NI CO=	1.4850112690E+00	Nueva=	1.4830006976E+00
NI CO2=	3.1580982161E-01	Nueva=	3.1387947507E-01
NI H2=	5.7182730938E+00	Nueva=	5.7045199928E+00
NI H2O=	8.8336908775E-01	Nueva=	8.8924035255E-01
NI CH4=	1.9917890933E-01	Nueva=	2.0311982736E-01
LAMBDA	1.0052968844E+00		

ITERACIONES= 4

Nnueva	CO=	1.4829688883E+00	Nant=	1.4829900478E+00
Nnueva	CO2=	3.1388433070E-01	Nant=	3.1386925025E-01
Nnueva	H2=	5.7044439875E+00	Nant=	5.7044471442E+00
Nnueva	H2O=	8.8926245048E-01	Nant=	8.8927145196E-01
Nnueva	CH4=	2.0314678104E-01	Nant=	2.0314070185E-01
Y(CO)=		1.7256452719E-01		
Y(CO2)=		3.6524907264E-02		
Y(H2)=		6.6379321061E-01		
Y(H2O)=		1.0347833695E-01		
Y(CH4)=		2.3639017984E-02		
G/RT de	CO	-2.3534810608E+01		
G/RT de	CO2	-4.7573907398E+01		
G/RT de	H2	0.0000000000E+00		
G/RT de	H2O	-2.3733265220E+01		
G/RT de	CH4	3.2253653657E+00		

REFORMACION DE GAS NATURAL
EJEMPLO DE WHITE Y SEIDER

Calculo del equilibrio químico por medio de constantes de equilibrio

Temperatura= 1.0670000000E+03

Presion= 1.2190000000E+01

Moles iniciales de CO 1.0000000000E-02

Moles iniciales de CO2 1.0000000000E-02

Moles iniciales de H2 2.0000000000E-01

Moles iniciales de H2O 8.4070000000E+01

Moles iniciales de CH4 1.5140000000E+01

Moles iniciales de N2 5.9000000000E-01

Kequilibrio Reaccion 1 107.70

Kequilibrio Reaccion 2 112.78

Iteracion 1

X reaccion 1= 1.7759091861E+00

X reaccion 2= 1.2880292030E+01

Iteracion 2

X reaccion 1= 2.9547282714E+00

X reaccion 2= 1.1481248393E+01

Iteracion 3

X reaccion 1= 3.7758101847E+00

X reaccion 2= 1.0539514692E+01

Iteracion 4

X reaccion 1= 4.3652743379E+00

X reaccion 2= 9.6768956898E+00

Iteracion 5

X reaccion 1= 4.7971674462E+00

X reaccion 2= 9.3976309668E+00

Iteracion 6

X reaccion 1= 5.1181726481E+00

X reaccion 2= 9.0445191570E+00

Iteracion 7

X reaccion 1= 5.3592388974E+00

X reaccion 2= 8.7809660107E+00

Iteracion 8

X reaccion 1= 5.5416546192E+00

X reaccion 2= 8.5824150646E+00

Iteracion 9

X reaccion 1= 5.6804735503E+00

X reaccion 2= 8.4318068857E+00

Iteracion 10

X reaccion 1= 5.7865665270E+00

X reaccion 2= 8.3169813408E+00

Iteracion 11

X reaccion 1= 5.8679109303E+00

X reaccion 2= 8.2291007876E+00

Iteracion 12

X reaccion 1= 5.9304336143E+00

X reaccion 2= 8.1616467938E+00

Iteracion 13

X reaccion 1= 5.9785801600E+00

X reaccion 2= 8.1097569267E+00

Iteracion 14

X reaccion 1= 6.0157097281E+00

X reaccion 2= 8.0697724369E+00

Iteracion 15

X reaccion 1= 6.0443750639E+00

X reaccion 2= 8.0389218347E+00

Iteracion 16
 X reaccion 1= 6.0665246489E+00
 X reaccion 2= 8.0150948699E+00
 Iteracion 17
 X reaccion 1= 6.0836508473E+00
 X reaccion 2= 7.9966783683E+00
 Iteracion 18
 X reaccion 1= 6.0968996770E+00
 X reaccion 2= 7.9824353187E+00
 Iteracion 19
 X reaccion 1= 6.1071530131E+00
 X reaccion 2= 7.9714149212E+00
 Iteracion 20
 X reaccion 1= 6.1150905380E+00
 X reaccion 2= 7.9628849977E+00
 Iteracion 21
 X reaccion 1= 6.1212367498E+00
 X reaccion 2= 7.9562809277E+00
 Iteracion 22
 X reaccion 1= 6.1259967750E+00
 X reaccion 2= 7.9511668166E+00

R E S U L T A D O S

Iteraciones realizadas = 22
 Reaccion 1: Xanterior = 6.1212367498E+00 Xnueva = 6.1259967750E+00
 Reaccion 2: Xanterior = 7.9562809277E+00 Xnueva = 7.9511668166E+00
 Composiciones finales :
 Moles de CO 6.1359967750E+00
 Moles de CO2 7.9611668166E+00
 Moles de H2 5.0382657591E+01
 Moles de H2O 6.2041669592E+01
 Moles de CH4 1.0628364084E+00
 Moles de N2 5.9000000000E-01

Y (CO) = 4.7872276062E-02
 Y (CO2) = 6.2112023457E-02
 Y (H2) = 3.9307916569E-01
 Y (H2O) = 4.8404131276E-01
 Y (CH4) = 8.2921161497E-03
 Y (N2) = 4.6031058869E-03

REFORMACION DE GAS NATURAL

EJEMPLO DE WHITE Y SEIDER

Calculo de la composicion al equilibrio por medio del metodo de la minimizacion de la energia de Gibbs

Moles iniciales de CO= 1.0000000000E-02

Moles iniciales de CO2= 1.0000000000E-02

Moles iniciales de H2= 2.0000000000E-01

Moles iniciales de H2O= 8.4070000000E+01

Moles iniciales de CH4= 1.5140000000E+01

Moles iniciales de N2= 5.9000000000E-01

Temperatura (K) 1.0670000000E+03

Presion (atm) 1.2190000000E+01

Iteracion 1

NI CO= 1.0000000000E-02 Nueva= 1.2959765652E-01

NI CO2= 1.0000000000E-02 Nueva= 1.4539983380E-01

NI H2= 2.0000000000E-01 Nueva= 1.1003923003E+00

NI H2O= 8.4070000000E+01 Nueva= 8.3679602684E+01

NI CH4= 1.5140000000E+01 Nueva= 1.4885002510E+01

NI N2= 5.9000000000E-01 Nueva= 5.9000000000E-01

LAMBDA 5.5348054978E+01

Iteracion 2

NI CO= 6.6296172661E+00 Nueva= 6.1543155240E+00

NI CO2= 7.5042528448E+00 Nueva= 7.9448460582E+00

NI H2= 5.0035862931E+01 Nueva= 5.0372330797E+01

NI H2O= 6.2461877496E+01 Nueva= 6.2055992362E+01

NI CH4= 1.0261299170E+00 Nueva= 1.0608384210E+00

NI N2= 5.9000000020E-01 Nueva= 5.9000000001E-01

LAMBDA 9.9190766677E-01

Iteracion 3

NI CO= 6.1581618241E+00 Nueva= 6.1586432491E+00

NI CO2= 7.9412806311E+00 Nueva= 7.9420501138E+00

NI H2= 5.0369607987E+01 Nueva= 5.0374130203E+01

NI H2O= 6.2059276920E+01 Nueva= 6.2057256524E+01

NI CH4= 1.0605575482E+00 Nueva= 1.0593066365E+00

NI N2= 5.9000000001E-01 Nueva= 5.9000000000E-01

LAMBDA 9.3786743545E-01

ITERACIONES= 4

Nnueva CO= 6.1586428342E+00 Nant= 6.1586133369E+00

Nnueva CO2= 7.9420499837E+00 Nant= 7.9420023039E+00

Nnueva H2= 5.0374128439E+01 Nant= 5.0373849226E+01

Nnueva H2O= 6.2057257198E+01 Nant= 6.2057382056E+01

Nnueva CH4= 1.0593071819E+00 Nant= 1.0593843589E+00

Nnueva N2= 5.9000000000E-01 Nant= 5.9000000001E-01

Y(CO)= 4.8046311901E-02

Y(CO2)= 6.1959464272E-02

Y(H2)= 3.9299098062E-01

Y(H2O)= 4.8413626432E-01

Y(CH4)= 8.2641264691E-03

Y(N2)= 4.6028524116E-03

G/RT de CO -2.0216301328E+01

G/RT de CO2 -4.2189084008E+01

G/RT de H2 2.5006159435E+00

G/RT de H2O -1.9426431487E+01

G/RT de CH4 6.3957023497E+00

G/RT de N2 2.5006159435E+00

PRODUCCION DE METANOL

Calculo de la composicion al equilibrio por medio del metodo de la minimizacion de la energia de Gibbs

Moles iniciales de CO= 1.8000000000E+00

Moles iniciales de CO2= 1.0000000000E-01

Moles iniciales de H2= 3.7000000000E+00

Moles iniciales de H2O= 1.0000000000E-01

Moles iniciales de CH4O= 1.0000000000E-01

Moles iniciales de O2= 1.0000000000E-01

Temperatura (K) 6.0000000000E+02

Presion (atm) 3.3000000000E+02

Iteracion 1

NI CO= 1.8000000000E+00 Nueva= -1.6326094835E+00

NI CO2= 1.0000000000E-01 Nueva= 3.3617730720E+00

NI H2= 3.7000000000E+00 Nueva= 2.5578782488E-01

NI H2O= 1.0000000000E-01 Nueva= 3.2025393523E+00

NI CH4O= 1.0000000000E-01 Nueva= 2.7083641141E-01

NI O2= 1.0000000000E-01 Nueva= -3.0821562121E+00

LAMBDA 1.5712616436E-02

Iteracion 2

NI CO= 1.7460647238E+00 Nueva= -1.4053451377E+00

NI CO2= 1.5125098918E-01 Nueva= 3.1149796992E+00

NI H2= 3.6458824152E+00 Nueva= 5.8929489119E-01

NI H2O= 1.4874901082E-01 Nueva= 2.8299742316E+00

NI CH4O= 1.0268428701E-01 Nueva= 2.9036543862E-01

NI O2= 5.0000000000E-02 Nueva= -2.7724769654E+00

LAMBDA 8.8574682120E-03

Iteracion 3

NI CO= 1.7181512111E+00 Nueva= -5.4418652432E-01

NI CO2= 1.7750212202E-01 Nueva= 2.1816840441E+00

NI H2= 3.6188087883E+00 Nueva= 1.4643072588E+00

NI H2O= 1.7249787798E-01 Nueva= 1.8106877808E+00

NI CH4O= 1.0434666684E-01 Nueva= 3.6250248018E-01

NI O2= 2.5000000000E-02 Nueva= -1.7961859125E+00

LAMBDA 6.8636595057E-03

Iteracion 4

NI CO= 1.7026232952E+00 Nueva= 1.4781141798E-01

NI CO2= 1.9125814432E-01 Nueva= 1.4276274180E+00

NI H2= 3.6040210234E+00 Nueva= 2.1451223400E+00

NI H2O= 1.8374185568E-01 Nueva= 1.0057553299E+00

NI CH4O= 1.0611856044E-01 Nueva= 4.2456116409E-01

NI O2= 1.2500000000E-02 Nueva= -1.0166913738E+00

LAMBDA 6.0727287066E-03

Iteracion 5

NI CO= 1.6931813445E+00 Nueva= 5.7104730550E-01

NI CO2= 1.9876627950E-01 Nueva= 9.6251476768E-01

NI H2= 3.5951615275E+00 Nueva= 2.5517477384E+00

NI H2O= 1.8873372050E-01 Nueva= 5.1537640810E-01

NI CH4O= 1.0805237598E-01 Nueva= 4.6643792673E-01

NI O2= 6.2500000001E-03 Nueva= -5.3894558796E-01

LAMBDA 5.7318879114E-03

Iteracion 6

NI CO=	1.6867493980E+00	Nueva=	8.0138483414E-01
NI CO2=	2.0314400023E-01	Nueva=	7.0547091488E-01
NI H2=	3.5891807967E+00	Nueva=	2.7662690853E+00
NI H20=	1.9060589977E-01	Nueva=	2.4744241313E-01
NI CH40=	1.1010660179E-01	Nueva=	4.9314425079E-01
NI O2=	3.1250000000E-03	Nueva=	-2.7645666388E-01
LAMBDA	5.5887069929E-03		

Iteracion 7

NI CO=	1.6818013549E+00	Nueva=	9.1873553699E-01
NI CO2=	2.0595135817E-01	Nueva=	5.7051445880E-01
NI H2=	3.5845817842E+00	Nueva=	2.8700462672E+00
NI H20=	1.9092364183E-01	Nueva=	1.0845372468E-01
NI CH40=	1.1224728698E-01	Nueva=	5.1075000405E-01
NI O2=	1.5625000000E-03	Nueva=	-1.3948409177E-01
LAMBDA	5.5389498620E-03		

Iteracion 8

NI CO=	1.6775747716E+00	Nueva=	9.7524363244E-01
NI CO2=	2.0797065490E-01	Nueva=	5.0137047136E-01
NI H2=	3.5806240078E+00	Nueva=	2.9150151952E+00
NI H20=	1.9046684510E-01	Nueva=	3.8213012518E-02
NI CH40=	1.1445457355E-01	Nueva=	5.2338589613E-01
NI O2=	7.8125000001E-04	Nueva=	-6.9791741863E-02
LAMBDA	5.5350494530E-03		

Iteracion 9

NI CO=	1.6736873340E+00	Nueva=	1.0002370810E+00
NI CO2=	2.0959463740E-01	Nueva=	4.6632829717E-01
NI H2=	3.5769398301E+00	Nueva=	2.9299106666E+00
NI H20=	1.8962411261E-01	Nueva=	3.2200896361E-03
NI CH40=	1.1671802864E-01	Nueva=	5.3343462186E-01
NI O2=	3.9062500001E-04	Nueva=	-3.4774193388E-02
LAMBDA	5.5542018688E-03		

Iteracion 10

NI CO=	1.6699468553E+00	Nueva=	1.0092030971E+00
NI CO2=	2.1102058797E-01	Nueva=	4.4861784420E-01
NI H2=	3.5733460995E+00	Nueva=	2.9296807350E+00
NI H20=	1.8858878703E-01	Nueva=	-1.4038852795E-02
NI CH40=	1.1903255672E-01	Nueva=	5.4217905887E-01
NI O2=	1.9531250000E-04	Nueva=	-1.7289495825E-02
LAMBDA	5.5862056361E-03		

Iteracion 11

NI CO=	1.6662564656E+00	Nueva=	1.0100828840E+00
NI CO2=	2.1234761750E-01	Nueva=	4.3962906518E-01
NI H2=	3.5697510961E+00	Nueva=	2.9218794275E+00
NI H20=	1.8745707000E-01	Nueva=	-2.2455529129E-02
NI CH40=	1.2139591695E-01	Nueva=	5.5028805079E-01
NI O2=	9.7656250002E-05	Nueva=	-8.5867680528E-03
LAMBDA	5.6224941687E-03		

PRODUCCION DE METANOL

Calculo de la composicion al equilibrio por medio del metodo de la minimizacion de la energia de Gibbs

Moles iniciales de CO= 1.8000000000E+00
 Moles iniciales de CO2= 1.0000000000E-01
 Moles iniciales de H2= 3.7000000000E+00
 Moles iniciales de H2O= 1.0000000000E-01
 Moles iniciales de CH4O= 1.0000000000E-01

Temperatura (K) 6.0000000000E+02

Presion (atm) 3.3000000000E+02

Iteracion 1

NI CO= 1.8000000000E+00	Nueva= 1.2656792374E+00
NI CO2= 1.0000000000E-01	Nueva= 2.2303468140E-01
NI H2= 3.7000000000E+00	Nueva= 3.0004625188E+00
NI H2O= 1.0000000000E-01	Nueva= -2.3034681302E-02
NI CH4O= 1.0000000000E-01	Nueva= 5.1128608125E-01
LAMBDA 4.0638947873E-01	

Iteracion 2

NI CO= 1.5828576638E+00	Nueva= 8.3676900604E-01
NI CO2= 1.5000000004E-01	Nueva= 2.0034511088E-01
NI H2= 3.4157153277E+00	Nueva= 2.0745733445E+00
NI H2O= 5.0000000000E-02	Nueva= -3.4511105364E-04
NI CH4O= 2.6714233617E-01	Nueva= 9.6288588331E-01
LAMBDA 4.9657254651E-01	

Iteracion 3

NI CO= 1.2123705191E+00	Nueva= 5.2246259122E-01
NI CO2= 1.7499999994E-01	Nueva= 1.8476219854E-01
NI H2= 2.7497410379E+00	Nueva= 1.3992117773E+00
NI H2O= 2.5000000000E-02	Nueva= 1.5237801561E-02
NI CH4O= 6.1262948108E-01	Nueva= 1.2927752108E+00
LAMBDA 9.3804048113E-01	

Iteracion 4

NI CO= 5.6520895449E-01	Nueva= 5.6639269204E-01
NI CO2= 1.8415733741E-01	Nueva= 1.8234195910E-01
NI H2= 1.4828899205E+00	Nueva= 1.4798112607E+00
NI H2O= 1.5842862679E-02	Nueva= 1.7658040973E-02
NI CH4O= 1.2506337086E+00	Nueva= 1.2512653491E+00
LAMBDA 1.0515525933E+00	

ITERACIONES= 5

Nnueva CO= 5.6647881768E-01	Nant= 5.6645371678E-01
Nnueva CO2= 1.8224790921E-01	Nant= 1.8224837164E-01
Nnueva H2= 1.4797013627E+00	Nant= 1.4796525478E+00
Nnueva H2O= 1.7752090939E-02	Nant= 1.7751628432E-02
Nnueva CH4O= 1.2512732732E+00	Nant= 1.2512979118E+00

Y(CO)= 1.6196893688E-01
 Y(CO2)= 5.2108744726E-02
 Y(H2)= 4.2307964416E-01
 Y(H2O)= 5.0757189978E-03
 Y(CH4O)= 3.57766895624E-01
 MULTIPLICADOR DE C 1.9058141006E+01
 MULTIPLICADOR DE O2 -4.7519809898E+01
 MULTIPLICADOR DE H2 2.4694489092E+00

PRODUCCION DE METANOL

Calculo de la composicion al equilibrio por medio del metodo de la minimizacion de la energia de Gibbs, usando la ecuacion de Peng-Robinson

Moles iniciales de CO= 1.8000000000E+00
 Moles iniciales de CO2= 1.0000000000E-01
 Moles iniciales de H2= 3.7000000000E+00
 Moles iniciales de H2O= 1.0000000000E-01
 Moles iniciales de CH4O= 1.0000000000E-01

Temperatura (K) 6.0000000000E+02

Presion (atm) 3.3000000000E+02

COEFICIENTE DE FUGACIDAD CO 1.1326696441E+00

COEFICIENTE DE FUGACIDAD CO2 1.0314676827E+00

COEFICIENTE DE FUGACIDAD H2 1.0907464351E+00

COEFICIENTE DE FUGACIDAD H2O 8.3951047264E-01

COEFICIENTE DE FUGACIDAD CH4O 9.3369139604E-01

Iteracion 1

NI CO= 1.8000000000E+00 Nueva= 1.2402518880E+00

NI CO2= 1.0000000000E-01 Nueva= 2.1493380800E-01

NI H2= 3.7000000000E+00 Nueva= 2.9253051997E+00

NI H2O= 1.0000000000E-01 Nueva= -1.4933807915E-02

NI CH4O= 1.0000000000E-01 Nueva= 5.4481430406E-01

LAMBDA 4.3503300645E-01

COEFICIENTE DE FUGACIDAD CO 1.1339271558E+00

COEFICIENTE DE FUGACIDAD CO2 1.0186696119E+00

COEFICIENTE DE FUGACIDAD H2 1.0936714388E+00

COEFICIENTE DE FUGACIDAD H2O 8.1254172862E-01

COEFICIENTE DE FUGACIDAD CH4O 9.0019804807E-01

Iteracion 2

NI CO= 1.5564910960E+00 Nueva= 7.0509456373E-01

NI CO2= 1.5000000004E-01 Nueva= 1.9315917248E-01

NI H2= 3.3629821920E+00 Nueva= 1.7896666449E+00

NI H2O= 5.0000000000E-02 Nueva= 6.8408276028E-03

NI CH4O= 2.9350890401E-01 Nueva= 1.1017462638E+00

LAMBDA 5.7925114434E-01

COEFICIENTE DE FUGACIDAD CO 1.1506504678E+00

COEFICIENTE DE FUGACIDAD CO2 9.8495894122E-01

COEFICIENTE DE FUGACIDAD H2 1.1150492064E+00

COEFICIENTE DE FUGACIDAD H2O 7.3047584142E-01

COEFICIENTE DE FUGACIDAD CH4O 8.0007388991E-01

Iteracion 3

NI CO= 1.0633186804E+00 Nueva= 3.1703077208E-01

NI CO2= 1.7500000006E-01 Nueva= 1.7813677993E-01

NI H2= 2.4516373609E+00 Nueva= 9.6847188309E-01

NI H2O= 2.5000000000E-02 Nueva= 2.1863220066E-02

NI CH4O= 7.6168131954E-01 Nueva= 1.5048324486E+00

LAMBDA 8.3172780663E-01

COEFICIENTE DE FUGACIDAD CO 1.2734227665E+00

COEFICIENTE DE FUGACIDAD CO2 9.6696670961E-01

COEFICIENTE DE FUGACIDAD H2 1.2414428898E+00

COEFICIENTE DE FUGACIDAD H2O 5.9389372469E-01

COEFICIENTE DE FUGACIDAD CH4O 6.4225463614E-01

Iteracion 4

NI CO= 4.4261027529E-01 Nueva= 3.4142976347E-01

NI CO2= 1.7760894710E-01 Nueva= 1.7156749826E-01

NI H2= 1.2180473912E+00 Nueva= 9.9756202142E-01

NI H2O= 2.2391052906E-02 Nueva= 2.8432501780E-02

NI CH4O= 1.3797807781E+00 Nueva= 1.4870027384E+00

LAMBDA 9.1299768465E-01

COEFICIENTE DE FUGACIDAD CO 1.3269214410E+00
COEFICIENTE DE FUGACIDAD CO2 9.7544651991E-01
COEFICIENTE DE FUGACIDAD H2 1.2938964320E+00
COEFICIENTE DE FUGACIDAD H2O 5.6820599857E-01
COEFICIENTE DE FUGACIDAD CH4O 6.1470440148E-01

Iteracion 5

NI CO= 3.5023270227E-01 Nueva= 3.2604371080E-01
NI CO2= 1.7209311830E-01 Nueva= 1.6911973328E-01
NI H2= 1.0167447591E+00 Nueva= 9.5944662093E-01
NI H2O= 2.79068881740E-02 Nueva= 3.0880266773E-02
NI CH4O= 1.4776741796E+00 Nueva= 1.5048365561E+00
LAMBDA 9.7498657514E-01

COEFICIENTE DE FUGACIDAD CO 1.3453799741E+00
COEFICIENTE DE FUGACIDAD CO2 9.7910103165E-01
COEFICIENTE DE FUGACIDAD H2 1.3118314282E+00
COEFICIENTE DE FUGACIDAD H2O 5.6093376168E-01
COEFICIENTE DE FUGACIDAD CH4O 6.0710808147E-01

Iteracion 6

NI CO= 3.2664876032E-01 Nueva= 3.1924467535E-01
NI CO2= 1.6919410783E-01 Nueva= 1.6841520736E-01
NI H2= 9.6087984361E-01 Nueva= 9.4373497267E-01
NI H2O= 3.0805892230E-02 Nueva= 3.1584792631E-02
NI CH4O= 1.5041571320E+00 Nueva= 1.5123401172E+00
LAMBDA 9.9109121516E-01

COEFICIENTE DE FUGACIDAD CO 1.3514030466E+00
COEFICIENTE DE FUGACIDAD CO2 9.8035484103E-01
COEFICIENTE DE FUGACIDAD H2 1.3176683475E+00
COEFICIENTE DE FUGACIDAD H2O 5.5869777762E-01
COEFICIENTE DE FUGACIDAD CH4O 6.0479307025E-01

Iteracion 7

NI CO= 3.1931063675E-01 Nueva= 3.1692427280E-01
NI CO2= 1.6842214641E-01 Nueva= 1.6821904353E-01
NI H2= 9.4388771263E-01 Nueva= 9.3850567612E-01
NI H2O= 3.1577853575E-02 Nueva= 3.1780956541E-02
NI CH4O= 1.5122672167E+00 Nueva= 1.5148566834E+00
LAMBDA 9.9696354653E-01

COEFICIENTE DE FUGACIDAD CO 1.3533517203E+00
COEFICIENTE DE FUGACIDAD CO2 9.8076643360E-01
COEFICIENTE DE FUGACIDAD H2 1.3195552645E+00
COEFICIENTE DE FUGACIDAD H2O 5.5798773840E-01
COEFICIENTE DE FUGACIDAD CH4O 6.0406003911E-01

Iteracion 8

NI CO= 3.1693151889E-01 Nueva= 3.1616515077E-01
NI CO2= 1.6821966025E-01 Nueva= 1.6815770823E-01
NI H2= 9.3852201842E-01 Nueva= 9.3680342598E-01
NI H2O= 3.1780339828E-02 Nueva= 3.1842291811E-02
NI CH4O= 1.5148488206E+00 Nueva= 1.5156771411E+00
LAMBDA 1.0002119108E+00

COEFICIENTE DE FUGACIDAD CO 1.3539806713E+00
COEFICIENTE DE FUGACIDAD CO2 9.8089988233E-01
COEFICIENTE DE FUGACIDAD H2 1.3201641251E+00
COEFICIENTE DE FUGACIDAD H2O 5.5775993002E-01
COEFICIENTE DE FUGACIDAD CH4O 6.0382507244E-01
ITERACIONES= 9
Nnueva CO= 3.1591932378E-01 Nant= 3.1616498837E-01
Nnueva CO2= 1.6813805845E-01 Nant= 1.6815769510E-01
Nnueva H2= 9.3625282274E-01 Nant= 9.3680306179E-01
Nnueva H2O= 3.1861941490E-02 Nant= 3.1842304939E-02
Nnueva CH4O= 1.5159426177E+00 Nant= 1.5156773166E+00
Y(CO)= 1.0643770504E-01
Y(CO2)= 5.6648098812E-02
Y(H2)= 3.15436866721E-01
Y(H2O)= 1.0734740407E-02
Y(CH4O)= 5.1074258853E-01

PRODUCCION DE METANOL

Calculo de la composicion al equilibrio por medio del metodo de la minimizacion de la energia de Gibbs, usando modificacion de Stryjek-Vera

Moles iniciales de CO= 1.8000000000E+00
Moles iniciales de CO2= 1.0000000000E-01
Moles iniciales de H2= 3.7000000000E+00
Moles iniciales de H2O= 1.0000000000E-01
Moles iniciales de CH4O= 1.0000000000E-01

Temperatura (K) 6.0000000000E+02

Presion (atm) 3.3000000000E+02

ITERACIONES= 9

Nnueva CO= 3.1370914725E-01 Nant= 3.1397702468E-01

Nnueva CO2= 1.6786623474E-01 Nant= 1.6788810348E-01

Nnueva H2= 9.3101699810E-01 Nant= 9.3161835956E-01

Nnueva H2O= 3.2133765572E-02 Nant= 3.2111896626E-02

Nnueva CH4O= 1.5184246182E+00 Nant= 1.5181348717E+00

Y(CO)= 1.0587012685E-01

Y(CO2)= 5.6651263510E-02

Y(H2)= 3.1419832209E-01

Y(H2O)= 1.0844458528E-02

Y(CH4O)= 5.1243582902E-01

REDUCCION DE OXIDO DE FIERRO

Calculo de la composicion al equilibrio por medio del metodo de la minimizacion de la energia de Gibbs

Moles iniciales de CO= 1.5000000000E+00
 Moles iniciales de CO2= 1.0000000000E-02
 Moles iniciales de H2= 1.5000000000E+00
 Moles iniciales de H2O= 1.0000000000E-02
 Moles iniciales de C= 4.0000000000E+00
 Moles iniciales de Fe= 4.0000000000E+00
 Moles iniciales de Fe3O4= 4.0000000000E+00
 Temperatura (K) 7.0000000000E+02
 Presion (atm) 1.0000000000E+00
 G/RT DE CO -3.0083569362E+01
 G/RT DE CO2 -6.8433131685E+01
 G/RT DE H2 0.0000000000E+00
 G/RT DE H2O -3.6161366157E+01

Iteracion 1

NI	CO=	1.5000000000E+00	Nueva=	-2.0251291766E+01
NI	CO2=	1.0000000000E-02	Nueva=	-8.1920341312E-02
NI	H2=	1.5000000000E+00	Nueva=	1.4690003555E+00
NI	H2O=	1.0000000000E-02	Nueva=	4.0999644499E-02
NI	C=	4.0000000000E+00	Nueva=	2.5843212107E+01
NI	Fe=	4.0000000000E+00	Nueva=	-1.2428099603E+01
NI	Fe3O4=	4.0000000000E+00	Nueva=	9.4760332009E+00
LAMBDA		3.4480710759E-02		

Iteracion 2

NI	CO=	7.5000000000E-01	Nueva=	-8.9587397039E+00
NI	CO2=	6.8305212984E-03	Nueva=	-2.9244573953E-02
NI	H2=	1.4989311102E+00	Nueva=	1.4657660723E+00
NI	H2O=	1.1068889776E-02	Nueva=	4.4233927712E-02
NI	C=	4.7531694787E+00	Nueva=	1.2497984278E+01
NI	Fe=	3.4335474493E+00	Nueva=	-2.3772461931E+00
NI	Fe3O4=	4.1868175169E+00	Nueva=	6.1257487310E+00
LAMBDA		4.8646084108E-02		

Iteracion 3

NI	CO=	3.7500000000E-01	Nueva=	-2.5993917518E+00
NI	CO2=	5.0756091806E-03	Nueva=	-1.1839589575E-02
NI	H2=	1.4973177610E+00	Nueva=	1.4610842899E+00
NI	H2O=	1.2682239000E-02	Nueva=	4.8915710127E-02
NI	C=	5.1299243908E+00	Nueva=	8.1218313214E+00
NI	Fe=	3.1508750930E+00	Nueva=	9.2143361447E-01
NI	Fe3O4=	4.2830416357E+00	Nueva=	5.0261687952E+00
LAMBDA		6.3025384823E-02		

Iteracion 4

NI	CO=	1.8750000000E-01	Nueva=	-1.0278987080E+00
NI	CO2=	4.0095235301E-03	Nueva=	-5.3680634092E-03
NI	H2=	1.4950341325E+00	Nueva=	1.4548174402E+00
NI	H2O=	1.4965867462E-02	Nueva=	5.5182559759E-02
NI	C=	5.3184904765E+00	Nueva=	6.5432667714E+00
NI	Fe=	3.0103638859E+00	Nueva=	2.1149107937E+00
NI	Fe3O4=	4.3298787714E+00	Nueva=	4.6283630688E+00
LAMBDA		7.7135181550E-02		

Iteracion 5

NI CO=	9.3750000001E-02	Nueva=	-4.1697860982E-01
NI CO2=	3.2861816591E-03	Nueva=	-2.6245902236E-03
NI H2=	1.4919320107E+00	Nueva=	1.4468797734E+00
NI H2O=	1.8067989323E-02	Nueva=	6.312026618E-02
NI C=	5.4129638183E+00	Nueva=	5.9296032000E-00
NI Fe=	2.9412927645E+00	Nueva=	2.5631693273E+00
NI Fe3O4=	4.3529024118E+00	Nueva=	4.4722768909E+00
LAMBDA	9.1780642594E-02		

Iteracion 6

NI CO=	4.6875000001E-02	Nueva=	-1.6893973098E-01
NI CO2=	2.7436872175E-03	Nueva=	-1.2831513777E-03
NI H2=	1.4877970874E+00	Nueva=	1.4371569736E+00
NI H2O=	2.2202912613E-02	Nueva=	7.2843026407E-02
NI C=	5.4603813128E+00	Nueva=	5.6802228824E+00
NI Fe=	2.9084239653E+00	Nueva=	2.7785027445E+00
NI Fe3O4=	4.3638586782E+00	Nueva=	4.4071657518E+00
LAMBDA	1.0860009367E-01		

Iteracion 7

NI CO=	2.3437500000E-02	Nueva=	-6.6700964916E-02
NI CO2=	2.3063721688E-03	Nueva=	-5.2828511112E-04
NI H2=	1.4822975663E+00	Nueva=	1.4254664663E+00
NI H2O=	2.7702433714E-02	Nueva=	8.453353697E-02
NI C=	5.4842561278E+00	Nueva=	5.5772292500E+00
NI Fe=	2.8943145086E+00	Nueva=	2.8650819989E+00
NI Fe3O4=	4.3685618305E+00	Nueva=	4.3783060004E+00
LAMBDA	1.3000831566E-01		

Iteracion 8

NI CO=	1.1718750000E-02	Nueva=	-2.4907834963E-02
NI CO2=	1.9378431504E-03	Nueva=	-5.3605169398E-05
NI H2=	1.4749090507E+00	Nueva=	1.4115541378E+00
NI H2O=	3.5090949300E-02	Nueva=	9.8445862215E-02
NI C=	5.4963434068E+00	Nueva=	5.5349614401E+00
NI Fe=	2.8905140392E+00	Nueva=	2.9075731127E+00
NI Fe3O4=	4.3698286536E+00	Nueva=	4.3641422958E+00
LAMBDA	1.5997601212E-01		

Iteracion 9

NI CO=	5.8593750001E-03	Nueva=	-8.3278186786E-03
NI CO2=	1.6192591898E-03	Nueva=	2.6391477374E-04
NI H2=	1.4647737844E+00	Nueva=	1.3951155027E+00
NI H2O=	4.5226215617E-02	Nueva=	1.1488449732E-01
NI C=	5.5025213658E+00	Nueva=	5.5180639039E+00
NI Fe=	2.8932430818E+00	Nueva=	2.9328133812E+00
NI Fe3O4=	4.3689189727E+00	Nueva=	4.3557288729E+00
LAMBDA	2.0650225594E-01		

Iteracion 10

NI CO=	2.9296875001E-03	Nueva=	-2.1208385417E-03
NI CO2=	1.3393775104E+00	Nueva=	4.7811713915E-04
NI H2=	1.4503891921E+00	Nueva=	1.3758544364E+00
NI H2O=	5.9610807933E-02	Nueva=	1.3414556362E-01
NI C=	5.5057309350E+00	Nueva=	5.5116427214E+00
NI Fe=	2.9014144379E+00	Nueva=	2.9522357195E+00
NI Fe3O4=	4.3661951874E+00	Nueva=	4.3492547602E+00
LAMBDA	2.9003785703E-01		

Iteracion 11

NI CO=	1.4648437500E-03	Nueva=	-4.2927136425E-05
NI CO2=	1.0895793979E-03	Nueva=	6.1544201486E-04
NI H2=	1.4287712913E+00	Nueva=	1.3536571224E+00
NI H2O=	8.1228708748E-02	Nueva=	1.5634287761E-01
NI C=	5.5074455768E+00	Nueva=	5.5094274851E+00
NI Fe=	2.9161545335E+00	Nueva=	2.9706481259E+00
NI Fe3O4=	4.3612818222E+00	Nueva=	4.3431172914E+00
LAMBDA	4.8576470178E-01		

Iteracion 12

NI CO=	7.3242187502E-04	Nueva=	4.8650465860E-04
NI CO2=	8.5926019344E-04	Nueva=	6.8974334908E-04
NI H2=	1.3922834794E+00	Nueva=	1.3292852954E+00
NI H2O=	1.1771652058E-01	Nueva=	1.8071470459E-01
NI C=	5.5084083179E+00	Nueva=	5.5088237520E+00
NI Fe=	2.9426255972E+00	Nueva=	2.9894355220E+00
NI Fe3O4=	4.3524581343E+00	Nueva=	4.3368548260E+00
LAMBDA	1.2741595800E+00		

Iteracion 13

NI CO=	4.1908409784E-04	Nueva=	5.1235196885E-04
NI CO2=	6.4326868223E-04	Nueva=	7.0261185885E-04
NI H2=	1.3120137397E+00	Nueva=	1.3118216903E+00
NI H2O=	1.9798626026E-01	Nueva=	1.9817830971E-01
NI C=	5.5089376472E+00	Nueva=	5.5087850362E+00
NI Fe=	3.0022689113E+00	Nueva=	3.0025719141E+00
NI Fe3O4=	4.3325770296E+00	Nueva=	4.3324760286E+00
LAMBDA	1.1023125789E+00		

Iteracion 14

NI CO=	5.2189444526E-04	Nueva=	5.2354739369E-04
NI CO2=	7.0868341229E-04	Nueva=	7.0543439345E-04
NI H2=	1.3118020412E+00	Nueva=	1.3118216120E+00
NI H2O=	1.9819795679E-01	Nueva=	1.9817838600E-01
NI C=	5.5087694222E+00	Nueva=	5.5087710182E+00
NI Fe=	3.0026029151E+00	Nueva=	3.0025846031E+00
NI Fe3O4=	4.3324656950E+00	Nueva=	4.3324717990E+00
LAMBDA	9.8226036047E-01		

ITERACIONES= 15

Nnueva	CO=	5.2355001627E-04	Nant=	5.2351807098E-04
Nnueva	CO2=	7.0544183228E-04	Nant=	7.0549202987E-04
Nnueva	H2=	1.3118216112E+00	Nant=	1.3118212648E+00
Nnueva	H2O=	1.9817838882E-01	Nant=	1.9817873517E-01
Nnueva	C=	5.5087710081E+00	Nant=	5.5087709899E+00
Nnueva	Fe=	3.0025846169E+00	Nant=	3.0025849280E+00
Nnueva	Fe3O4=	4.3324717944E+00	Nant=	4.3324716907E+00
Y(CO)=		3.4643989699E-04		
Y(CO2)=		4.6680009190E-04		
Y(H2)=		8.6804952674E-01		
Y(H2O)=		1.3113723327E-01		

REDUCCION DE OXIDO DE FIERRO

Calculo de la composicion al equilibrio por medio del metodo de la minimizacion de la energia de Gibbs

Moles inicilaes de CO= 1.5000000000E+00
 Moles inicilaes de CO2= 1.0000000000E-02
 Moles inicilaes de H2= 1.5000000000E+00
 Moles inicilaes de H2O= 1.0000000000E-02
 Moles inicilaes de C= 4.0000000000E+00
 Moles inicilaes de Fe= 4.0000000000E+00
 Moles inicilaes de FeO= 4.0000000000E+00
 Temperatura (K) 1.1000000000E+03

Presion (atm) 1.0000000000E+00
 G/RT DE CO -2.3500189168E+01
 G/RT DE CO2 -4.4551781703E+01
 G/RT DE H2 0.0000000000E+00
 G/RT DE H2O -2.1120543051E+01

Iteracion 1

NI CO=	1.5000000000E+00	Nueva=	1.2106003416E+01
NI CO2=	1.0000000000E-02	Nueva=	1.3450436554E-01
NI H2=	1.5000000000E+00	Nueva=	1.4458736610E+00
NI H2O=	1.0000000000E-02	Nueva=	6.4126338996E-02
NI C=	4.0000000000E+00	Nueva=	-6.7305077813E+00
NI Fe=	4.0000000000E+00	Nueva=	1.4909138486E+01
NI FeO=	4.0000000000E+00	Nueva=	-6.9091384865E+00
LAMBDA	1.8333253378E-01		

Iteracion 2

NI CO=	3.4444254795E+00	Nueva=	4.0086513584E+01
NI CO2=	3.2825700801E-02	Nueva=	5.4689314944E-01
NI H2=	1.4900768811E+00	Nueva=	1.3966359353E+00
NI H2O=	1.9923118873E-02	Nueva=	1.1336406470E-01
NI C=	2.0327488197E+00	Nueva=	-3.5123406734E+01
NI Fe=	6.0000000000E+00	Nueva=	4.3763663946E+01
NI FeO=	2.0000000000E+00	Nueva=	-3.5763663946E+01
LAMBDA	2.6480481360E-02		

Iteracion 3

NI CO=	4.4147256113E+00	Nueva=	5.9546286939E+01
NI CO2=	4.6438454303E-02	Nueva=	8.5501606186E-01
NI H2=	1.4876025199E+00	Nueva=	1.3653508859E+00
NI H2O=	2.2397480099E-02	Nueva=	1.2464913409E-01
NI C=	1.0488359344E+00	Nueva=	-5.4891303001E+01
NI Fe=	7.0000000000E+00	Nueva=	6.3850968197E+01
NI FeO=	1.0000000000E+00	Nueva=	-5.5850968197E+01
LAMBDA	8.7949249741E-03		

Iteracion 4

NI CO=	4.8996035569E+00	Nueva=	7.0648658760E+01
NI CO2=	5.3549833697E-02	Nueva=	1.0337629092E+00
NI H2=	1.4867032243E+00	Nueva=	1.3813271280E+00
NI H2O=	2.3296775724E-02	Nueva=	1.2867287209E-01
NI C=	5.5684660942E-01	Nueva=	-8.6172421689E+01
NI Fe=	7.5000000000E+00	Nueva=	7.5314857470E+01
NI FeO=	5.0000000001E-01	Nueva=	-6.7314857470E+01
LAMBDA	3.6865077850E-03		

Iteracion 5

NI CO=	5.1419879608E+00	Nueva=	7.6543151673E+01
NI CO2=	5.7163396831E-02	Nueva=	1.1292240344E+00
NI H2=	1.4863147545E+00	Nueva=	1.3796011625E+00
NI H2O=	2.3685245524E-02	Nueva=	1.3039883753E-01
NI C=	3.1084864235E-01	Nueva=	-7.2162375708E+01
NI Fe=	7.7500000000E+00	Nueva=	8.1401998579E+01
NI FeO=	2.5000000000E-01	Nueva=	-7.3401998579E+01
LAMBDA	1.6971705101E-03		

Iteracion 6

NI CO=	5.2631679102E+00	Nueva=	7.9576239848E+01
NI CO2=	5.8982866530E-02	Nueva=	1.1784676757E+00
NI H2=	1.4861336433E+00	Nueva=	1.3787989507E+00
NI H2O=	2.3866356685E-02	Nueva=	1.3120104938E-01
NI C=	1.8784922321E-01	Nueva=	-7.5244707524E+01
NI Fe=	7.8750000000E+00	Nueva=	8.4534376249E+01
NI FeO=	1.2500000000E-01	Nueva=	-7.6534376249E+01
LAMBDA	8.1529491967E-04		

Iteracion 7

NI CO=	5.3237549803E+00	Nueva=	8.1114245924E+01
NI CO2=	5.9895576807E-02	Nueva=	1.2034667548E+00
NI H2=	1.4860461339E+00	Nueva=	1.3784118937E+00
NI H2O=	2.3953866115E-02	Nueva=	1.3158810635E-01
NI C=	1.2634944291E-01	Nueva=	-7.6807712679E+01
NI Fe=	7.9375000000E+00	Nueva=	8.6122767539E+01
NI FeO=	6.2500000001E-02	Nueva=	-7.8122767540E+01
LAMBDA	3.9969166806E-04		

Iteracion 8

NI CO=	5.3540478080E+00	Nueva=	8.1888614636E+01
NI CO2=	6.0352652679E-02	Nueva=	1.2160604881E+00
NI H2=	1.4860031134E+00	Nueva=	1.3782217450E+00
NI H2O=	2.3996886624E-02	Nueva=	1.3177825504E-01
NI C=	9.5599539292E-02	Nueva=	-7.7594675125E+01
NI Fe=	7.9687500000E+00	Nueva=	8.6922513868E+01
NI FeO=	3.1250000001E-02	Nueva=	-7.8922513868E+01
LAMBDA	1.9790063494E-04		

Iteracion 9

REDUCCION DE OXIDO DE FIERRO

Calculo de la composicion al equilibrio por medio del metodo de la minimizacion de la energia de Gibbs

Moles iniciales de CO= 1.5000000000E+00
 Moles iniciales de CO2= 2.0100000000E+00
 Moles iniciales de H2= 1.5000000000E+00
 Moles iniciales de H2O= 1.0000000000E-02
 Moles iniciales de C= 2.0000000000E+00
 Moles iniciales de Fe= 8.0000000000E+00
 Temperatura (K) 1.1000000000E+03
 Presion (atm) 1.0000000000E+00
 Iteracion 1

NI	CO=	1.5000000000E+00	Nueva=	4.1570755960E+00
NI	CO2=	2.0100000000E+00	Nueva=	6.6689600644E-01
NI	H2=	1.5000000000E+00	Nueva=	1.4708676084E+00
NI	H2O=	1.0000000000E-02	Nueva=	3.9132391566E-02
NI	C=	2.0000000000E+00	Nueva=	6.8602839762E-01
NI	Fe=	8.0000000000E+00	Nueva=	8.0000000000E+00

LAMBDA 1.2713541961E+00

Iteracion 2

NI	CO=	4.8780842083E+00	Nueva=	4.8407129999E+00
NI	CO2=	3.0243910205E-01	Nueva=	3.0427807686E-01
NI	H2=	1.4629624117E+00	Nueva=	1.4292691535E+00
NI	H2O=	4.7037588259E-02	Nueva=	8.0730846507E-02
NI	C=	3.2947668981E-01	Nueva=	3.6500892342E-01
NI	Fe=	8.0000000000E+00	Nueva=	8.0000000000E+00

LAMBDA 1.4278770040E+00

Iteracion 3

NI	CO=	4.8247227193E+00	Nueva=	4.8281657175E+00
NI	CO2=	3.0506493189E-01	Nueva=	3.0330597399E-01
NI	H2=	1.4148525831E+00	Nueva=	1.4147776659E+00
NI	H2O=	9.5147416902E-02	Nueva=	9.5222334072E-02
NI	C=	3.8021234908E-01	Nueva=	3.7852830811E-01
NI	Fe=	8.0000000000E+00	Nueva=	8.0000000000E+00

LAMBDA 9.9693789070E-01

ITERACIONES= 4

Nnueva	CO=	4.8281576371E+00	Nant=	4.8281551747E+00
Nnueva	CO2=	3.0331005824E-01	Nant=	3.0331136011E-01
Nnueva	H2=	1.4147777538E+00	Nant=	1.4147778953E+00
Nnueva	H2O=	9.5222246242E-02	Nant=	9.5222104667E-02
Nnueva	C=	3.7853230466E-01	Nant=	3.7853346483E-01
Nnueva	Fe=	8.0000000000E+00	Nant=	8.0000000000E+00

Y(CO)= 7.2697148561E-01
 Y(CO2)= 4.5669131005E-02
 Y(H2)= 2.1302185280E-01
 Y(H2O)= 1.4337530590E-02

APENDICE 5

Listado de los programas

Los dos métodos de solución propuestos en los capítulos 3 y 4 , fueron aplicar en el desarrollo de programas de computo. Dichos programas fueron realizados en Turbo Pascal v.5.0 para computadoras PC. A continuación se presentan los listados de los programas del método de minimización de la energía de Gibbs y del método de las constantes de equilibrio. Además se incluye el programa para el cálculo de los coeficientes de fugacidad por medio de ecuaciones estado cúbicas (Soave y Peng-Robinson). Aparte se tiene la opción de manejar las modificaciones de estas ecuaciones para el caso de compuestos polares y se cuenta con diferentes reglas de mezclado.

Los programas tienen una parte que permite el manejo de bases de datos de propiedades termodinámicas. De este modo se evita el tener que dar los datos cada vez que se quiere hacer algún cálculo.

Programa Gibbs

Este programa nos permite calcular el mínimo de la energía de Gibbs para una fase multicomponente en presencia de fases sólidas puras. El programa puede manejar 10 componentes en la fase multicomponente, 3 fases sólidas puras y 5 elementos químicos. En caso de necesitar un mayor valor en algunos de estos parámetros se cambian estos al principio del programa.

Al inicio del programa se preguntan las especies presentes al equilibrio. Se dan primero la especies de la fase multicomponente y después las fases sólidas puras. Se tiene la opción de dar las propiedades termodinámicas o que de estas se encuentren en un archivo. En este último caso se pregunta el nombre del archivo, además se debe tener cuidado de escribir los nombres de las especies químicas igual a como están en el archivo. Las condiciones necesarias de reacción son la temperatura, la presión y la composición inicial. Hay que recordar

que si se da cero como valor para la composición de algún componente, este componente no apareciera en las subsecuentes iteraciones.

Además se tiene la opción de dejar que el programa tome decisiones sobre que compuestos pueden no estar en el equilibrio.

Programa Constantes

En este programa se preguntan el número de reacciones y los coeficientes de cada componente en las reacciones. Para los reactivos los coeficientes son negativos. También se debe tener cuidado de no dar cero como composición inicial para algún componente.

Programa Fugacidades

En este programa se escoge que ecuación de estado que se usa. Se preguntan los valores de las constantes de interacción binarios y da la regla de mezclado. En caso de usar la regla de mezclado como la de Soave, se dan los mismos valores para K_{ij} y K_{ji} . Recordar que K_{ii} es igual a cero.

Programa Archivos

Este programa sirve para crear o modificar archivos. En el caso de crear o agregar datos a un archivo, el programa pregunta los parámetros necesarios. En caso de no existir valores dar cero. Para el caso de los coeficientes atómicos dar el número de átomos presentes en la especie, y para el caso de los coeficientes de formación dar el número de moléculas necesarias para formar la especie.

También se puede saber que especies están en un archivo y actualizar los datos de alguna especie.

Program Gibbs;

(* PROGRAMA PARA CALCULAR LAS COMPOSICIONES AL EQUILIBRIO DE
UN REACTOR ISOTERMICO A TRAVES DE LA MINIMIZACION DE LA
ENERGIA LIBRE DE GIBBS *)

uses Crt, Archi, Printer, Fugacidad;

Const

Elementos =4; (*elementos quimicos presentes *)
Componentes=4; (*especies quimicas presentes al equilibrio *)
Solidos =3; (*especies solidas presentes *)
Tolerancia =1E-4;
T0 =298.15; (*temperatura de referencia (K) *)
R =8.317; (*constante universal de los gases (Joules) *)

Type

Vector2 =Array [1..Componentes+Solidos,1..5] of real;
Vector3 =Array [1..Elementos+Solidos+1,1..Elementos+Solidos+2] of real;
Vector4 =Array [1..Elementos,1..Componentes+Solidos] of real;
Vector5 =Array [1..Elementos] of real;
Vector6 =Array [1..Elementos,1..Elementos] of real;
Vector7 =Array [1..Componentes+Solidos] of string[8];
Vector8 =Array [1..Elementos] of string[2];
Vector9 =Array [1..Elementos,1..5] of real;
Vectora =Array [1..Elementos,1..Componentes+Solidos] of string[2];
Vectorc =Array [1..5] of string[2];
Vectorn =Array [1..2,1..4] of real;
Archivo =string[12];
Datos =record
Nombre :string[8];
Entalpia:real;
EnergiaG:real;
CaCal :array[1..5] of real;
TempC :real;
PresC :real;
Factor :real;
CoefMat :real;
Comp :Vectorc;
Co :Vectorn;
end;

Var

NO, (* composicion inicial *)
NI, (* composicion en la iteracion k *)
Nnueva, (* composicion en la iteracion k+1 *)
Hfstd, (* entalpia de formacion a T0(joules/gmol) *)
Sfstd, (* entropia de formacion a T0(joules/gmol/K) *)
Gfstd, (* energia libre de formacion a T0(joule/gmol) *)
Lkeq, (* logaritmo ctes de eq. a la temp del reactor *)
Lkeqc, (* ctes corregidas por fugacidad *)
Sreac, (* propiedades para el componente j a la temp. *)
Hreac, (* del reactor *)

```

Greac,
Tc, (* temperatura critica *)
Pc, (* presion critica *)
Omega, (* factor acentrico *)
pMathias, (* coeficiente Mathias *)
Fuga :Vector1; (* fugacidades *)
CP :Vector2; (* capacidades calorificas (Joules/gmol/K) *)
Afor, (* coeficientes de formacion *)
Af,
Aant, (* coeficientes elementos desordenados *)
Ai :Vector4; (* coeficientes Elementos *)
PI, (* multiplicadores de lagrange *)
GEL, (* Energia Gibbs de elementos quimicos *)
Bt :Vector5; (* numero total de atomos de un elemento *)
Nombres :Vector7; (* nombres de los compuestos reaccionantes *)
NomEl :Vector8; (* nombres de los elementos quimicos *)
Cpuro :Vector9; (* Cp elementos puros *)
Kaij :Vectork; (* parametros de interaccion *)
ElArch :Vectora; (* elementos presentes *)
Iteracion, (* contador de itaracion *)
Renglones,
Columnas,
i,j,k :integer; (* contadores *)
Paso, (* cuanto se toma del limite de lambda *)
Incremento, (* mide diferencia entre iteraciones *)
U, (* variable de la matriz de multiplicadores *)
Error, (* error en el grado de avance *)
Errortotal, (* error mayor *)
T, (* temperatura (K) *)
P :real; (* presion del reactor (Atm) *)
NombreArch :Archivo; (* variables para el manejo *)
NombreLog :file of Datos; (* de archivos *)
Registro :Datos;
Calculo,
ReglaMez,
Fase,
Ecuacion,
OpcionArch :char;

```

```

{-----}
Function Pot(Base:real;Exponente:real):real;

```

```

Begin
  if base=0 then Pot:=0 else Pot:=Exp(Exponente*Ln(Base));
end;

```

```

Procedure Inicializa (Var Hfstd,Gfstd:Vector1;Var Tc,Pc,Omega,pMathias:Vector1;
  Var CP:Vector2;Var Ecuacion,ReglaMez:char;
  Var Ai:Vector4);

```

```

Var
  k,kk,k2,k3 :integer;
  OpcionDat :char;

```

```

Procedure LeerArch(Var Hfstd,Gfstd:Vector1;Var Tc,Pc,Omega,pMathias:Vector1;

```

```

Var CP:Vector2);

Var
  i,k,j,l :integer;
  NombreTemp :string[8];

Procedure OrdenaA(Var A1:Vector4;Var NomEl:Vector8);
Var
  i,j,k,l :integer;
  Compara :Boolean;
begin
  for i:=1 to Elementos do
    begin
      NomEl[i]:='';
      for j:=1 to Componentes+Solidos do
        begin
          Ai[i,j]:=0;
          Af[i,j]:=0;
        end; {for}
      end;
      k:=1;
      Ai[k,k]:=Aant[k,k];
      Af[k,k]:=Afor[k,k];
      NomEl[k]:=ElArch[k,k];
      for i:=1 to Componentes+Solidos do
        begin
          j:=1;
          While (ElArch[j,i]<>'') do
            begin
              l:=0;
              Compara:=True;
              repeat
                l:=l+1;
                if ElArch[j,i]=NomEl[l] then
                  begin
                    Compara:=False;
                    Ai[l,i]:=Aant[j,i];
                    Af[l,i]:=Afor[j,i];
                  end;
                if (l=k) and (Compara) then
                  begin
                    Compara:=False;
                    k:=k+1;
                    NomEl[k]:=ElArch[j,i];
                    Ai[k,i]:=Aant[j,i];
                    Af[k,i]:=Afor[j,i];
                  end; {if}
              until (Compara=False);
              j:=j+1;
            end; {while}
          end;
        end; {Ordena}
      end;
    begin
      Clrscr;
    end;
  end;

```

```

write('Nombre del archivo a leer ');readln(Nombreach);
NOMBREARCH:='PRUEBA1';
Assign(Nombrelog,Nombreach);
Reset(Nombrelog);
i:=FileSize(Nombrelog);
for l:=1 to Elementos do
  for k:=1 to Componentes+Solidos do
    begin
      ElArch[l,k]:='';Aant[l,k]:=0;Afor[l,k]:=0;
    end; {for}
for k:=1 to Componentes+Solidos do
begin
  NombreTemp:=Nombres[k];
  j:=0;
  Reset(Nombrelog);
  repeat
    j:=j+1;
    Read(Nombrelog,Registro);
  until (registro.nombre=NombreTemp) or (j=i);
  if registro.nombre<>NombreTemp then
    writeln('Las propiedades de ',NombreTemp,' no estan en ',Nombreach)
  else
    begin
      Clrscr;
      with Registro do
        begin
          writeln(nombres[k]);
          writeln('Entalpia ',entalpia);Hfstd[k]:=entalpia;
          writeln('Energia libre ',energiag);Gfstd[k]:=energiag;
          for l:=1 to 5 do
            begin
              CP[k,l]:=CaCal[l];
              writeln('Coeficiente ',l,' del CP ',CaCal[l]);
            end; {for}
          if (k<=Componentes) then begin
            writeln('Temperatura critica ',TempC);Tc[k]:=TempC;
            writeln('Presion critica ',PresC);Pc[k]:=PresC;
            writeln('Factor acentrico ',Factor);Omega[k]:=Factor;
            if (Ecuacion in ['m','M'])
              then begin
                writeln('Factor Mathias ',CoefMat);pMathias[k]:=CoefMat;
              end {then}
            else pMathias[k]:=0;
          end; {if}
          l:=1;
          if Comp[l]<>' ' then writeln('Elemento Atomos Reaccion Formacion');
          while (Comp[l]<>' ') do
            begin
              writeln(' ',Comp[l]:3,' ',Co[1,l]:2:1,' ',Co[2,l]:2:1);
              ElArch[l,k]:=Comp[l];
              Aant[l,k]:=Co[1,l];
              Afor[l,k]:=Co[2,l];
              l:=l+1;
            end; {while}
          GotoXY(20,15);

```

```

        write('Oprima cualquier tecla para continuar');readln;
    end; {with}
end; {else}
end; {for}
Close(Nombrelog);
OrdenaA(A1,NomEl);
Assign(Nombrelog,Nombreach);
Reset(Nombrelog);
i:=FileSize(Nombrelog);
for k:=1 to Elementos do
begin
    NombreTemp:=NomEl[k];
    j:=0;
    Reset(Nombrelog);
    repeat
        j:=j+1;
        Read(Nombrelog,Registro);
    until (registro.nombre=NombreTemp) or (j=i);
    if registro.nombre<>NombreTemp then
        writeln('Las propiedades de ',NombreTemp,' no estan en ',Nombreach)
    else
        for l:=1 to 5 do
            Cpuro[k,l]:=Registro.CaCal[l];
        end; {for}
end; {Lectura}

Procedure PedirDat;
Var
    i,k,kk      :integer;
    NombreTemp  :string[8];
Begin
    for i:=1 to componentes+Solidos do
        begin
            Clrscr;
            writeln(nombres[i]);
            write('Entalpia (Joules/gmol) de formacion a ',T0,' ');readln(Hfstnd[i]);
            write('Energia libre de formacion (Joule/gmol) ');readln(Gfstnd[i]);
            writeln('CP de la forma a1+a2*T+a3*T^2+a4*T^3+a5/T^2 (Joule/gmol/K)');
            for j:=1 to 5 do
                begin
                    write('a',j,'? ');readln(CP[i,j]);
                end; {for}
            if not(Ecuacion in ['I','i']) and (k<=Componentes) then
                begin
                    write('Temperatura critica (K) ');readln(tc[i]);
                    write('Presion critica (atm) ');readln(Pc[i]);
                    write('Factor acentrico ');readln(Omega[i]);
                    if (Ecuacion in ['m','M'])
                        then begin
                            write('Factor de Mathias ');readln(pMathias[i]);
                            end {then}
                        else pMathias[i]:=0;
                    end; {if}
                end; {for}
            Clrscr;

```

```

writeln('Dar el nombre de los elementos quimicos presentes');
for k:=1 to elementos do
begin
write('Elemento quimico ',k,' ? ');readln(NomEl[k]);
end;
for k:=1 to componentes+Solidos do
begin
Clrscr;
writeln('Dar el numero de atomos presentes de cada elemento en ',Nombres[k]);
for kk:=1 to Elementos do
begin
write('Atomos de ',NomEl[kk],' ');readln(Ai[kk,k]);
write('Atomos de ',NomEl[kk],' en reaccion de formacion ');
readln(Af[kk,k]);
end;{for}
end;{for}
for k:=1 to Elementos do
begin
Clrscr;
writeln('Para el elemento ',NomEl[k],' dar el Cp');
writeln('Cp=a1+a2T+a3T^2+a4T^3+a5/T^2 (Joules/grmol/K)');
for kk:=1 to 5 do
begin
write('Coeficiente ',kk,' del Cp ');readln(Cpuro[k,kk]);
end;
end;
End; {PedirDat}

Begin
Clrscr;
Renglones:=Elementos+Solidos+1;
Columnas:=Elementos+Solidos+2;

writeln('Compuestos presentes al equilibrio');
writeln('Dar primero los que estan en una fase y al final solidos');
writeln('Nombre de los compuestos en forma de formula condensada');
writeln('en orden alfabetico Ej.O2S, C3H4O');
for k:=1 to Componentes+Solidos do
begin
write('Compuesto ',k,' ');readln(Nombres[k]);
end; {for}
Clrscr;
writeln('Fase de reaccion <L> liquido');
writeln(' <G> gas ');
readln(Fase);
Repeat
Clrscr;
writeln('Ecuacion de Estado a usar');
writeln(' <S> Soave');
writeln(' <P> Peng-Robinson');
writeln(' <M> Modificacion de Mathias');
writeln(' <V> Modificacion de Vera');

```

```

writeln(' <I> Gas ideal o Solucion Ideal');
readln(Ecuacion);
Until (Ecuacion in ['s','S','p','P','m','M','V','v','i','I']);
Repeat
  Clrscr;
  writeln('Las propiedades termodinamicas se van a:');
  writeln(' 1-Leer de un archivo');
  writeln(' 2-Se van a dar');
  readln(OpcionDat);
Until (OpcionDat in ['1','2']);
Case OpcionDat of
  '1' : LeerArch(HfstD,GfstD,Tc,Pc,Omega,pMathias,CP);
  '2' : PedirDat;
end; {case}
if not(Ecuacion in ['I','i']) then
begin
  Clrscr;
  writeln('Diga la regla de mezclado a emplear ');
  writeln('En caso de querer solo un parametro escoger Margules o Van Laar');
  writeln('y dar Kij=Kji');
  writeln(' <M> tipo Margules');
  writeln(' <V> tipo Van Laar');
  writeln(' <N> si no se tiene valoare de Kij ');
  readln(ReglaMez);
end;
if ((ReglaMez in ['n','N']) or (Ecuacion in ['i','I'])) then
begin
  for k:=1 to componentes do
    for kk:=1 to componentes do
      Kaij[k,kk]:=0;
    end {then}
  else begin
    Clrscr;
    for k:=1 to componentes do
      for kk:=1 to componentes do
        begin
          write('K',k,kk,' = ');readln(Kaij[k,kk]);
        end; {for}
      end; {else}
    end; {Inicializa}
  }
}
-----
Procedure Lectura(Var T,P:real;Var NO,Nnueva: Vector1);
Var
  i:integer;
Begin
  Clrscr;
  Writeln('Calculo de composiciones al equilibrio en un sistema multireactivo');
  Writeln('Porcentaje en el que se reduce la concentracion cuando un componente');
  write('limita la convergencia ');readln(paso);

```

```

Write('Temperatura (Kelvin) =');Readln (T);
Write('Presion (atmósferas) =');Readln (P);
Clrscr;
for i:=1 to componentes+Solidos do
begin
  Write('Moles iniciales de ',Nombres[i],' ');Readln (NO[i]);
end;
Clrscr;
for i:=1 to Componentes+Solidos do
  Nnueva[i]:=0;
WRITELN(LST,'REDUCCION DE OXIDO DE FIERRO');
writeln(lst,'Calculo de la composicion al equilibrio por medio del metodo de');
writeln(lst,'la minimizacion de la energia de Gibbs');
for i:=1 to componentes+solidos do
writeln(lst,'Moles iniciales de ',nombres[i],' = ',NO[i]);
writeln(lst,'Temperatura (K) ',T);
writeln(lst,'Presion (atm) ',P);
readln;
Clrscr;
end; {Lectura}

```

```

-----}
Procedure Consteq(T:real;Hfstd,Gfstd:Vector1;CP:Vector2;Var Lkeq:Vector1);

```

```

Var
  i,j :Integer;
Begin
  for j:=1 to Componentes+Solidos do
    Sfstd[j]:=(Hfstd[j]-Gfstd[j])/T0;
    for j:=1 to Componentes+Solidos do
      begin
        Hreac[j]:=Hfstd[j]+CP[j,1]*(T-T0)+CP[j,2]/2*(T*T-T0*T0)+
          CP[j,3]/3*(T*T*T-T0*T0*T0)+CP[j,4]/4*(T*T*T*T-T0*T0*T0*T0)
          -CP[j,5]*(1/T-1/T0);
        Sreac[j]:=Sfstd[j]+CP[j,1]*Ln(T/T0)+CP[j,2]*(T-T0)+
          CP[j,3]/2*(T*T-T0*T0)+CP[j,4]/3*(T*T*T-T0*T0*T0)
          -CP[j,5]*(1/T/T-1/T0/T0);
        Greac[j]:=Hreac[j]-T*Sreac[j];
        Lkeq[j] :=Greac[j]/R/T;
      end; {for}
    for j:=1 to elementos do
      GE1[j]:=(Cpuro[j,1]*(T-T0)+Cpuro[j,2]/2*(T*T-T0*T0)+Cpuro[j,3]/3*
        (T*T*T-T0*T0*T0)+Cpuro[j,4]/4*(T*T*T*T-T0*T0*T0*T0)-Cpuro[j,5]*
        (1/T-1/T0)-T*(Cpuro[j,1]*Ln(T/T0)+Cpuro[j,2]*(T-T0)+Cpuro[j,3]/2
        *(T*T-T0*T0)+Cpuro[j,4]/3*(T*T*T-T0*T0*T0)-Cpuro[j,5]*(1/T/T-1/T0/T0)))
        /R/T;
    for j:=1 to Componentes+Solidos do
      for i:=1 to Elementos do
        Lkeq[j]:=Lkeq[j]-Af[i,j]*GE1[i];
      for j:=1 to Componentes do
        LKeq[j]:=LKeq[j]+Ln(P);
      end;
    end;
  end;

```

```

FOR J:=1 TO COMPONENTES+SOLIDOS DO
WRITELN(LKEQ[J]);READLN;
End; {consteq}

```

(*****)

```

Procedure CalculaB(AI:Vector4;NO:Vector1;Var B:Vector5);

```

```

Var
  i,j,k :integer;
  Traza :char;
  Comp :real;
Begin
  for j:=1 to Elementos do
  begin
    B[j]:=0;
    for i:=1 to Componentes+Solidos do
      B[j]:=B[j]+Ai[j,i]*NO[i];
    end;
  end;
End; {CalculaB}

```

```

{-----}
Procedure Correccion(Lkeq:Vector1;Var Lkeqc:Vector1);

```

```

Var
  i,j :integer;
  Comp :Vector1;
  Molest :real;
begin
  Molest:=0;
  for i:=1 to Componentes do
    Molest:=Molest+Ni[i];
  for i:=1 to Componentes do
    Comp[i]:=Ni[i]/Molest;
  Fugacidades(Componentes,T,P,Ecuacion,Fase,ReglaMez,Tc,Pc,Omega,pMathias,
    Comp,Fuga,Kaij);
  FOR I:=1 TO COMPONENTES DO
  writeln('COEFICIENTE DE FUGACIDAD ',NOMBRES[I],', ',FUGA[I]);
  for i:=1 to Componentes do
    Lkeqc[i]:=Lkeq[i]+ln(fuga[i]);
  end; {correccion}
{-----}

```

```

Procedure Multiplicadores(AI:Vector4;B:Vector5;Lkeq:Vector1;
  Var U:real; Var PI:Vector5);

```

```

Var
  Rlan :Vector6;
  Matriz :Vector3;
  i,j,k,l:integer;
  Ntk :real;

```

```

Procedure InvMatriz(Var Orig:Vector3);

```

```

Var
  HH,H1,
  Renglon,

```

```

i,j,k      :Integer;
Pivote,
Pivote2,
Aux,Mayor  :real;
Begin
for k:=1 to Renglonos do
  Begin
  Mayor:=0;
  for j:=k to Renglonos do
    if abs(Orig[j,k])>abs(Mayor) then
      Begin
        Renglon:=j;
        Mayor:=Orig[j,k];
      End; {if}
  for i:=1 to Columnas do
    Begin
      Aux:=Orig[Renglon,i];
      Orig[Renglon,i]:=Orig[k,i];
      Orig[k,i]:=Aux;
    End; {for}
    Pivote:=Orig[k,k];
    for i:=k to Columnas do
      Orig[k,i]:=Orig[k,i]/Pivote;
    for i:=1 to Renglonos do
      Begin
        Pivote2:=Orig[i,k];
        for j:=k to Columnas do
          if i<k then orig[i,j]:=orig[i,j]-Pivote2*Orig[k,j];
        End; {for i}
      End; {for k}
    End; {InvMatriz}
  }-----}

Begin
Ntk:=0;
for l:=1 to Componentes do
  Ntk:=Ntk+NI[l];
for l:=1 to renglonos do
  for j:=1 to Columnas do
    Matriz[l,j]:=0;
  }elementos ecuaciones}
for l:=1 to Elementos do
  for j:=1 to Elementos do
    Begin
      Rlan[j,l]:=0;
      for I:=1 to Componentes do
        Rlan[j,l]:=Rlan[j,l]+Ai[j,i]*Ai[l,i]*NI[i];
      Matriz[j,l]:=Rlan[j,l];
    End; {for}
  for j:=1 to Elementos do
    for i:=Componentes+1 to Componentes+Solidos do
      Matriz[j,Elementos+i-Componentes]:=Ai[j,i];
  for j:=1 to Elementos do
    begin
      for i:=1 to Componentes do

```

```

    Matriz[j,Columnas-1]:=Matriz[j,Columnas-1]+Ni[i]*Ai[j,i];
Matriz[j,Columnas-1]:=Matriz[j,Columnas-1]/Ntk;
end;
    {solidos Ecuaciones}
for i:=Componentes+1 to Componentes+Solidos do
for j:=1 to Elementos do
    Matriz[Elementos+i-Componentes,j]:=Ai[j,i];
for j:=1 to Elementos do
    Matriz[Reglones,j]:=B[j];
    {ultima columna}
for j:=1 to Elementos do
Begin
    for i:=1 to Componentes do
        if NI[i]>0 then Matriz[j,Columnas]:=Matriz[j,Columnas]+
            Ai[j,i]*NI[i]*(Lkeq[i]+Ln(NI[i]/Ntk));
            Matriz[j,Columnas]:=Matriz[j,Columnas]+B[j];
    End; {for i}
for i:=Componentes+1 to Componentes+Solidos do
    Matriz[Elementos+i-Componentes,Columnas]:=Lkeq[i];
for i:=1 to Componentes do
    if NI[i]>0 then Matriz[Reglones,Columnas]:=Matriz[Reglones,Columnas]+
        NI[i]*(Lkeq[i]+Ln(NI[i]/Ntk));
for i:=Componentes+1 to Componentes+Solidos do
    Matriz[Reglones,Columnas]:=Matriz[Reglones,Columnas]+Ni[i]*Lkeq[i];
InvMatriz(Matriz);
for i:=1 to Elementos do
    PI[i]:=Matriz[i,Columnas];
for i:=Componentes+1 to Componentes+Solidos do
    Nnueva[i]:=Matriz[Elementos+i-Componentes,Columnas];
U:=Matriz[Reglones,Columnas];
End; {Multiplicadores}

{-----}
Procedure Composicion (A:Vector4;Lkeq:Vector1;NI:Vector1;U:real;
    PI:Vector5;Var Nnueva:Vector1);

Var
    i,j :integer;
    Aux,
    NT :real;

Begin
    NT:=0;
    for j:=1 to Componentes do
        NT:=NT+NI[j];
    for i:=1 to Componentes do
        Begin
            Aux:=0;
            for j:= 1 to Elementos do
                Aux:=Aux+PI[j]*A[j,i];
            if NI[i]>0 then Nnueva[i]:=(Aux-Lkeq[i]-Ln(NI[i]/NT)+U/Nt)*NI[i];
        End; {for}
    End; {Composicion}

```

(*****)

Procedure Convergencia(NI,Nnueva:Vector1;Var Incremento:real);

```
Var
  i           :integer;
  NT,
  SumaIncrem :real;

Begin
  NT:=0;
  for i:=1 to Componentes do
    NT:=NT+NI[i];
  Incremento:=0;
  for i:=1 to Componentes+Solidos do
    Begin
      if NI[i]<>0 then SumaIncrem:=(Nnueva[i]-NI[i])/Nnueva[i];
      if Abs(SumaIncrem)>Incremento then Incremento:=Abs(SumaIncrem);
    End; {for}
  End;
```

(*****)

Procedure Actualiza (Lkeq:Vector1;T:real;Var NI,Nnueva:Vector1);

```
Var
  Lam,
  aux   :real;
  i     :integer;
```

Procedure Lambda(Lkeq,NI,Nnueva:Vector1;T:real;Var Lam:real);

```
Var
  contador,
  i,j       :integer;
  Delta     :Vector1;
  MolesT,
  DeltaT,
  ValorFun,
  ValorDer,
  Aux1,Aux2,
  Aux3      :real;
```

Function IterLambda(Lambda:real):real;

```
Var
  k           :integer;
  Sumatoria  :real;
Begin
  Sumatoria:=0;
  for k:=1 to Componentes do
    if Delta[k]>0 then
      Sumatoria:=Sumatoria+Delta[k]*(Lkeq[k]+Ln((NI[k]+Lambda*
        Delta[k])/(MolesT+Lambda*DeltaT)));
  for k:=Componentes+1 to Componentes+Solidos do
```

(*****)

Procedure Convergencia(NI,Nnueva:Vector1;Var Incremento:real);

```
Var
  i           :integer;
  NT,
  SumaIncrem :real;
Begin
  NT:=0;
  for i:=1 to Componentes do
    NT:=NT+NI[i];
  Incremento:=0;
  for i:=1 to Componentes+Solidos do
    Begin
      if NI[i]>0 then SumaIncrem:=(Nnueva[i]-NI[i])/Nnueva[i];
      if Abs(SumaIncrem)>Incremento then Incremento:=Abs(SumaIncrem);
    End; {for}
  End;
```

(*****)

Procedure Actualiza (Lkeq:Vector1;T:real;Var NI,Nnueva:Vector1);

```
Var
  Lam,
  aux           :real;
  i             :integer;
```

Procedure Lambda(Lkeq,NI,Nnueva:Vector1;T:real;Var Lam:real);

```
Var
  contador,
  i,j       :integer;
  Delta     :Vector1;
  MolesT,
  DeltaT,
  ValorFun,
  ValorDer,
  Aux1,Aux2,
  Aux3      :real;
```

Function IterLambda(Lambda:real):real;

```
Var
  k           :integer;
  Sumatoria  :real;
Begin
  Sumatoria:=0;
  for k:=1 to Componentes do
    if Delta[k]>0 then
      Sumatoria:=Sumatoria+Delta[k]*(Lkeq[k]+Ln((NI[k]+Lambda*
        Delta[k])/(MolesT+Lambda*DeltaT)));
  for k:=Componentes+1 to Componentes+Solidos do
```

```

        Sumatoria:=Sumatoria+Delta[k]*Lkeq[k];
IterLambda:=Sumatoria;
End;

Function DerLambda(Lambda:real):real;
Var
    k      :integer;
    Sumatoria :real;
Begin
    Sumatoria:=0;
    for k:=1 to Componentes do
        if Delta[k]>0 then
            Sumatoria:=Sumatoria+Delta[k]*(Delta[k]/(NI[k]+Lambda*Delta[k])-
                DeltaT/(MolesT+Lambda*DeltaT));
        DerLambda:=Sumatoria;
    End;
Begin
    MolesT:=0;
    DeltaT:=0;
    for i:=1 to Componentes do
        Begin
            Delta[i]:=Nnueva[i]-NI[i];
            MolesT:=MolesT+NI[i];
            DeltaT:=DeltaT+Delta[i];
        End; {for}
        for i:=Componentes+1 to Componentes+Solidos do
            Delta[i]:=Nnueva[i]-NI[i];
        Contador:=0;
        for i:=1 to Componentes+Solidos do
            if (Delta[i]<0) then begin
                Contador:=Contador+1;
                if Contador=1 then Lam:=NI[i]/Abs(Delta[i])
                    else begin
                        Aux1:=NI[i]/Abs(Delta[i]);
                        if (Aux1<Lam) then Lam:=Aux1;
                        end; {else}
                    end; {if}
            Aux1:=MolesT+Lam*DeltaT;
            if (Aux1<0) then begin
                Aux1:=MolesT/DeltaT;
                Lam:=Abs(Aux1);
                end;{if}
            Aux1:=Lam/4;Aux3:=Lam;
            if Aux1>0.5 then Aux1:=0.5;
            Repeat
                Lam:=Aux1;
                ValorFun:=IterLambda(Lam);
                ValorDer:=DerLambda(Lam);
                Aux1:=(Lam-ValorFun)/ValorDer;
                Aux2:=Abs((Lam-Aux1)/Lam);
            Until ((Aux2<Tolerancia) or (Aux1>Aux3));
            if (Aux1>Aux3) then Lam:=Paso*Aux3;
            writeln(lst,'LAMBDA ',LAM);
            writeln(lst);

```

```

    readln;
End; {lambda}

Begin
writeln(lst, 'Iteracion ', iteracion);
for i:=1 to Componentes+Solidos do
    writeln(lst, 'NI', nombres[i]:5, '=', NI[i], ' Nueva= ', Nnueva[i]);
readln;
Lambda (Lkeq, NI, Nnueva, T, Lam);
for i:=1 to Componentes+Solidos do
    NI[i]:=NI[i]+Lam*(Nnueva[i]-NI[i]);
End; {Actualiza}

(*****)

Procedure Imprimir;
var
    i, j : integer;
    aux, au: real;
    aux1 : Vector1;
begin
    aux:=0;
    for i:=1 to Componentes do
        aux:=aux+Nnueva[i];
    for i:=1 to Componentes do
        if Nnueva[i]=0 then begin
            au:=0;
            for j:=1 to Elementos do
                au:=au+Pi[j]*Ai[j, i];
            if au>-70 then Nnueva[i]:=-aux*exp(au-Lkeq[i]);
            end;
        writeln(lst, 'ITERACIONES= ', iteracion);
        for i:=1 to Componentes+Solidos do
            writeln(lst, 'Nnueva ', nombres[i]:6, '=', Nnueva[i], ' Nant=', NI[i]);
        readln;
        for i:=1 to Componentes do
            begin
                aux1[i]:=Nnueva[i]/aux;
                writein(lst, 'Y(', nombres[i]:4, ')= ', aux1[i]);
            end;
        readln;
    end; {imprimir}

Begin
Clrscr;
write('Desea crear o modificar archivos <S> o <N> '); readln(OpcionArch);
if (OpcionArch in ['s', 'S']) then Archivos;
Inicializa(Hfstd, Gfstd, Tc, Pc, Omega, pMathias, CP, Ecuacion, ReglaMez, Ai);
Repeat
    Lectura(T, P, NO, Nnueva);
    Consteq(T, Hfstd, Gfstd, CP, Lkeq);
    NI:=NO;
    CalculaB(Ai, NO, Bt);
    Iteracion:=0;

```

```

Repeat
  if not(Ecuacion in ['I','I']) then Correccion(Lkeq,Lkeqc)
  else Lkeqc:=Lkeq;
  Multiplicadores(Ai,Bt,Lkeqc,U,PI);
  Composicion(Ai,Lkeqc,NI,U,PI,Nnueva);
  Iteracion:=Iteracion+1;
  Convergencia(NI,Nnueva,Incremento);
  if Incremento>0.001 then Actualiza(LKeqc,T,NI,Nnueva);
Until (Incremento<0.001);
Imprimir;
Clrscr;
write('Desea hacer otro calculo (S/N)');readln(Calculo);
Until (Calculo in ['n','N']);
End.

```

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

Program CONSTANTES;

(* PROGRAMA PARA CALCULAR LAS COMPOSICIONES AL EQUILIBRIO DE UN REACTOR ISOTERMICO A TRAVES DEL CALCULO DE CONSTANTES DE EQUILIBRIO *)

Uses crt,archi,printer,fugacidad;

Const

```
Componentes=10;      (* numero de especies quimicas involucradas *)
Reacciones =5;      (* reacciones quimicas independientes *)
Solidos =3;         (* numero de especies solidas presentes *)
T0 =298.2;          (* temperatura de referencia (K) *)
R =8.317;           (* cte universal de los gases JOULES/GMOL/K *)
Tolerancia =1E-3;  (* tolerancia en las iteraciones *)
```

Type

```
Vector2 =Array [1..Reacciones] of real;
Vector3 =Array [1..Componentes+Solidos,1..5] of real;
Vector4 =Array [1..Reacciones,1..Componentes+Solidos] of integer;
Vector6 =Array [1..Reacciones,1..2] of real;
Vector7 =Array [1..Componentes+Solidos] of string[8];
Vector8 =Array [1..Componentes+Solidos] of boolean;
Vectorc =Array [1..5] of string[2];
Vectorn =Array [1..2,1..5] of real;
Archivo =string[12];
Datos =record
    Nombre :string[8];
    Entalpia:real;
    EnergiaG:real;
    CaCal :array[1..4] of real;
    TempC :real;
    PresC :real;
    Factor :real;
    CoefMat :real;
    Comp :Vectorc;
    Co :Vectorn;
end;
```

Var

```
Tc, (* temperatura critica *)
Pc, (* presion critica *)
Omega, (* factor acentrico *)
pMathias, (* coeficiente Mathias *)
Hfstd, (* entalpia de formacion a T0 (JOULES/GMOL) *)
Gfstd, (* energia libre de form a T0 (JOULES/GMOL) *)
Fuga, (* fugacidades *)
Sreac, (* propiedades para la reaccion J a la temp *)
Hreac, (* *)
Greac, (* del reactor *)
NO :Vector1; (* composicion de la alimentacion *)
Keq, (* Constantes de equilibrio a temp reactor *)
KeqReal, (* constantes de equilibrio considerando *)
X, (* fugacidades *)
Xnueva :Vector2; (* grado de avance de reaccion *)
(* nuevo grado de avance de reaccion *)
```

```

CP      :Vector3;      (* capacidades calorificas (JOULES/GMOL/K) *)
Coef   :Vector4;      (* coeficientes estequiometricos *)
NN      :Vector6;      (* ordenador de las constantes de equilibrio *)
Nombres:Vector7;      (* nombres de los compuestos reaccionantes *)
Solido :Vector8;      (* cuales son solidos *)
Kaij   :Vectork;      (* parametros de interaccion *)
Iteracion,
Numero,
i,j,k  :integer;      (* reaccion sobre la que se converge *)
Error,
Errortotal,
T,
P      :real;          (* contadores *)
Nombreach :Archivo;   (* error en el grado de avance *)
Nombrelog :file of Datos; (* error mayor *)
Registro :Datos;      (* temperatura (K) *)
ReglaMez,
Fase,
Ecuacion,
OpcionArch :char;    (* presion del reactor (Atm) *)

```

{-----}

```
Function Pot(base:real;exponente:real):real;
```

```

Begin
  if base=0 then pot:=0 else pot:=exp(exponente*Ln(base));
End; {pot}
      (*****)
```

```

Procedure Inicializa (Var Hfstd,Gfstd,Tc,Pc,Omega,pMathias:Vector1;
                     Var CP:Vector3;Var Coef:Vector4;Var X:Vector2;
                     Var Ecuacion:char);
```

```

Var
  k,kk,k2,k3 :integer;
  OpcionDat :char;
```

```

Procedure LeerArch(Var Hfstd,Gfstd,Tc,Pc,Omega,pMathias:Vector1;
                  Var CP:Vector3;Nombres:Vector7);
```

```

Var
  i,k,j,l :integer;
  NombreTemp :string[8];
Begin
  Clrscr;
  write('Nombre del archivo a leer ');readln(Nombreach);
  NOMBREARCH:= 'PRUEBA';
  Assign(Nombrelog,Nombreach);
  Reset(Nombrelog);
  i:=FileSize(Nombrelog);
  for k:=1 to Componentes+Solidos do
  begin
    NombreTemp:=Nombres[k];
    j:=0;
    Reset(Nombrelog);

```

```

repeat
  j:=j+1;
  Read(Nombrelog,Registro);
until (registro.nombre=NombreTemp) or (j=i);
if registro.nombre<>NombreTemp then
  writeln('Las propiedades de ',NombreTemp,' no estan en ',Nombrearch)
else
  begin
  Clrscr;
  with Registro do
  begin
  writeln(nombres[k]);
  writeln('Entalpia ',entalpia);Hfstd[k]:=entalpia;
  writeln('Energia libre ',energiag);Gfstd[k]:=energiag;
  for l:=1 to 5 do
  begin
    CP[k,l]:=CaCal[l];
    writeln('Coeficiente ',l,' del CP ',CaCal[l]);
  end; {for}
  if k<Componentes then
  begin
    writeln('Temperatura critica ',TempC);Tc[k]:=TempC;
    writeln('Presion critica ',PresC);Pc[k]:=PresC;
    writeln('Factor acentrico ',Factor);Omega[k]:=Factor;
    if (Ecuacion in ['m','M'])
    then begin
      writeln('Factor Mathias ',CoefMat);pMathias[k]:=CoefMat;
    end {then}
    else pMathias[k]:=0;
  end; {if}
  GotoXY(20,15);
  write('Oprima cualquier tecla para continuar');readln;
  end; {with}
  end; {else}
  end; {for}
Close(Nombrelog);
End; {Lectura}

Procedure PedirDat;
Var
  i,k      :integer;
  NombreTemp :string[8];
Begin
  for i:=1 to componentes do
  begin
  Clrscr;
  writeln(nombres[i]);
  write('Entalpia (Joules/gmol) de formacion a ',T0,' ');readln(Hfstd[i]);
  write('Energia libre de formacion (Joule/gmol) ');readln(Gfstd[i]);
  writeln('CP de la forma a1+a2*T+a3*T^2+a4*T^3 (Joule/gmol/K)');
  for j:=1 to 4 do
  begin
    write('a',j,'? ');readln(CP[i,j]);
  end; {for}
  if not(Ecuacion in ['I','i']) and (i<Componentes) then
  begin

```

```

write('Temperatura critica (K) ');readln(tc[i]);
write('Presion critica (atm) ');readln(Pc[i]);
write('Factor acentrico ');readln(Omega[i]);
if (Ecuacion in ['m','M'])
then begin
    write('Factor de Mathias ');readln(pMathias[i]);
    end {then}
    else pMathias[i]:=0;
end; {if}
end; {for}
End; {PedirDat}

Begin
Clrscr;
writeln('Compuestos presentes al equilibrio');
writeln('Nombre de los compuestos en forma de formula condensada');
writeln('en orden alfabetico Ej.O2S, C3H4O');
writeln('Dar al final los compuestos solidos');
for k:=1 to Componentes+Solidos do
begin
    write('Compuesto ',k,' ');readln(Nombres[k]);
end; {for}
Clrscr;
writeln('Fase de reaccion <L> liquido');
writeln('          <G> gas ');
readln(Fase);}
Repeat
Clrscr;
writeln('Ecuacion de Estado a usar');
writeln('  <S> Soave');
writeln('  <P> Peng-Robinson');
writeln('  <M> Modificacion de Mathias');
writeln('  <I> Gas ideal ');
readln(Ecuacion);
Until (Ecuacion in ['s','S','p','P','m','M','i','I']);}
Repeat
Clrscr;
writeln('Las propiedades termodinamicas se van a:');
writeln('  1-Leer de un archivo');
writeln('  2-Se van a dar');
readln(OpcionDat);
Until (OpcionDat in ['1','2']);}
Case OpcionDat of
    '1' : LeerArch(Hfstd,Gfstd,Tc,Pc,Omega,pMathias,CP,Nombres);
    '2' : PedirDat;
end; {case}
if not(Ecuacion in ['I','i']) then
begin
    Clrscr;
    writeln('Diga la regla de mezclado a emplear ');
    writeln('En caso de querer solo un parametro escoger Margules o Van Laar');
    writeln('y dar Kij=Kji');
    writeln('  <M> tipo Margules');
    writeln('  <V> tipo Van Laar');
    writeln('  <N> si no se tiene valore de Kij ');
    readln(ReglaMez);

```

```

end;
if ((ReglaMez in ['n','N']) or (Ecuacion in ['I','I'])) then
begin
for k:=1 to componentes do
for kk:=1 to componentes do
Kaij[k,kk]:=0;
end {then}
else begin
Clrscr;
for k:=1 to componentes do
for kk:=1 to componentes do
begin
write('K',k,kk,' = ');readln(Kaij[k,kk]);
end; {for}
end; {else}
for k:=1 to reacciones do
for kk:=1 to componentes+Solidos do
Coef[k,kk]:=0;
for k:=1 to componentes+Solidos do
begin
Clrscr;
writeln('Indique en que reacciones aparece cada compuesto');
writeln('Los coeficientes para reactivos son negativos');
writeln('Compuesto ',nombres[k]);
write('Aparece en cuantas reacciones ');readln(k2);
for kk:=1 to k2 do
begin
write('Reaccion ');readln(k3);
write('Coeficiente ');readln(Coef[k3,k]);
end; {for}
end; {for}
for i:=1 to reacciones do
X[i]:=0;
End; {Inicializa}

```

(*****)

```

Procedure Lectura(Var T,P:real;Var NO: Vector1);

```

```

Begin
Clrscr;
writeln('Calculo de composiciones al equilibrio en un sistema multireactivo');
writeln;
Write('Temperatura (Kelvin) =');Readln (T);
Write('Presion (atmósferas) =');Readln (P);
Clrscr;
for i:=1 to componentes do
begin
Write('Moles iniciales de ',Nombres[i], ' ');Readln (NO[i]);
end; {for}
Clrscr;
WRITELN(LST,'REFORMACION DE GAS NATURAL');
WRITELN(LST,'EJEMPLO DE SMITH Y VAN NESS');
writeln(LST,'Calculo del equilibrio quimico por medio de constantes de equilibrio
writeln(LST,'Temperatura= ',T);

```

```
writeln(LST,'Presion= ',P);
for i:=1 to Componentes+Solidos do
  writeln(LST,'Moles iniciales de ',Nombres[i],' ',NO[i]);
readln;
end; {Lectura}
```

(*****)

```
Procedure Consteq(T:real; Hfstd,Gfstd:Vector1;CP:Vector3;Coef:Vector4;
  Var Keq:Vector2);
```

```
Var
  i,j,k :integer;
  Sfstd :Vector1;
Begin
  for j:=1 to reacciones do
    Keq[j]:=0;
  for j:=1 to Componentes+Solidos do
  begin
    Sfstd[j]:=(Hfstd[j]-Gfstd[j])/TO;
    Hreac[j]:=Hfstd[j]+CP[j,1]*(T-TO)+CP[j,2]/2*(T*T-TO*TO)+
      CP[j,3]/3*(T*T*T-TO*TO*TO)+CP[j,4]/4*(T*T*T*T-
      TO*TO*TO*TO)-CP[j,5]*(1/T-1/TO);
    Sreac[j]:=Sfstd[j]+CP[j,1]*ln(T/TO)+CP[j,2]*(T-TO)+
      CP[j,3]/2*(T*T-TO*TO)+CP[j,4]/3*(T*T*T-
      TO*TO*TO)-CP[j,5]*(1/T-1/TO/TO);
    Greac[j]:= Hreac[j]-T*Sreac[j];
  end; {for}
  for j:=1 to reacciones do
  begin
    for k:=1 to componentes+Solidos do
      Keq[j]:=Keq[j]+Coef[j,k]*Greac[k];
    Keq[j]:=Exp(-Keq[j]/R/T);
  end;
  for j:=1 to reacciones do
    writeln(LST,'Equilibrio Reaccion ',j,' ',Keq[j]:4:2);
  end; {ConstEq}
```

(*****)

```
Procedure Ordena(Keq:Vector2;Var NN:Vector6;Coef:Vector4);
```

```
Var
  i,j : integer;
  aux : real;
Begin
  for i:=1 to Componentes+Solidos do
    if Solido[i] then for j:=1 to Reacciones do Coef[j,i]:=0;
  for i:=1 to Reacciones do
  begin
    NN[i,1]:=i;
    NN[i,2]:=Keq[i];
  end; {for}
  for i:=1 to (Reacciones-1) do
    for j:=reacciones downto (i+1) do
```

```

begin
if NN[j,2]>NN[j-1,2] then begin
aux:=NN[j,2];
NN[j,2]:=NN[j-1,2];
NN[j-1,2]:=aux;
aux:=NN[j,1];
NN[j,1]:=NN[j-1,1];
NN[j-1,1]:=aux;
end; {for}
End; {ordena}
(*****

Procedure Correccion(Var KeqReal:Vector2;Keq:Vector2;Numero:integer);
Var
i,j      :integer;
NT       :real;
Y,NI     :Vector1;
begin
NT:=0;
for i:=1 to componentes do
begin
NI[i]:=NO[i];
for j:=1 to reacciones do
begin
NI[i]:=NI[i]+Coef[j,i]*X[j];
end; {for}
NT:=NT+NI[i];
end; {for}
for i:=1 to componentes do
Y[i]:=NI[i]/NT;
Fugacidades(Componentes,T,P,Ecuacion,Fase,ReglaMez,Tc,Pc,Omega,pMathias,
Y,Fuga,Kaij);
KeqReal[Numero]:=Keq[Numero];
for i:=1 to componentes do
KeqReal[numero]:=KeqReal[Numero]*Pot(fuga[i],Coef[Numero,i]);
end; {correccion}
(*****

Procedure ReaccionSola (Numero:integer;NO:Vector1;
Var X:Vector2;Keq:Vector2; P:real );
Var
Contador,
Contador2,
i,j      :integer;
CoefReac :Vector1;
Xmin,
Xmax,
FuncionX,
DerFuncionX,
Xiteracion,
Xpossible,
aux3,aux4 :real;
(* coeficientes de reaccion *)
(* limites de avance de *)
(* reaccion *)
(* ecuacion de equilibrio *)
(* derivada ecuacion equil *)
(* variable iteracion *)

```

```
Function FuncionEq(P:real;X:Vector2;KonstEq:real;CoefReac:Vector1):real;
```

```
Var
```

```
  k,l      :integer;  
  Aux1,  
  Aux2,  
  NT       :real;   (* moles totales *)  
  NI,Y     :Vector1; (* moles de c/componente y composiciones *)
```

```
begin
```

```
  NT:=0;  
  for l:=1 to componentes do  
    begin  
      NI[L]:=NO[L];  
      for k:=1 to reacciones do  
        begin  
          NI[l]:=NI[l]+Coef[k,l]*X[k];  
        end; {for}  
      NT:=NT+NI[L];  
    end; {for}  
  for l:=1 to componentes do  
    Y[l]:=NI[l]/NT; (* COMPOSICIONES *)
```

```
  Aux1:=1;  
  Aux2:=1;  
  for k:=1 to componentes do  
    begin  
      if CoefReac[k]>0 then Aux1:=Aux1*Pot(P*Y[k],CoefReac[k]);  
      if CoefReac[k]<0 then Aux2:=Aux2*Pot(P*Y[k],-CoefReac[k]);  
    end; {for}  
  FuncionEq:=Aux1-Aux2*KonstEq;  
end; {funcion}
```

```
begin
```

```
  for i:=1 to componentes do  
    CoefReac[i]:=Coef[Numero,i];  
  Contador:=0;Contador2:=0;  
  for i:=1 to componentes do  
    begin  
      Aux3:=0;Aux4:=0;  
      if CoefReac[i]<0 then  
        begin  
          Contador:=Contador+1;  
          for j:=1 to Reacciones do  
            if j>Numero then Aux3:=Aux3+Coef[j,i]*X[j];  
            Aux3:=- (Aux3+NO[i])/Coef[Numero,i];  
            if Contador=1 then Xmax:=Aux3;  
            if Aux3<Xmax then Xmax:=Aux3;  
          end; {if}  
        end;  
      if CoefReac[i]>0 then  
        begin  
          Contador2:=Contador2+1;  
          for j:=1 to Reacciones do  
            if j>Numero then Aux4:=Aux4+Coef[j,i]*X[j];  
            Aux4:=- (Aux4+NO[i])/Coef[Numero,i];
```

```

    if Contador2=1 then Xmin:=Aux4;
    if Aux4>Xmin then Xmin:=Aux4;
end; {if}
end; {for}
      (* inicia Newton - Ramphson *)

```

```

if Xmax<>Xmin then
begin
  Xiteracion:=(Xmax+Xmin)/2;
  if Xiteracion=0 then Xiteracion:=(2*Xmax+Xmin)/3;
  Repeat
    Xnueva[Numero]:=Xiteracion;
    Xposible:=Xnueva[Numero]*1.0001;
    if (Xposible>Xmax) or (Xnueva[Numero]<Xmin)
      then Xnueva[Numero]:=(Xmin+Xmax)/2;
    FuncionX:=FuncionEq(P,Xnueva,Keq[Numero],CoefReac);
    if FuncionX<0 then Xmin:=Xnueva[Numero]
      else Xmax:=Xnueva[Numero];
    Xnueva[Numero]:=Xnueva[Numero]*1.0001;
    DerFuncionX:=(FuncionEq(P,Xnueva,Keq[Numero],CoefReac)-FuncionX)
      /(Xnueva[numero]/1.0001*0.0001);
    Xnueva[Numero]:=Xnueva[Numero]/1.0001;
    Xiteracion:=Xnueva[Numero]-FuncionX/DerfuncionX;
  until (Abs((Xiteracion-Xnueva[Numero])/Xiteracion))<Tolerancia;
  end {then}
  else Xnueva[Numero]:=Xmax;
end;{ReaccionSola}
      (*****

```

```

Procedure Imprime;

```

```

Var
  l,j      :integer;
  NT       :real;
  Nu,Y     :Vector1;

```

```

Begin
  Clrscr;
  Writeln(LST,'          R E S U L T A D O S');
  Writeln;
  Writeln(LST,'Iteraciones realizadas = ',Iteracion);
  for J:=1 to Reacciones do
    Writeln(LST,'Reaccion ',J,' :   Xanterior =',X[J],'   Xnueva =',Xnueva[J]);
  NT:=0;
  for l:=1 to Componentes do
    begin
      Nu[l]:=N0[l];
      for K:=1 TO REACCIONES do
        Nu[l]:=Nu[l]+Coef[k,l]*Xnueva[k];
      NT:=NT+Nu[l];
    end; {for}
  Writeln;
  Writeln(LST,'Composiciones finales :');
  FOR L:=1 TO COMPONENTES DO
  WRITELN(lst,'Moles de ',nombres[l],' ',Nu[l]);
  for l:=1 to Componentes do

```

```

begin
  Y[1]:=Nu[1]/NT;
  Writeln (LST,'
end; {for}
Readln;
end; {imprime}

```

(*****)

```

Begin
  Clrscr;
  Write('Desea crear o modificar archivos (S/N)');readln(OpcionArch);
  if (OpcionArch in ['s','S']) then Archivos;
  Inicializa(Hfstd,Gfstd,Tc,Pc,Omega,pMathias,CP,Coef,X,Ecuacion);
  Lectura(T,P,NO);
  Consteq(T,Hfstd,Gfstd,CP,Coef,Keq);
  Ordena(Keq,NN,Coef);
  Iteracion:=0;
  Xnueva:=X;
  Repeat
    Iteracion:=Iteracion+1;
    GotoXY(10,15);
    Writeln('realizando iteracion # ',iteracion);
    X:=Xnueva;
    for j:=1 to Reacciones do
      begin
        Numero:=Trunc(NN[j,1]);
        if not(Ecuacion in ['i','I']) then Correccion(KeqReal,Keq,Numero)
          else KeqReal:=Keq;
        ReaccionSola (Numero,NO,Xnueva,KeqReal,P);
      end; {for}
    ErrorTotal:=0;
    writeln(lst,'Iteracion ',iteracion);
    for j:=1 to reacciones do
      writeln(lst,' X reaccion ',j,' = ',Xnueva[j]);
    readln;
    for j:=1 to Reacciones do
      begin
        if Xnueva[J]=0 then error:=Abs(X[j]-Xnueva[j])
          else error:=Abs((X[j]-Xnueva[j])/Xnueva[j]);
        if error>Errorrtotal then ErrorTotal:=Error;
      end; {for}
    Until ErrorTotal<Tolerancia;
    Imprime;
  end.

```

```

Program Fugacidad;
Uses crt;
Const
  Compuestos =3;
Type
  Vector1=Array[1..Compuestos] of real;
  Vectork=Array[1..Compuestos,1..Compuestos] of real;
Var
  Tc,Pc,Omega,
  pMathias,
  X,
  fuga,
  B           :Vector1;
  A,
  Kij        :Vectork;
  Temporal,
  Z,Zmc,
  dm,Amc,
  Bm,Am,
  Bd,ymc,
  T,P,
  u,w,
  Aprima,
  Corr,
  Coef1,Coef2,
  Coef3      :real;
  i          :integer;
  Ecuacion,Fase,
  ReglaMez  :char;

Procedure Inicio;
Var i,j:integer;
begin
  Tc[1]:=305.4;Pc[1]:=48.2;Omega[1]:=0.091;
  Tc[2]:=369.8;Pc[2]:=41.9;Omega[2]:=0.145;
  Tc[3]:=562.1;Pc[3]:=48.3;Omega[3]:=0.21;
  for i:=1 to Compuestos do
  begin Pmathias[i]:=0;
  for j:=1 to Compuestos do
  Kij[i,j]:=0;
  end;
  T:=200;
  P:=1;
  X[1]:=0.2;X[2]:=0.3;X[3]:=0.5;
  Ecuacion:='S';
  Fase:='L';
  ReglaMez:='M';

```

end;

{-----}

```
Procedure Empieza(Ecuacion,ReglaMez:char;Tc,Pc,Omega,pMathias,  
X:Vector1;Var Bm,Am,Coef1,Coef2,  
Coef3:real;Kij:Vectork;Var A:Vectork;Var B:Vector1;  
T,P:real;Compuestos:integer);
```

type

Arreglo =Array[1..4] of real;

var

m,alfa :Vector1;

C :Arreglo;

aux1,aux2,

SigmaA,

SigmaB :real;

i,j :integer;

Begin

if (Ecuacion in ['s','S','m','M']) then

begin

u:=1;

w:=0;

SigmaA:=0.42748023;

SigmaB:=0.086640350;

end {then}

else

begin

u:=2;

w:=-1;

SigmaA:=0.45723553;

SigmaB:=0.077796074;

end; {else}

Case Ecuacion of

's','S':begin

c[1]:=0.480;

c[2]:=1.574;

c[3]:=-0.176;

c[4]:=0;

end;

'p','P':begin

c[1]:=0.37464;

c[2]:=1.54226;

c[3]:=-0.26992;

c[4]:=0;

end;

'm','M':begin

c[1]:=0.48508;

c[2]:=1.55191;

c[3]:=-0.15613;

c[4]:=0;

end;

'v','V':begin

c[1]:=0.378893;

c[2]:=1.4897153;

```

c[3]:=-0.17131848;
c[4]:=0.0196554;
end;
end;
for i:=1 to Compuestos do
begin
m[i]:=c[1]+c[2]*Omega[i]+c[3]*Omega[i]*Omega[i]+c[4]
*Omega[i]*Omega[i]*Omega[i];
alfa[i]:=Sqr(1+m[i]*(1-Sqrt(T/Tc[i]))-pMathias[i]*(1-T/Tc[i])*(0.7-T/Tc[i]));
aux1:=T/Tc[i];
if ((Ecuacion in ['m', 'M']) and (aux1>1)) then
begin
aux2:=1+m[i]/2+0.3*pMathias[i];
alfa[i]:=Sqr(exp((aux2-1)/aux2*(1-exp(aux2*ln(aux1)))));
end;
A[i,i]:=SigmaA*P/Pc[i]*Tc[i]*Tc[i]/T*alfa[i];
B[i]:=SigmaB*P/Pc[i]*Tc[i]/T;
end; {for}
Bm:=0;Am:=0;
for i:=1 to Compuestos do
begin
Bm:=X[i]*B[i]+Bm;
for j:=1 to (i-1) do
begin
if (ReglaMez in ['v', 'V']) then
A[i,j]:=Sqr(A[i,i]*A[j,j])*(1-Kij[i,j]*Kij[j,i]/(X[i]*
Kij[i,j]+X[j]*Kij[j,i]))
else
A[i,j]:=Sqr(A[i,i]*A[j,j])*(1-X[i]*Kij[i,j]-
X[j]*Kij[j,i]);
A[j,i]:=A[i,j];
end; {for}
end; {for}
for i:=1 to Compuestos do
for j:=1 to Compuestos do
Am:=Am+X[i]*X[j]*A[i,j];
Coef1:=u*Bm-Bm-1;
Coef2:=Am+u*Bm*Bm-u*Bm*Bm-u*Bm;
Coef3:=-w*Bm*Bm-Bm-w*Bm*Bm-Am*Bm;
End; {Empieza}

{-----}

Procedure Zeta(Coef1,Coef2,Coef3,Bm:real;Var Z:real;Fase:char);
Var
p,q,d :real;

Procedure CalculoAux(Var Zmc,ymc,Amc,dm:real);
Var
Fy,DFy,
yaux,
ynue :real;
Begin

```

```

ynue:=3+0.85*u+0.75*w;
repeat
  ymc:=ynue;
  Fy:=ymc*ymc*ymc-3*ymc*ymc-3*ymc*(u+w)-u*(u+w)+w;
  DFy:=3*ymc*ymc-6*ymc-3*(u+w);
  ynue:=ymc-Fy/DFy;
  yaux:=Abs((ymc-ynue)/ynue);
until (yaux<1e-5);
Zmc:=Bm*ymc;
Amc:=Sqr(ymc*ymc+u*ymc+w)/((ymc-1)*(ymc-1)*(2*ymc+u));
dm:=1/ymc;
End; {CalculoAux}

Function CalculaBd(d:real):real;
Begin
  CalculaBd:=1/Sqr(1-d)-Am/Bm*(2*d+u*d*d)/Sqr(1+u*d+w*d*d);
End;

Procedure Liquivapl(var Z:real);
Var
  M,N :real;
Function Cubica(base:real):real;
Begin
  if (base<0) then
    begin
      base:=-base;
      base:= exp(ln(base)/3);
      Cubica:=-base;
    end {then}
  else
    Cubica:= exp(ln(base)/3);
  end; {cubica}
begin
  M:=-q/2+Sqr(d);
  N:=Cubica(M);
  M:=-q/2-Sqr(d);
  N:=Cubica(N);
  Z:=M+N-Coeff1/3;
end; {liquidol}

Procedure Liquido2(Var Z1:real);
Var
  Fi,arco :real;
Begin
  Arco:=Sqr(-27*q*q/4/p/p/p);
  Fi:=-ArcTan(Arco/Sqr(1-Arco*Arco))+1.570796;
  if q>0 then Z1:=-2*Sqr(-p/3)*cos(Fi/3)-Coef1/3
    else Z1:=2*Sqr(-p/3)*cos(Fi/3+2*Pi/3)-Coef1/3;
  if (Z1<=Bm) then if q<0 then Z1:=2*Sqr(-p/3)*cos(Fi/3)-Coef1/3
    else Z1:=-2*Sqr(-p/3)*cos(Fi/3+2*Pi/3)-Coef1/3;
End; {liquido2}

Procedure Vapor2(Var Zv:real);
Var
  Fi,arco :real;

```

```

Begin
  Arco:=Sqrt(-27*q*q/4/p/p/p);
  Fi:=-ArcTan(Arco/Sqrt(1-Arco*Arco))+1.570796;
  if q>0 then Zv:=-2*Sqrt(-p/3)*cos(Fi/3+2*pi/3)-Coef1/3
    else Zv:=2*Sqrt(-p/3)*cos(Fi/3)-Coef1/3;
End; {vapor2}

Procedure Extremo(Var d,Bd,Bdext:real);
Var
  i      :integer;
  Dnueva,
  Error,
  Bdaux:real;
Begin
  i:=0;
  if (Fase in ['L','L']) then Dnueva:=0.5
    else Dnueva:=0.1;
  Repeat
    d:=Dnueva;
    i:=i+1;
    Bd:=CalculaBd(d);
    Bd:=Bd-0.1;
    Bdaux:=2/((1-d)/(1-d)/(1-d)-(Am/Bm)*2*(1-w*d*d*(3+u*d))/(1+u*d+w*d*d))
      /(1+u*d+w*d*d)/(1+u*d+w*d*d);
    Dnueva:=d-Bd/Bdaux;
    if (Fase in ['v','V']) then begin
      if Dnueva<0 then Dnueva:=(Dnueva+Bd/Bdaux)/2;
      if Dnueva>Dm then Dnueva:=(Dnueva+Bd/Bdaux+Dm)/2;
      end else begin
      if Dnueva>1 then Dnueva:=(Dnueva+Bd/Bdaux+1)/2;
      if Dnueva<Dm then Dnueva:=(Dnueva+Bd/Bdaux+Dm)/2;
      end;
  Until (i>25) or (Abs(Bd)<1e-4);
  D:=Dnueva;
  if (i>25) or (D>1) or (D<0) then D:=dm;
  Bd:=CalculaBd(D);
  Bdext:=d/(1-d)-Am/Bm*d*d/(1+u*d+w*d*d);
end; {extremo}

Procedure ExtrapoLiq(Var Zl:real);
Var
  Bdext,
  Cc1,Cc0,
  din,dex :real;
begin
  Bd:=CalculaBd(dm);
  if ((Am/Bm)<Amc) and (Bd>0.1) then
    begin
      dex:=dm;
      Bdext:=dm/(1-dm)-Am/Bm*dm*dm/(1+u*dm+w*dm*dm)
    end
    else Extremo(dex,Bd,Bdext);
  Cc1:=Bd*(dex-0.7*dm);
  Cc0:=Bdext-Cc1*ln(Dex-0.7*dm);
  din:=exp((Bm-Cc0)/Cc1)+0.7*dm;

```

```

Z1:=Bm/din;
Corr:=(din/(1-din)-Am/Bm*din*din/(1+u*din+w*din*din))/Bm;
end; {Extrapoliq}

```

```

Procedure ExtrapoVap(Var Zv:real);

```

```

Var
  Bdext,Cc2,
  Cc1,Cc0,
  aux,
  din,dex :real;

```

```

begin
  dex:=0.1;
  Extremo(dex,Bd,Bdext);
  aux:=(dex+dm)/2;
  Cc2:=(Bd*(dm-dex)/2-Bdext)/Sqr(Bdext*(dm-dex)/2);
  Cc1:=2*Cc2*dex+Bd/Bdext/Bdext;
  Cc0:=1/Bdext+Cc1*dex-Cc2*dex*dex;
  din:=(Cc1-Sqrt(Cc1*Cc1-4*Cc2*(Cc0-1/Bm)))/2/Cc2;
  Zv:=Bm/din;
  Corr:=1;
end; {ExtrapoVap}

```

```

Begin

```

```

  p:=Coef2-Coef1*Coef1/3;
  q:=Coef3-Coef1*Coef2/3+2/27*Coef1*Coef1*Coef1;
  d:=p*p*p/27+q*q/4;
  If (fase in ['L','L']) then
  begin
    if (d>=0) then liquivap1(Z)
      else liquido2(Z);
    CalculoAux(Zmc,ymc,Amc,dm);
    Bd:=CalculaBd(Bm/Z);
    if (Z>Zmc) or (Bd<0.1) then Extrapoliq(Z)
      else Corr:=1;
  end {then}
  else
  begin
    CalculoAux(Zmc,ymc,Amc,dm);
    if (d>=0) then liquivap1(Z)
      else Vapor2(Z);
    Temporal:=Am/Bm;
    Bd:=CalculaBd(Bm/Z);
    if not((Temporal<Amc) or ((Z>Zmc) and (Bd>0.1))) then ExtrapoVap(Z)
      else Corr:=1;
  end; {else}
End; {zetas}

```

```

-----
Procedure Termina(A:Vectork;B,X:Vector1;Z,Am,Bm:real;
  Var Fuga:Vector1;Ecuacion,ReglaMez:char;Compuestos:integer);
Var
  Landa,
  aux :real;

```

```

i      :integer;

Procedure VanLaar(Var aux:real;i:integer);
Var
  j,m  :integer;
begin
  aux:=0;
  for j:=1 to compuestos do
    begin
      aux:=aux+2*X[j]*A[i,j];
      if j<>i then aux:=aux-2*X[i]*X[j]*Sqrt(A[i,i]*A[j,j])
        *Kij[i,j]*Kij[j,i]*(X[i]*Kij[i,j]+X[j]*
          Kij[j,i]-Kij[i,j])/Sqr(X[i]*Kij[i,j]+X[j]
            *Kij[j,i]);
      for m:=(j+1) to Compuestos do
        if (j<>i) and (m<>i) then aux:=aux+2*X[j]*X[m]*Sqrt(A[j,j]*
          A[m,m])*Kij[j,m]*Kij[m,j]/(X[j]*Kij[j,m]
            +X[m]*Kij[m,j]);
    end; {for}
  end; {VanLaar}

Procedure Margules(Var aux:real;i:integer);
Var
  j,m  :integer;
begin
  aux:=0;
  for j:=1 to compuestos do
    begin
      aux:=aux+2*X[j]*A[i,j];
      if j<>i then aux:=aux+2*X[i]*X[j]*Sqrt(A[i,i]*A[j,j])*
        (X[i]*Kij[i,j]+X[j]*kij[j,i]-Kij[i,j]);
      for m:=(j+1) to Compuestos do
        if (j<>i) and (m<>i) then aux:=aux+2*X[j]*X[m]*Sqrt(
          A[j,j]*A[m,m])*(X[j]*Kij[j,m]+X
            [m]*Kij[m,j]);
    end; {for}
  end; {margules}

Begin
Landa:=ln((Z+bm*(u-Sqrt(u*u-4*w))/2)/(z+bm*(u+Sqrt(u*u-4*w))/2))/Sqrt(u*u-4*w);
for i:=1 to Compuestos do
  begin
    if (ReglaMez in ['v','V']) then VanLaar(aux,i)
      else Margules(aux,i);
    fuga[i]:=B[i]/Bm*(Z-1)-ln(Z-Bm)+Am/Bm*(aux/Am-B[i]/Bm)*Landa;
    fuga[i]:=exp(fuga[i])/Corr;
  end; {for}
End; {termina}
{-----}

BEGIN
Inicio;
Empieza(Ecuacion,ReglaMez,Tc,Pc,Omega,pMathias,X,Bm,Am,
  Coef1,Coef2,Coef3,Kij,A,B,T,P,Compuestos);
Zeta(Coef1,Coef2,Coef3,Bm,Z,Fase);

```

```
WRITELN(Z);READLN;  
Termina(A,B,X,Z,Am,Bm,Fuga,Ecuacion.ReglaMez,Compuestos);  
for i:=1 to Compuestos do  
  writeln(fuga[i]);readln;  
End.
```

Unit Archi;

Interface
uses Crt;

Type

```
Vectorc =Array [1..5] of string[2];  
Vectorn =Array [1..2,1..4] of real;  
Archivo =string[12];  
Datos =record  
    Nombre :string[8];  
    Entalpia:real;  
    EnergiaG:real;  
    CaCal :array[1..5] of real;  
    TempC :real;  
    PresC :real;  
    Factor :real;  
    CoefMat :real;  
    Comp :Vectorc;  
    Co :Vectorn;  
end;
```

```
Var  
    i,j,k :integer; (* contadores *)  
    Nombreach :Archivo; (* variables para el manejo *)  
    Nombrelog :file of Datos; (* de archivos *)  
    Registro :Datos;
```

Procedure Archivos;

Implementation
Procedure ARCHIVOS;

```
Var  
    Cantidad :integer;  
    Opcion,  
    Cambio,  
    Operacion :char;  
    Compuesto :string[8];
```

Procedure GuardaDatos;

```
Var  
    i,j,k,l :integer;  
Begin  
    Clrscr;  
    write('Nombre del archivo a usar ');readln(Nombreach);  
    Assign(Nombrelog,Nombreach);  
    writeln('Es nuevo el archivo <S>');  
    writeln('Agregando datos a uno ya existente <A>');  
    readln(operacion);  
    If operacion in ['S','s'] then  
        Rewrite(Nombrelog)
```

```

else
begin
  Reset(Nombrelog);
  Seek(Nombrelog,FileSize(Nombrelog));
end; {else}
Clrscr;
writeln('Guardando datos en el archivo ',Nombrearch);
writeln;
writeln('El nombre del compuesto darlo como formula condensada en orden');
writeln('alfabetico. Ej: C4H30, O2S');
writeln('La entalpia de formacion y energia libre de formacion deben ser');
writeln('a 25 C [298.15 K] en Joule/grmol ');
writeln('El CP=a1+a2*T+a3*T^2+a4*T^3+a5/T^2 en Joule/grmol/K');
writeln('Para terminar dar <RETURN> al momento de pedir el nombre');
writeln('La Tc es en K y la Pc es en atm');
GotoXY(15,20);
write('Oprima cualquier tecla para continuar');readln;
i:=0+FileSize(Nombrelog);
repeat
with Registro do
begin
  i:=i+1;
  Clrscr;
  writeln('Compuesto ',i);
  write('Nombre del compuesto ');readln(nombre);
  if nombre<>' ' then
  begin
    write('Entalpia de formacion ');readln(entalpia);
    write('Energia libre de formacion ');readln(energiag);
    for j:=1 to 5 do
    begin
      write('Coeficiente Cp',j,' del CP ');readln(CaCal[j]);
    end; {for}
    write('Temperatura Critica (K) ');readln(TempC);
    write('Presion Critica (atm) ');readln(PresC);
    write('Factor Acentrico ');readln(Factor);
    write('Factor de Mathias ');readln(CoefMat);
    j:=0;
    Clrscr;
    writeln('De el nombre de los elementos que lo componen');
    writeln('De el numero de atomos presentes de cada elemento');
    writeln('Tambien el coeficiente en la reaccion de formacion');
    writeln('Para terminar de <RETURN>');
    repeat
      j:=j+1;
      write('Elemento ',j,' ');readln(Comp[j]);
      if (Comp[j]<>' ') then begin
        write('Atomos presentes ');readln(Co[1,j]);
        write('Coeficiente en reaccion de formacion ');readln(Co[2,j]);
      end; {if}
    until (Comp[j]=' ');
    write(Nombrelog, Registro);
  end; {if}
end; {with}
until registro.nombre='';

```

```

Close(Nombrelog);
End; {GuardaDatos}

Procedure Actualiza;
Var
  i,j,k,l :integer;
Begin
  Clrscr;
  write('Nombre del archivo a actualizar ');readln(Nombreach);
  Assign(Nombrelog,Nombreach);
  Reset(Nombrelog);
  i:=FileSize(Nombrelog);
  write('Nombre del compuesto a actualizar ');readln(Compuesto);
  j:=0;
  repeat
    j:=j+1;
    Read(Nombrelog,Registro);
  until (registro.nombre=compuesto) or (j=i);
  if registro.nombre<>compuesto then
  begin
    write('Ese compuesto no, existe en el archivo ',Nombreach);
    readln;
  end {then}
  else
  begin
    Clrscr;
    with Registro do
      begin
        writeln('Entalpia ',entalpia:4:2,' Joules/grmol');
        write('Quiere cambiarlo <S> o <N>? ');readln(cambio);
        if cambio in ['S','s'] then
          begin
            write('Nueva entalpia ');readln(entalpia);
          end; {if}
        writeln('Energia libre ',energiag:4:2,' Joules/grmol');
        write('Quiere cambiarlo <S> o <N>? ');readln(cambio);
        if cambio in ['S','s'] then
          begin
            write('Nueva energia libre ');readln(energiag);
          end; {if}
        writeln(' Cp de la forma =a1+a2T+a3T^2+a4T^3+a5/T^2 Joules/grmol/K');
        for i:=1 to 5 do
          begin
            writeln('Coeficiente ',i,' del CP ',CaCal[i]);
            write('Quiere cambiarlo <S> o <N>? ');readln(cambio);
            if cambio in ['S','s'] then
              begin
                write('Nuevo coeficiente ');readln(CaCal[i]);
              end; {if}
            end; {for}
          writeln('Temperatura Critica ',TempC:4:2,' K');
          write('Quiere cambiarlo <S> o <N>? ');readln(cambio);
          if cambio in ['S','s'] then
            begin
              write('Nueva temperatura critica ');readln(TempC);
            end;
          end;
        end;
      end;
    end;
  end;
End;

```

```

end; {if}
writeln('Presion Critica ',PresC:4:2,' atm');
write('Quiere cambiarlo <S> o <N>? ');readln(cambio);
if cambio in ['S','s'] then
begin
  write('Nueva presion critica ');readln(PresC);
end; {if}
writeln('Factor Acentrico ',Factor:4:4);
write('Quiere cambiarlo <S> o <N>? ');readln(cambio);
if cambio in ['S','s'] then
begin
  write('Nueva factor acentrico ');readln(Factor);
end; {if}
writeln('Factor de Mathias ',CoefMAT:4:4);
write('Quiere cambiarlo <S> o <N>? ');readln(cambio);
if cambio in ['S','s'] then
begin
  write('Nueva factor de Mathias ');readln(CoefMat);
end; {if}
l:=1;
writeln('Para no cambiar atomos dar <Q> ');
while not(Cambio in ['q','Q']) and (l<5) and (Comp[l]<>'') do
begin
  writeln('Elemento ',Comp[l],' atomos ',Co[1,l]:4:2,
        ' formacion ',Co[2,l]:4:2);
  write('Quiere cambiarlo <S> <N> ');readln(cambio);
  if (cambio in ['s','S']) then begin
    write('Nombre elemento ');readln(Comp[l]);
    write('Atomos ');readln(Co[1,l]);
    write('Formacion ');readln(Co[2,l]);
  end; {if}
  l:=l+1;
end; {while}
end; {with}
Seek(Nombrelog,j-1);
Write(Nombrelog,Registro);
end; {else}
Close(Nombrelog);
End; {Actualiza}

```

Procedure Lectura;

```

Var
  i,j,k,l :integer;
Begin
  Clrscr;
  write('Nombre del archivo a leer ');readln(Nombreach);
  write('Numero de compuestos a buscar ');readln(cantidad);
  Assign(Nombrelog,Nombreach);
  Reset(Nombrelog);
  i:=FileSize(Nombrelog);
  for k:=1 to cantidad do
  begin
    Clrscr;
    write('Nombre del compuesto a buscar ');readln(Compuesto);
    j:=0;
  end;

```

```

Reset(Nombrelog);
repeat
  j:=j+1;
  Read(Nombrelog,Registro);
until (registro.nombre=compuesto) or (j=1);
if registro.nombre<>compuesto then
  writeln('Ese compuesto no existe en el archivo ',Nombrearch)
else
  begin
  Clrscr;
  with Registro do
  begin
    writeln('Entalpia ',entalpia,' Joule/grmol');
    writeln('Energia libre ',energiag,' Joule/grmol');
    writeln('Cp=a1+a2T+a3T^2+a4T^3+a5/T^2 Joule/grmol/K');
    for l:=1 to 5 do
      writeln('Coeficiente ',l,' del CP ',CaCal[l]);
    writeln('Temperatura critica ',TempC,' K');
    writeln('Presion critica ',PresC,' atm');
    writeln('Factor acentrico ',Factor);
    writeln('Factor de Mathias ',CoefMat);
    l:=1;
    if Comp[l]<>' ' then writeln('Elemento Atomos Reaccion Formacion');
    while (Comp[l]<>' ') do
      begin
        writeln(' ',Comp[l]:3,' ',Co[1,l]:2:1,' ',Co[2,l]:2:1);
        l:=l+1;
      end; {while}
      GotoXY(20,20);
      write('Oprima cualquier tecla para continuar');readln;
    end; {with}
  end; {else}
end; {for}
Close(Nombrelog);
End; {Lectura}

Procedure CompArch;
Var
i,j :integer;
begin
  Clrscr;
  write('Nombre del archivo a leer ');readln(Nombrearch);
  Assign(Nombrelog,Nombrearch);
  Reset(Nombrelog);
  i:=FileSize(Nombrelog);
  for j:=1 to i do
  begin
    read(Nombrelog,Registro);
    writeln('Compuesto ',j,' ',Registro.Nombre);
  end; {for}
  Close(NombreLog);
  readln;
end;

BEGIN

```

```
Repeat
  Clrscr;
  writeln('Desea 1-Crear un banco de datos o agregar datos');
  writeln('      2-Actualizar datos de un banco');
  writeln('      3-Leer datos de un banco');
  writeln('      4-Saber que compuestos estan en un archivo');
  writeln('      5-Terminar ');
  write('? '); readln(opcion);
  Case opcion of
    '1': GuardaDatos;
    '2': Actualiza;
    '3': Lectura;
    '4': CompArch;
  else
    Clrscr;
  end; {case}
Until (opcion='5');
end;

.end.
```

Equation of State", Chem. Eng. Sci. 27 1197-1203 (1972)

16.-Stryjek, R., and Vera, J.H., "PRSV: An Improved Peng-Robinson Equation of State for pure Compounds and Mixtures", Can. J. Chem. Eng., 64, 323-333 (1986)

17.-Stryjek, R., and Vera, J.H., "PRSV: An Improved Peng-Robinson Equation of State with New Mixing Rules for Strongly Nonideal Mixtures", Can. J. Chem. Eng., 64, 334-340 (1986)

18.-Stryjek, R., and Vera, J.H., "PRSV2: A Cubic Equation of State for Accurate Vapor-Liquid Equilibria Calculations", Can. J. Chem. Eng., 64, 820-826 (1986)

19.-Walas, S.M., "Phase Equilibria in Chemical Engineering"

20.-White, C.W., and Seider, W.D., "Computation of Phase and Chemical Equilibrium Part IV: Approach to Chemical Equilibrium" AIChE J. 27,3 (1981)

21.-Wylie, C. R., "Matematicas Superiores para Ingenieria" Mc Graw Hill 4a Ed. (1982)

Bibliografia

- 1.-Bazua R., Enrique Apuntes de Termodinamica UNAM (1986)
- 2.-Balzhiser, R.E., Samuels, M.R., and J.D. Eliassen "Chemical Engineering Thermodynamics" Prentice Hall (1972)
- 3.-Castellan, W.G., "Fisicoquimica" Fondo Educativo Interamericano 2aEd. (1976)
- 4.-Conte, S.D., and de Boor, C., "Elementary Numerical Analysis" Mc Graw Hill 3a Ed. (1983)
- 5.-Gautam, R., and Seider, W.D., "Computation of Phase and Chemical Equilibrium Part I:Local and Constrained Minima in Gibbs Free Energy" AIChE J. 25,6 (1979a)
- 6.-Gautam, R., and Seider, W.D., "Computation of Phase and Chemical Equilibrium Part II:Phase-Splitting" AIChE J., 25,6 (1979b)
- 7.-Gautam, R., and Seider, W.D., "Computation of Phase and Chemical Equilibrium Part III:Electrolytic Solution" AIChE J., 25,6 (1979c)
- 8.-Mathias, P.M., "A Versatil Phase Equilibrium Equation of State", Ind. Eng. Chem. Process Des. Dev. 22, 385-391 (1983)
- 9.-Peng. D.Y., and Robinson, D.B., "A New Two-Constant Equation of State", Ind. Eng. Chem. Fundamentals 15, 59-64 (1976)
- 10.-Prausnitz, J.M. "Molecular Thermodynamics of Fluid-Phase Equilibria" Prentice-Hall (1969)
- 11.-Ralston, A., and Rabinowitz, P., "A First Course in Numerical Analysis" 2aEd. McGraw Hill (1984)
- 12.-Reid, C.R., Prausnitz, J.M., and T.K., Sherwood "The Proprieties of Gases and Liquids" Mc Graw Hill. 3a Ed. (1977)
- 13.-Scraton, R.E., "Metodos Numericos Basicos" Mc Graw Hill (1987)
- 14.-Smith, J.M. and H.C. van Ness "Introduction to Chemical Engineering Thermodynamics" McGraw Hill, 3a Ed. (1985)
- 15.-Soave, G., "Equilibrium Constans from a Modified Redlich-Kwong