PROBLEMAS

Grupo A

1 Suponga que Mondo ya no debe cumplir a tiempo con su demanda. Por cada trimestre en que se incumple la demanda de una motocicleta, se fija un costo de penalización de 110 dólares por motocicleta. Por consiguiente, la demanda se puede acumular ahora. Sin embargo, todas las demandas se deben cumplir al finalizar el trimestre 4. Modifique el planteamiento del problema de Mondo para permitir que la demanda se acumule. (*Sugerencia*: La demanda incumplida corresponde a $i_t \leq 0$. Por lo tanto, i_t ahora no tiene restricciones de signo, y se tiene que sustituir $i_t = i_t' - i_t''$. Ahora i_t'' será la cantidad de demanda que se incumple al final del trimestre t.)

2 Resuelva la siguiente PL mediante el algoritmo simplex

$$\begin{aligned} \max z &= 2x_1 + x_2 \\ \text{s.a} \quad 3x_1 + x_2 &\leq 6 \\ x_1 + x_2 &\leq 4 \\ x_1 \geq 0,\ x_2 &\text{ urs} \end{aligned}$$

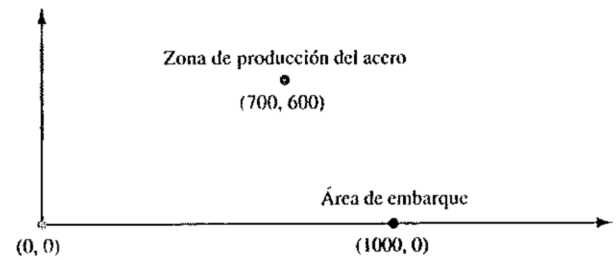
Grupo B

3 Steelco hace frente a la demanda siguiente de acero durante los tres meses siguientes: 100 toneladas (t) (mes 1); 200 t (mes 2); 50 t (mes 3). Durante cualquier mes, un trabajador produce hasta 15 t de acero. Cada trabajador recibe como pago 5 000 dólares al mes. Los trabajadores pueden ser contratados o despedidos a un costo de 3 000 dólares por trabajador despedido y 4 000 dólares por trabajador contratado (toma 0 tiempo contratar a un obrero). El costo de retener una tonelada de acero en inventario por un mes es 100 dólares. La demanda se podría acumular (*backlog*) a un costo de 70 dólares por cada mes. Es decir, si una t de la demanda del mes 1 se cumple durante el mes 3, entonces se incurre en un costo por acumulación de 140 dólares. Al empezar el mes 1, Steelco tiene 8 trabajadores. Pueden ser contratados cuando mucho 2 trabajadores al mes. Toda la demanda debe quedar cumplida al final del mes 3. La materia prima usada para producir una tonelada de acero cuesta 300 dólares. Plantee un PL para minimizar los costos de Steelco.

4 Demuestre cómo podría usar la programación lineal para resolver el problema siguiente:

$$\begin{aligned} \max z &= |2x_1 - 3x_2| \\ \text{s.a} \quad 4x_1 + x_2 &\leq 4 \\ \text{s.a} \quad 2x_1 - x_2 &\leq 0.5 \\ x_1, x_2 &\geq 0 \end{aligned}$$

FIGURA 13



5† La planta principal de Steelco tiene en la actualidad un área de manufactura y una zona de embarque, ubicadas como se indica en la figura 13 (distancias en pies). La compañía debe determinar dónde ubicar las instalaciones de fundición y las de ensamble y almacenamiento, con el fin de minimizar los costos diarios por desplazar el material por la planta. La cantidad de viajes hechos todos los días son los que se indican en la tabla 50.

Si se supone que todos los viajes (acarreo) son sólo en dirección este-oeste o norte-sur, plantee un PL que se pueda aplicar para determinar dónde se podrían ubicar las instalaciones de fundición y de ensamble y almacenamiento con objeto de minimizar los costos de transporte diario. (*Sugerencia*: Si las instalaciones de fundición tienen coordenadas (c1, c2) ¿cómo se debería interpretar la restricción $c1 - 700 = e_1 - w_1$?)

6 Demuestre que después de cierto número de pivoteos el coeficiente de x_i' en cada renglón del tableau simplex será igual al negativo del coeficiente de x_i'' en el mismo renglón.

7 Clothco fabrica pantalones. Durante los seis meses siguientes puede vender *hasta* la cantidad de pantalones de la tabla 51.

La demanda que no se surte durante un mes se pierde. Entonces, Clothco puede vender hasta, por ejemplo, 500 pantalones durante el mes 1. Un pantalón se vende en 40 dólares, requiere 2 horas de mano de obra y 10 dólares de materia prima. Cuando empieza el mes 1, Clothco tiene 4 obreros. Un obrero puede trabajar haciendo pantalones hasta 200 horas al mes, y su salario es de 2 000 dólares mensuales (sin tener en cuenta cuántas horas trabaja). Al principio de todos los meses, se puede contratar o despedir empleados. Cuesta 1 500 dólares contratar un empleado y 1 000 despedirlo. Se fija un

TABLA 50

De	A	Cantidad diaria de viajes	Costo (centavos) por 100 pies de acarreo
Fundición	Ensamble y almacenamiento	40	10
Producción de acero	Fundición	8	10
Produción de acero	Ensamble y almacenamiento	8	10
Embarque	Ensamble y almacenamiento	2	20

TABLA 51

Mes	Demanda máxima
1	500
2	600
3	300
4	400
5	300
6	800

†Basado en Love y Yerex (1976).

costo de 5 dólares por retener un pantalón contra el inventario final de cada mes.

Determine cómo Clothco puede maximizar su utilidad para los seis meses siguientes. Ignore el hecho de que durante cada mes la cantidad de empleados contratados o despedidos tiene que ser un entero.

4.15 Método de Karmarkar para resolver PL

A continuación se presenta una explicación breve del método de Karmarkar para resolver PL. Si desea una explicación más detallada, refiérase a la sección 10.6. El método de Karmarkar requiere que el PL esté en la forma siguiente

$$\begin{aligned} \min z &= \mathbf{cx} \\ \text{s.a} \quad K\mathbf{x} &= 0 \\ x_1 + x_2 + \cdots x_n &= 1 \\ x_i &\geq 0 \end{aligned}$$

y que

1 El punto $\mathbf{x}^0 = [\frac{1}{n} \quad \frac{1}{n} \quad \cdots \quad \frac{1}{n}]$ debe ser factible para este PL.

2 El valor óptimo de z para el PL es igual a cero.

Lo más sorprendente es que cualquier PL se puede expresar en esta forma. En el método de Karmarkar se utiliza una transformación de la geometría proyectiva para crear un conjunto de variables transformadas $y_1, y_2, \ldots, y_n$. Esta transformación (llamémosla f) transforma siempre el punto actual en el "centro" de la región factible en el espacio definido por las variables transformadas. Si la transformación convierte al punto $\mathbf{x}$ en el punto $\mathbf{y}$, se escribe $f(\mathbf{x}) = \mathbf{y}$. En el espacio transformado, el algoritmo empieza con el desplazamiento desde $f(\mathbf{x}^0)$ en el espacio transformado en una "buena" dirección (una dirección que tiende a mejorar z y conserva la factibilidad). Con lo anterior se genera un punto $\mathbf{y}^1$ en el espacio transformado, el cual está cercano a la frontera de la región factible. Este nuevo punto es $\mathbf{x}^1$, que satisface $f(\mathbf{x}^1) = \mathbf{y}^1$. El procedimiento se repite (esta vez $\mathbf{x}^1$ reemplaza a $\mathbf{x}^0$) hasta que el valor z para $\mathbf{x}^k$ es suficientemente cercano a 0.

Si el punto actual es $\mathbf{x}^k$, entonces la transformación tendrá la propiedad de que $f(\mathbf{x}^k) = [\frac{1}{n} \quad \frac{1}{n} \quad \cdots \quad \frac{1}{n}]$. Por lo tanto, en el espacio transformado, siempre nos alejamos del "centro" de la región factible.

Se ha demostrado que el método de Karmarkar es un **algoritmo de tiempo polinómico**. Esto implica que si un PL de tamaño n se resuelve mediante el método de Karmarkar, entonces existen los números positivos a y b tales que para cualquier n se puede resolver un PL de tamaño n en un tiempo de cuando mucho an^b.[†]

En contraste con el método de Karmarkar, el algoritmo simplex es un **algoritmo de tiempo exponencial** para resolver los PL. Si un PL de tamaño n se resuelve mediante el simplex, entonces existe un número positivo c tal que, para cualquier n, el algoritmo simplex encontrará la solución óptima en un tiempo de cuando mucho $c2^n$. Para n suficientemente grande (para a, b y c positivas), $c2^n > an^b$. Esto significa que, en teoría, un algoritmo de tiempo polinómico es superior a un algoritmo de tiempo exponencial. Las pruebas preliminares del método de Karmarkar (efectuadas por Karmarkar) demuestran que para PL grandes que surgen en la aplicación actual, este método podría ser hasta 50 veces más rápido que el algoritmo simplex. Esperemos que el método de Karmarkar ayude a los investigadores a resolver varios problemas grandes de PL que, en la actualidad, requieren una cantidad prohibitivamente elevada de tiempo de computadora cuando se busca la solución del PL mediante el simplex. Si el método de Karmarkar cumple su promesa preliminar, la capacidad para plantear modelos de PL será aun más importante en el futuro cercano de lo que es hoy.

El *Military Airlift Command* utilizó recientemente el método de Karmarkar para determinar la frecuencia en que se deben volar varias rutas y qué aviones usar. El PL resultan-

[†] La dimensión de un PL podría ser definida como el número de símbolos necesarios para representar el PL en la notación binaria.

te contenía 150 000 variables y 12 000 restricciones, y se resolvió en una hora de tiempo de computadora mediante el método de Karmarkar. Un PL de estructura similar que contenga 36 000 variables y 10 000 restricciones requeriría 4 horas de tiempo de computadora si se utiliza el método simplex. Delta Airlines utilizó el método de Karmarkar para elaborar los horarios mensuales para 7 000 pilotos y más de 400 aviones. Cuando el proyecto finalizó, Delta esperaba ahorrar millones de dólares.

4.16 Toma de decisiones con varios atributos en ausencia de incertidumbre: programación por objetivos

En ciertas situaciones, un tomador de decisiones podría hacer frente a varios objetivos, y podría no haber un punto en la región factible del PL que satisfaga todos los objetivos. En tal caso, ¿cómo puede quien toma las decisiones elegir una decisión satisfactoria? La **programación por objetivos** es una técnica a la que se puede recurrir en estas situaciones. Por medio del ejemplo siguiente, se ilustran las ideas principales de la programación por objetivos.

EJEMPLO 10 Programación por objetivos de Burnit

La agencia de publicidad Leon Burnit trata de determinar un horario de anuncios por TV para la compañía automotriz Priceler. Priceler tiene tres objetivos:

Objetivo 1 Por lo menos 40 millones de varones de altos ingresos (VAI) deben ver los anuncios.

Objetivo 2 Por lo menos 60 millones de personas de bajos ingresos (PBI) deben ver los anuncios.

Objetivo 3 Por lo menos 35 millones de mujeres de altos ingresos (MAI) deben ver los anuncios.

Leon Burnit puede comprar dos tipos de anuncios: los que se muestren en los juegos de futbol americano y los que se vean en las telenovelas. Puede gastar, cuando mucho, 600 000 dólares en los anuncios. Los costos de publicidad y los posibles televidentes para un anuncio de cada tipo de un minuto se proporcionan en la tabla 52. Leon Burnit tiene que determinar cuántos anuncios en el futbol y cuántos en las telenovelas debe comprar para Priceler.

Solución Sea

$$x_1 = \text{número de minutos de anuncios mostrados en los juegos de futbol}$$
$$x_2 = \text{número de minutos de anuncios mostrados en las telenovelas}$$

Entonces, cualquier solución factible del PL siguiente cumpliría con los objetivos de Priceler:

$$\begin{array}{lrl}
\min \text{ (o max) } z = & 0x_1 + 0x_2 & \text{(o cualquier otra función objetivo)} \\
\text{s.a} & 7x_1 + 3x_2 \geq 40 & \text{(restricción de los VAI)} \\
& 10x_1 + 5x_2 \geq 60 & \text{(restricción de las PBI)} \\
& 5x_1 + 4x_2 \geq 35 & \text{(restricción de las MAI)} \\
& 100x_1 + 60x_2 \leq 600 & \text{(Restricción del presupuesto)} \\
& x_1, x_2 \geq 0 &
\end{array} \quad (21)$$

En la figura 14 se encuentra que ningún punto que cumple con la restricción del presupuesto satisface los otros tres objetivos de Priceler. Por lo tanto, (21) no tiene solución factible. Es imposible cumplir todos los objetivos de Priceler, así que Burnit podría pedir a Priceler establecer, por cada objetivo, un costo (por unidad faltante para cumplir cada objetivo) que se genera por no cumplir con la meta. Suponga que Priceler determina que

TABLA 52
Costo y número de televidentes de los anuncios de Priceler

	Millones de televidentes			
Anuncio	VAI	PBI	MAI	Costo (dólares)
Futbol	7	10	5	100 000
Telenovela	3	5	4	60 000

Por cada millón de exposiciones con el que Priceler no alcanza el objetivo de los VAI genera a esta compañía una penalización de 200 000 dólares debido a las ventas perdidas.

Por cada millón de exposiciones con el que Priceler no alcanza el objetivo de las PBI genera a esta compañía una penalización de 100 000 dólares debido a las ventas perdidas.

Por cada millón de exposiciones con el que Priceler no alcanza el objetivo de las MAI genera a esta compañía una penalización de 50 000 dólares debido a las ventas perdidas.

Ahora, Burnit puede plantear un PL que minimiza el costo incurrido por desviarse de los tres objetivos de Priceler. La estrategia es transformar cada restricción de desigualdad en (21) que representa uno de los objetivos de Priceler en una restricción de igualdad. Como no se sabe si la solución que minimiza el costo estará por abajo o por arriba de un objetivo dado, es necesario definir las variables siguientes:

s_i^+ = cantidad con la que se excede numéricamente el i-ésimo objetivo

s_i^- = cantidad con la que queda por debajo numéricamente del i-ésimo objetivo

Tanto s_i^+ como s_i^- reciben el nombre de **variables de desviación.** Por lo que se refiere al problema de Priceler, se supone que cada s_i^+ y s_i^- se mide en millones de exposiciones. Es posible volver a escribir las tres primeras restricciones de (21), usando las variables de desviación, como

$$7x_1 + 3x_2 + s_1^- - s_1^+ = 40 \qquad \text{(restricción de los VAI)}$$
$$10x_1 + 5x_2 + s_2^- - s_2^+ = 60 \qquad \text{(restricción de las PBI)}$$
$$5x_1 + 4x_2 + s_3^- - s_3^+ = 35 \qquad \text{(restricción de las MAI)}$$

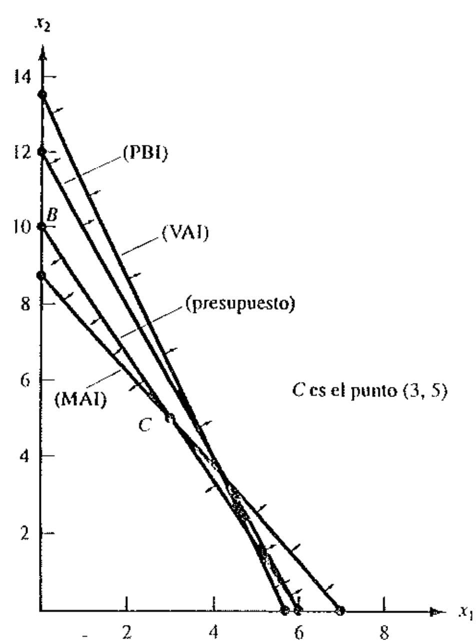


FIGURA 14
Restricciones de Priceler

Por ejemplo, suponga que $x_1 = 5$ y $x_2 = 2$. Con este programa de anuncios se obtiene $7(5) + 3(2) = 41$ millones de exposiciones de VAI. Este valor excede el objetivo de VAI por $41 - 40 = 1$ millón de exposiciones, y $s_1^- = 0$ y $s_1^+ = 1$. Asimismo, este plan origina $10(5) + 5(2) = 60$ millones de exposiciones de PBI. Así se cumple exactamente el requisito de PIB, y $s_2^- = s_2^+ = 0$. Por último, este programa da como resultado $5(5) + 4(2) = 33$ millones de exposiciones de MAI. Este valor está por abajo del objetivo de MAI con $35 - 33 = 2$ millones de exposiciones, y $s_3^- = 2$ y $s_3^+ = 0$.

Suponga que Priceler quiere minimizar la penalización total por las ventas perdidas. En términos de las variables de desviación, la penalización total por las ventas perdidas (en miles de dólares) ocasionadas por desviarse de los tres objetivos es $200s_1^- + 100s_2^- + 50s_3^-$. El coeficiente de la función objetivo para la variable asociada con el objetivo i se llama peso o ponderación para el objetivo i. El objetivo más importante tiene el peso más elevado (es decir, es el más ponderado), etcétera. Por consiguiente, en el ejemplo de Priceler, el objetivo 1 (VAI) es el más importante, el objetivo 2 (PBI) es el segundo en importancia y el objetivo 3 (MAI) es el menos importante.

Burnit puede minimizar la penalización por las ventas perdidas mediante la resolución del PL siguiente:

$$
\begin{array}{lrll}
\min z = 200s_1^- + 100s_2^- + 50s_3^- & & & \\
\text{s.a} & 7x_1 + 3x_2 + s_1^- - s_1^+ = 40 & \text{(restricción de los VAI)} & \\
 & 10x_1 + 5x_2 + s_2^- - s_2^+ = 60 & \text{(restricción de las PBI)} & \\
 & 5x_1 + 4x_2 + s_3^- - s_3^+ = 35 & \text{(restricción de las MAI)} & \quad (22)\\
 & 100x_1 + 60x_2 \leq 600 & \text{(restricción del presupuesto)} & \\
 & \text{Todas las variables son no negativas} & &
\end{array}
$$

La solución óptima para este PL es $z = 250$, $x_1 = 6$, $x_2 = 0$, $s_1^+ = 2$, $s_2^+ = 0$, $s_3^+ = 0$, $s_1^- = 0$, $s_2^- = 0$, $s_3^- = 5$. De esta manera se cumplen el objetivo 1 y el 2 (los objetivos con los costos más altos, o pesos, por cada unidad de desviación desde el objetivo), pero no se cumple con el objetivo menos importante (objetivo 3).

OBSERVACIONES Si se fracasa en el cumplimiento del objetivo i cuando el valor logrado de un atributo es numéricamente menor que el valor deseado del objetivo i, entonces aparecerá un término con s_i^- en la función objetivo. Si el incumplimiento con el objetivo i se presenta cuando el valor logrado de un atributo es numéricamente mayor que el valor deseado del objetivo i, entonces aparecerá un término con s_i^+ en la función objetivo. Asimismo, si se desea alcanzar exactamente un objetivo y se fija una penalización al quedar por arriba o por abajo del mismo, entonces aparecerán los términos s_i^- y s_i^+.

Suponga que se modifica el ejemplo de Priceler y se toma la decisión de que la restricción del presupuesto de 600 000 dólares es un objetivo. Si se establece una penalización de 1 dólar por cada dólar con el que se incumpla el objetivo, entonces la formulación apropiada de la programación por objetivos sería

$$
\begin{array}{lrl}
\min z = 200s_1^- + 100s_2^- + 50s_3^- + s_4^+ & & \\
\text{s.a} & 7x_1 + 3x_2 + s_1^- - s_1^+ = 40 & \text{(restricción de los VAI)} \\
 & 10x_1 + 5x_2 + s_2^- - s_2^+ = 60 & \text{(restricción de las PBI)} \\
 & 5x_1 + 4x_2 + s_3^- - s_3^+ = 35 & \text{(restricción de las MAI)} \\
 & 100x_1 + 60x_2 + s_4^- - s_4^+ = 600 & \text{(restricción del presupuesto)} \\
 & \text{Todas las variables son no negativas} &
\end{array}
$$

En contraste con la solución óptima previa, la solución óptima para este PL es $z = 33\frac{1}{3}$, $x_1 = 4\frac{1}{3}$, $x_2 = 3\frac{1}{3}$, $s_1^+ = \frac{1}{3}$, $s_2^+ = 0$, $s_3^+ = 0$, $s_4^+ = 33\frac{1}{3}$, $s_1^- = 0$, $s_2^- = 0$, $s_3^- = 0$, $s_4^- = 0$. Por lo tanto, cuando se define que la restricción del presupuesto es un objetivo, la solución óptima es cumplir los tres objetivos publicitarios quedando $33\frac{1}{3}$ miles de dólares por arriba del presupuesto.

Programación por objetivos prioritarios

En el planteamiento del PL para el ejemplo de Burnit, se supuso que Priceler podría determinar con exactitud la importancia relativa de los tres objetivos. Por ejemplo, Priceler determinó que el objetivo de los VAI era $\frac{200}{100} = 2$ veces la importancia del objetivo PBI, y que el objetivo PBI era $\frac{100}{50} = 2$ veces la importancia del objetivo de los VAI. Sin embargo, en muchas situaciones, el tomador de decisiones podría no ser capaz de determinar con toda precisión la importancia relativa de los objetivos. Cuando éste es el caso, la *programación por objetivos prioritarios* podría ser una herramienta útil. Para poder aplicar este tipo de programación, el tomador de decisiones tiene que jerarquizar sus objetivos desde el más importante (objetivo 1) hasta el menos importante (objetivo n). El coeficiente de la función objetivo para la variable que representa el objetivo i será P_i. Se supone que

$$P_1 >>> P_2 >>> P_3 >>> \cdots >>> P_n$$

Por lo tanto, la ponderación o peso del objetivo 1 es mucho mayor que el del objetivo 2, y el peso del objetivo 2 es mucho mayor que el del objetivo 3, y así sucesivamente. Esta definición de $P_1, P_2, \ldots, P_n$ asegura que quien toma las decisiones primero trata de satisfacer el objetivo más importante (objetivo 1). Entonces, entre todos los puntos que satisfacen el objetivo 1, el tomador de decisiones trata de acercarse tanto como sea posible al objetivo 2, y así sucesivamente. Se continúa de este modo hasta que la única manera de cumplir en forma aproximada con un objetivo es incrementar la desviación desde un objetivo de alta prioridad.

Para el problema de Priceler, el planteamiento de la programación por objetivos prioritarios se consigue a partir de (22) al reemplazar la función objetivo de (22) por $P_1s_1^- + P_2s_2^- + P_3s_3^-$. Por lo tanto, el planteamiento del problema de Priceler con programación por objetivos prioritarios es

$$\begin{array}{lrll}
\min z = P_1s_1^- + P_2s_2^- + P_3s_3^- & & & \\
\text{s.a} & 7x_1 + 3x_2 + s_1^- - s_1^+ = 40 & \text{(restricción de los VAI)} & \\
 & 10x_1 + 5x_2 + s_2^- - s_2^+ = 60 & \text{(restricción de las PBI)} & \\
 & 5x_1 + 4x_2 + s_3^- - s_3^+ = 35 & \text{(restricción de las MAI)} & (23) \\
 & 100x_1 + 60x_2 \leq 600 & \text{(restricción del presupuesto)} & \\
 & \text{Todas las variables son no negativas} & &
\end{array}$$

Supóngase que el tomador de decisiones tiene n objetivos. Para aplicar la programación por objetivos prioritarios, es necesario separar la función objetivo en n componentes, donde el componente i es el término de la función objetivo que se relaciona con el objetivo i. Se define

$$z_i = \text{término de la función objetivo que se relaciona con el objetivo } i$$

Por lo que se refiere al ejemplo de Priceler, $z_1 = P_1s_1^-$, $z_2 = P_2s_2^-$ y $z_3 = P_3s_3^-$. Los problemas de programación por objetivos prioritarios se pueden resolver mediante una generalización del simplex conocida como **programación por objetivos simplex**. Para preparar un problema que se pueda solucionar por medio de programación por objetivos simplex es necesario calcular n renglones 0, en donde el i-ésimo renglón 0 corresponde al objetivo i. Por consiguiente, para el problema de Priceler, se tiene

$$\text{Renglón 0 (objetivo 1): } z_1 - P_1s_1^- = 0$$
$$\text{Renglón 0 (objetivo 2): } z_2 - P_2s_2^- = 0$$
$$\text{Renglón 0 (objetivo 3): } z_3 - P_3s_3^- = 0$$

A partir de (23) se encuentra que $BV = \{s_1^-, s_2^-, s_3^-, s_4\}$ (s_4 = variable de holgura para la cuarta restricción) es una solución factible básica inicial que se podría usar para resolver (23) por medio del algoritmo simplex (o el algoritmo de la programación por objetivos simplex). Al igual que con el simplex regular, primero se tienen que eliminar de cada renglón 0 todas las variables en la base inicial. Al sumar P_1 (restricción de los VAI) al renglón 0 (objetivo 1) se obtiene

$$\text{Renglón 0 (objetivo 1): } z_1 + 7P_1x_1 + 3P_1x_2 - P_1s_1^+ = 40P_1 \qquad \text{(VAI)}$$

Al sumar P_2 (restricción de las PBI) al renglón 0 (objetivo 2) se tiene

$$\text{Renglón 0 (objetivo 2): } z_2 + 10P_2x_1 + 5P_2x_2 - P_2s_2^+ = 60P_2 \qquad \text{(PBI)}$$

Al sumar P_3 (restricción de los VAI) al renglón 0 (objetivo 3) se tiene

$$\text{Renglón 0 (objetivo 3): } z_3 + 5P_3x_1 + 4P_3x_2 - P_3s_3^+ = 35P_3 \qquad \text{(VAI)}$$

El problema de Priceler ya se puede resolver mediante la programación por objetivos simplex.

Las diferencias entre la programación por objetivos simplex y el simplex ordinario, son las siguientes:

1 El simplex ordinario tiene un solo renglón 0, en tanto que la programación por objetivos simplex requiere n renglones 0 (uno por cada objetivo).

2 En la programación por objetivos simplex se utiliza el método que sigue para determinar la variable entrante: se determina el objetivo con la prioridad más alta (objetivo i') que no ha sido satisfecho (o se busca el objetivo i' cuya prioridad es la más alta que tiene $z_{i'} > 0$). Se busca la variable con el coeficiente más positivo en el renglón 0 (objetivo i') y se introduce esta variable (sujeta a las restricciones siguientes) en la base. Con esto se reduce $z_{i'}$ y se asegura que se está cerca de cumplir con el objetivo i'. *Pero si una variable tiene un coeficiente negativo en el renglón* 0 *relacionado con un objetivo que tiene una prioridad mayor que* i', *entonces la variable no puede entrar a la base.* Si se introdujera dicha variable a la base, se incrementaría la desviación respecto a algún objetivo de mayor prioridad. Si la variable con el coeficiente más positivo en el renglón 0 (objetivo i') no puede entrar a la base, entonces trate de hallar otra variable con un coeficiente positivo en el renglón 0 (objetivo i'). Si ninguna variable del renglón 0 (objetivo i') puede entrar a la base, entonces no hay modo de aproximarse en el cumplimiento del objetivo i' sin incrementar la desviación respecto a un objetivo de mayor prioridad. En este caso, cambie de renglón 0 (objetivo $i' + 1$) para intentar cumplir lo más aproximadamente posible con el objetivo $i' + 1$.

3 Cuando se ejecuta un pivoteo, es necesario actualizar el renglón 0 para cada objetivo.

4 Un tableau generará la solución óptima si se cumplen todos los objetivos (es decir, $z_1 = z_2 = \cdots = z_n = 0$), o si cada variable que entra a la base y disminuye el valor de z_i' para un objetivo incumplido i' incrementará la desviación respecto a algún objetivo i que tiene una prioridad mayor que el objetivo i'.

Enseguida se utiliza la programación por objetivos simplex para resolver el ejemplo de Priceler. En cada tableau, los renglones 0 se listan según las prioridades de los objetivos (desde la prioridad mayor, a la menor prioridad). El tableau inicial está en la tabla 53. La sfb actual es $s_1^- = 40$, $s_2^- = 60$, $s_3^- = 35$, $s_4 = 600$. Como $z_1 = 40P_1$, el objetivo 1 no se satisface. Con el fin de reducir la penalización relacionada con el incumplimiento del objetivo 1, se introduce la variable con el coeficiente más positivo (x_1) en el renglón 0 (VAI). La prueba del cociente indica que x_1 debe entrar a la base en la restricción de los VAI.

Después de que x_1 entra a la base, se llega a la tabla 54. La solución básica actual es $x_1 = \frac{40}{7}$, $s_2^- = \frac{20}{7}$, $s_3^- = \frac{45}{7}$, $s_4 = \frac{200}{7}$. Como $s_1^- = 0$ y $z_1 = 0$, se cumplió con el objetivo 1. Enseguida se trata de satisfacer el objetivo 2 (con la certeza de que se alcanzó el objetivo 1 de mayor prioridad). La variable con el coeficiente más positivo en el renglón 0 (PBI) es s_1^+. Observe que al introducir s_1^+ a la base no se incrementa z_1 [porque el coeficiente de s_1^+ en el renglón 0 (VAI) es 0]. Por lo tanto, después de introducir s_1^+ en la base, el objetivo 1 quedará satisfecho. La prueba del cociente indica que s_1^+ podría entrar a la base en la restricción de las PBI o en la del presupuesto. Se elige en forma arbitraria introducir s_1^+ en la base en la restricción del presupuesto.

Después de efectuar iteraciones con s_1^+ en la base se obtiene la tabla 55. Puesto que $z_1 = z_2 = 0$, se alcanzan los objetivos 1 y 2. Como $z_3 = 5P_3$, no se cumple el objetivo 3. La sfb actual es $x_1 = 6$, $s_2^- = 0$, $s_3^- = 5$, $s_1^+ = 2$. Ahora se trata de llegar lo más cerca posible del objetivo 3 (sin violentar el objetivo 1 o el objetivo 2). Como x_2 es la única variable con un coeficiente positivo en el renglón 0 (MAI), la única manera de acercarse al objetivo 3 (MAI)

TABLA 53
Tableau inicial de la programación por objetivos prioritarios para Priceler

	x_1	x_2	s_1^+	s_2^+	s_3^+	s_1^-	s_2^-	s_3^-	s_4	Ld
Renglón 0 (VAI)	$7P_1$	$3P_1$	$-P_1$	0	0	0	0	0	0	$z_1 = 40P_1$
Renglón 0 (PBI)	$10P_2$	$5P_2$	0	$-P_2$	0	0	0	0	0	$z_2 = 60P_2$
Renglón 0 (MAI)	$5P_3$	$4P_3$	0	0	$-P_3$	0	0	0	0	$z_3 = 35P_3$
VAI	7	3	-1	0	0	1	0	0	0	40
PBI	10	5	0	-1	0	0	1	0	0	60
MAI	5	4	0	0	-1	0	0	1	0	35
Presupuesto	100	60	0	0	0	0	0	0	1	600

TABLA 54
Primer tableau de la programación por objetivos prioritarios para Priceler

	x_1	x_2	s_1^+	s_2^+	s_3^+	s_1^-	s_2^-	s_3^-	s_4	Ld
Renglón 0 (VAI)	0	0	0	0	0	$-P_1$	0	0	0	$z_1 = 0$
Renglón 0 (PBI)	0	$\frac{5P_2}{7}$	$\frac{10P_2}{7}$	$-P_2$	0	$-\frac{10P_2}{7}$	0	0	0	$z_2 = \frac{20P_2}{7}$
Renglón 0 (MAI)	0	$\frac{13P_3}{7}$	$\frac{5P_3}{7}$	0	$-P_3$	$-\frac{5P_3}{7}$	0	0	0	$z_3 = \frac{45P_3}{7}$
VAI 1	$\frac{3}{7}$	$-\frac{1}{7}$	0	0	$\frac{1}{7}$	0	0	0	$\frac{40}{7}$	
PBI	0	$\frac{5}{7}$	$\frac{10}{7}$	-1	0	$-\frac{10}{7}$	1	0	0	$\frac{20}{7}$
MAI	0	$\frac{13}{7}$	$\frac{5}{7}$	0	-1	$-\frac{5}{7}$	0	1	0	$\frac{45}{7}$
Presupuesto	0	$\frac{120}{7}$	$\frac{100}{7}$	0	0	$-\frac{100}{7}$	0	0	1	$\frac{200}{7}$

TABLA 55
Tableau óptimo de la programación por objetivos prioritarios para Priceler

	x_1	x_2	s_1^+	s_2^+	s_3^+	s_1^-	s_2^-	s_3^-	s_4	Ld
Renglón 0 (VAI)	0	0	0	0	0	$-P_1$	0	0	0	$z_1 = 0$
Renglón 0 (PBI)	0	$-P_2$	0	$-P_2$	0	0	0	0	$-\frac{P_2}{10}$	$z_2 = 0$
Renglón 0 (MAI)	0	P_3	0	0	$-P_3$	0	0	0	$-\frac{P_3}{20}$	$z_3 = 5P_3$
VAI	1	$\frac{3}{5}$	0	0	0	0	0	0	$\frac{1}{100}$	6
PBI	0	-1	0	-1	0	0	1	0	$-\frac{1}{10}$	0
MAI	0	1	0	0	-1	0	0	1	$-\frac{1}{20}$	5
Presupuesto	0	$\frac{6}{5}$	1	0	0	-1	0	0	$\frac{7}{100}$	2

es introducir x_2 a la base. Pero observe que x_2 tiene un coeficiente negativo en el renglón 0 para el objetivo 2 (PBI). Por lo tanto, la única manera de alcanzar casi el objetivo 3 (MAI) es incumplir un objetivo de mayor prioridad, el objetivo 2 (PBI). Éste es, por lo tanto, un tableau óptimo. La solución de la programación por objetivos prioritarios es comprar 6 minutos de anuncios en el futbol y ninguno en las telenovelas. Se cumplen los objetivos 1 y 2 (VAI y PBI), y a Priceler le faltan 5 millones de exposiciones para cumplir con el objetivo 3 (MAI).

Si el analista tiene acceso a un código de programación por objetivos computarizado, entonces, es posible generar muchas soluciones al reordenar las prioridades asignadas a los objetivos. De entre estas soluciones, el tomador de decisiones puede elegir una solución que se ajuste mejor a sus preferencias. En la tabla 56 se listan las soluciones determinadas mediante el método de la programación por objetivos prioritarios para cada conjunto posible de prioridades. Por lo tanto, acomodos diferentes de prioridades originan distintas estrategias de publicidad.

Cuando un problema de programación por objetivos prioritarios tiene sólo dos variables de decisión, es posible determinar en forma gráfica la solución óptima. Por ejemplo, supon-

TABLA 56

Soluciones óptimas para Priceler determinadas mediante programación por objetivos prioritarios

Prioridades			Óptimo				
					Desviaciónes con respecto a		
La más alta	Segunda en importancia	La más baja	Valor x_1	Valor x_2	VAI	PBI	MAI
VAI	PBI	MAI	6	0	0	0	5
VAI	MAI	PBI	5	$\frac{5}{3}$	0	$\frac{5}{3}$	$\frac{10}{3}$
PBI	VAI	MAI	6	0	0	0	5
PBI	MAI	VAI	6	0	0	0	5
MAI	VAI	PBI	3	5	4	5	0
MAI	PBI	VAI	3	5	4	5	0

ga que MAI es el objetivo de mayor prioridad, PBI es el segundo en importancia y los VAI es el de menor prioridad. En la figura 14, se observa que el conjunto de puntos que satisfacen el objetivo de mayor prioridad (VAI) y la restricción del presupuesto está limitado por el triángulo *ABC*. Se trata ahora de buscar entre estos puntos la manera de satisfacer lo más posible el objetivo segundo en importancia (PBI). Infortunadamente, ningún punto en el triángulo *ABC* satisface el objetivo PBI. No obstante, en la figura se observa que entre todos los puntos que satisfacen el objetivo de mayor prioridad, el punto *C* (*C* es donde el objetivo MAI se satisface con exactitud y la restricción del presupuesto es activa) es el único punto que está lo más cerca de satisfacer el objetivo PBI. Al resolver en forma simultánea las ecuaciones

$$5x_1 + 4x_2 = 35 \qquad \text{(el objetivo MAI se cumple exactamente)}$$
$$100x_1 + 60x_2 = 600 \qquad \text{(restricción activa del presupuesto)}$$

se encuentra el punto $C = (3, 5)$. Por lo tanto, para este conjunto de prioridades, la solución con la programación por objetivos prioritarios es comprar 3 anuncios en el futbol y 5 en las telenovelas.

La programación por objetivos no es el único recurso usado para analizar problemas de toma de decisiones de objetivos múltiples cuando hay certidumbre. Otros métodos para tomar decisiones cuando los objetivos son varios y hay certidumbre se encuentran en Steuer (1985) y en Zionts y Wallenius (1976).

Solución de problemas de programación por objetivos prioritarios con la ayuda de LINDO y LINGO

Los estudiantes que no tienen acceso a un programa de computadora que resuelva problemas de programación por objetivos prioritarios puede recurrir a LINDO (o a cualquier otro paquete de PL) para solucionarlos. Con el fin de ilustrar cómo se puede usar LINDO para resolver este tipo de problemas de programación examinemos el ejemplo de Priceler con el conjunto original de prioridades (VAI seguido por PBI seguido por MAI).

Comenzamos pidiendo LINDO para reducir al mínimo la derivación del objetivo de mayor prioridad solucionando el PL:

$$\begin{aligned}
&\min z = s_1^- \\
&\text{s.a} \quad 7x_1 + 3x_2 + s_1^- - s_1^+ = 40 && \text{(restricción de VAI)} \\
&\quad\quad 10x_1 + 5x_2 + s_2^- - s_2^+ = 60 && \text{(restricción de PBI)} \\
&\quad\quad 5x_1 + 4x_2 + s_3^- - s_3^+ = 35 && \text{(restricción de MAI)} \\
&\quad\quad 100x_1 + 60x_2 \leq 600 && \text{(restricción del presupuesto)}
\end{aligned}$$

Todas las variables son no negativas

El objetivo 1 (VAI) se puede satisfacer, así que LINDO informa que hay un valor óptimo de $z = 0$. Ahora queremos llegar tan cerca como sea posible al objetivo 2 mientras se asegura que la desviación respecto al objetivo 1 se conserva en su nivel actual (0). Se usa una función objetivo de s_2^- (para minimizar el objetivo 2) y se suma la restricción $s_1^- = 0$ (para asegurar que el objetivo 1 todavía se cumple), y se pide a LINDO que resuelva

$$
\begin{array}{llr}
\min z = s_2^- & & \\
\text{s.a} & 7x_1 + 3x_2 + s_1^- - s_1^+ = 40 & \text{(restricción de VAI)} \\
& 10x_1 + 5x_2 + s_2^- - s_2^+ = 60 & \text{(restricción de PBI)} \\
& 5x_1 + 4x_2 + s_3^- - s_3^+ = 35 & \text{(restricción de MAI)} \\
& 100x_1 + 60x_2 \leq 600 & \text{(restricción del presupuesto)} \\
& s_1^- = 0 &
\end{array}
$$

Todas las variables son no negativas

Como es posible cumplir en forma simultánea con los objetivos 1 y 2, este PL también generará un valor óptimo de $z = 0$. Ahora se pretende alcanzar el objetivo 3 (MAI) tan cerca como sea posible, mientras se conservan las desviaciones respecto a los objetivos 1 y 2 en sus niveles actuales. Esto requiere que LINDO resuelva

$$
\begin{array}{llr}
\min z = s_3^- & & \\
\text{s.a} & 7x_1 + 3x_2 + s_1^- - s_1^+ = 40 & \text{(restricción de VAI)} \\
& 10x_1 + 5x_2 + s_2^- - s_2^+ = 60 & \text{(restricción de PBI)} \\
& 5x_1 + 4x_2 + s_3^- - s_3^+ = 35 & \text{(restricción de MAI)} \\
& 100x_1 + 60x_2 + s_3^- - s_3^+ \leq 600 & \text{(restricción del presupuesto)} \\
& s_1^- = 0 & \\
& s_2^- = 0 &
\end{array}
$$

Todas las variables son no negativas

Naturalmente, el editor de pantalla completa de LINDO (o LINGO) facilita ir de un paso del problema de programación por objetivos al siguiente paso. Para ir del paso i al paso $i + 1$, se modifica simplemente la función objetivo para minimizar la desviación desde el objetivo de mayor prioridad $i + 1$ y se suma una restricción que asegura que la desviación desde el objetivo i-ésimo de la más alta prioridad se mantiene en su nivel actual.

OBSERVACIONES

1 La solución óptima para este PL es $z = 5, x_1 = 6, x_2 = 0, s_1^- = 0, s_2^- = 0, s_3^- = 5, s_1^+ = 2, s_2^+ = 0, s_3^+ = 0$, lo cual va de acuerdo con la solución obtenida mediante el método de la programación por objetivos prioritarios. El valor de $z = 5$ indica que si los objetivos 1 y 2 se cumplen, entonces lo mejor que puede hacer Priceler es llegar a los 5 millones de exposiciones que cumplen con el objetivo 3.

2 Incidentalmente, suponga que sólo se pudo llegar a dos unidades de alcanzar el objetivo 1. Cuando se resuelve el segundo PL se tendrían que sumar las restricciones $s_1^- = 2$ (en lugar de $s_1^- = 0$).

3 La metodología de la programación por objetivos de esta sección se puede aplicar sin cambios cuando algunas o todas las variables de decisión están restringidas a ser enteros o variables 0-1 (véanse problemas 11, 12 y 14).

4 Si se usa LINGO, la metodología de la programación por objetivos de esta sección se puede aplicar sin cambios aun cuando la función objetivo o algunas de las restricciones sean no lineales.

PROBLEMAS

Grupo A

1 Determine en forma gráfica la solución de la programación por objetivos prioritarios para el ejemplo de Priceler con las prioridades siguientes:

a PBI es el objetivo con la prioridad más alta, seguido por MAI y luego por VAI.

b El objetivo con la prioridad más alta es VAI, seguido por PBI y luego por MAI.

c El objetivo con la prioridad más alta es VAI seguido por MAI y luego por PBI.

d El objetivo con la prioridad más alta es MAI, seguido por VAI y luego por PBI.

2 La compañía de computadoras Fruit está lista para hacer su compra anual de microprocesadores para sus computadoras. Fruit puede comprar microprocesadores (en lotes de 100) con tres proveedores. Cada microprocesador se clasifica, de acuerdo con su calidad, en excelente, bueno o mediocre. El año venidero, Fruit necesitará 5 000 microprocesadores excelentes, 3 000 microprocesadores buenos y 1 000 mediocres.

Las características de los microprocesadores comprados a cada proveedor se proporcionan en la tabla 57. Fruit presupuesta cada año 28 000 dólares para gastarlos en microprocesadores. Si la compañía no consigue suficientes microprocesadores de una calidad dada, entonces podría hacer un pedido especial de más microprocesadores a 10 dólares el excelente, 6 dólares el bueno y 4 dólares el mediocre. Fruit establece una penalización de 1 dólar por cada dólar que la cantidad pagada a los proveedores 1 a 3 sobrepase el presupuesto anual. Plantee y resuelva un PL con el cual Fruit minimice la penalización asociada con el cumplimiento de los requisitos de los microprocesadores en el año. Además aplique la programación por objetivos prioritarios para establecer una estrategia de compra. Demos a la limitación del presupuesto la más alta prioridad seguida por las restricciones de los microprocesadores excelentes, buenos y mediocres.

3 Highland Appliance tiene que determinar cuántos televisores a color y videocaseteras debe mantener en existencia. La compra de un televisor a color le cuesta a Highland 300 dólares, y la de una videocasetera, 200 dólares. Un televisor a color requiere 3 yardas cuadradas de espacio para el almacenamiento y una videocasetera necesita una yarda cuadrada de espacio. La venta de un televisor a color le proporciona a Highland una utilidad de 150 dólares, en tanto que la venta de una videocasetera da una utilidad de 100 dólares. Highland se ha fijado los objetivos siguientes (en orden de importancia):

Objetivo 1 Se puede gastar un máximo de 20 000 dólares en la compra de televisores a color y videocaseteras.
Objetivo 2 Highland debe ganar por lo menos 11 000 dólares en utilidades por la venta de televisores a color y videocaseteras.
Objetivo 3 Los televisores y las videocaseteras deben abarcar no más de 200 yardas cuadradas de espacio de almacenamiento.

Plantee un modelo de programación por objetivos prioritarios que Highland pueda usar para determinar cuántos televisores a color y videocaseteras tiene que pedir. ¿Cómo se modificaría el planteamiento por objetivos prioritarios si los objetivos de Highland tuvieran una utilidad de exactamente 11 000 dólares?

4 Una compañía elabora dos productos. La información pertinente para cada producto se proporciona en la tabla 58. La compañía tiene un objetivo de 48 dólares en utilidades e incurre en una penalización de 1 dólar por cada dólar que le falta para cumplir con este objetivo. Dispone de un total de 32 horas de mano de obra. Se incurre en una penalización de 2 dólares por cada hora de tiempo extra utilizada (mano de obra después de 32 horas). Por último, hay una penalización de 1 dólar por cada hora de mano de obra disponible que no se use. Las consideraciones de mercado exigen que se elaboren por lo menos 10 unidades del producto 2. Por cada unidad (de cualquier producto) que falte para cubrir la demanda, se fija una penalización de 5 dólares.

a Plantee un PL que se pueda usar para minimizar la penalización en que incurre la compañía.
b Suponga que la compañía establece (en orden de importancia) los objetivos siguientes:

Objetivo 1 Evitar la subutilización de la mano de obra.
Objetivo 2 Cumplir con la demanda del producto 1.
Objetivo 3 Cumplir con la demanda del producto 2.
Objetivo 4 No usar nada de tiempo extra.

Plantee y resuelva un modelo de programación por objetivos prioritarios para esta situación.

TABLA **57**

Proveedor	Características de un lote de 100 microprocesadores: Excelente	Bueno	Mediocre	Precio por 100 microprocesadores(dól.)
1	60	20	20	400
2	50	35	15	300
3	40	20	40	250

TABLA **58**

	Producto 1	Producto 2
Mano de obra requerida	4 h	2 h
Contribución a la utilidad	\$4	\$2

5† Deancorp produce embutidos mediante la mezcla de cabeza de res, lomo de cerdo, carne de oveja y agua. El costo por libra, grasa por libra, proteína por libra de estos ingredientes se da en la tabla 59. Deancorp necesita producir 100 lb de embutido y ha establecido los objetivos siguientes, listadas en orden de prioridad:

Objetivo 1 El embutido debe contener por lo menos 15% de proteína.
Objetivo 2 El embutido debe contener cuando mucho 8% de grasa.
Objetivo 3 El costo por libra de embutido no debe exceder 8 centavos.

Plantee un modelo de programación por objetivos prioritarios para Deancorp.

6‡ La firma contable Touche Young debe terminar tres trabajos durante el mes próximo. El trabajo 1 requiere 500 horas de labor, el trabajo 2 requiere 300 horas y el trabajo 3, de 100 horas. En la actualidad, la compañía tiene 5 socios, 5 empleados con amplia experiencia y 5 empleados jóvenes; todos trabajan hasta 40 horas al mes. La cantidad en dólares (por hora) que la compañía puede facturar depende del tipo de contador que se asigne a cada trabajo, como se indica en la tabla 60. (La X quiere decir que un empleado joven no tiene suficiente experiencia para desempeñar el trabajo 1.) Todos los trabajos se tienen que terminar. Touche Young ha fijado también los objetivos siguientes, que se listan en orden de prioridad:

Objetivo 1 La facturación mensual debe sobrepasar 68 000 dólares.
Objetivo 2 Se debe contratar cuando mucho un socio.
Objetivo 3 Se deben contratar cuando mucho tres empleados con experiencia.
Objetivo 4 Se deben contratar cuando mucho cinco empleados jóvenes.

TABLA **59**

	Cabeza	Lomo	Oveja	Humedad
Grasa (por lb)	0.05	0.24	0.11	0
Proteína (por lb)	0.20	0.26	0.08	0
Costo (en centavos)	0.12	9	8	0

†Basado en Steuer (1984).
‡Basado en Welling (1977).

TABLA 60

	Trabajo 1	Trabajo 2	Trabajo 3
Socio	160	120	110
Empleado con experiencia	120	90	70
Empleado joven	X	50	40

Formule un modelo de programación por objetivos prioritarios para esta situación.

7 Hay cuatro maestros en la Escuela de Contaduría de la Universidad Faber. Cada semestre, 200 estudiantes toman todos los siguientes cursos: mercadotecnia, finanzas, producción y estadística. La "efectividad" de cada maestro al enseñar su materia se da en la tabla 61. Cada maestro puede enseñar a un total de 200 alumnos durante el semestre. El director ha establecido el objetivo de obtener un nivel de efectividad en la enseñanza promedio de alrededor de 6 en cada curso. Las desviaciones respecto a este objetivo en cualquier curso se consideran de igual importancia. Plantee un modelo de programación por objetivos que se pueda usar para determinar los niveles de la enseñanza en el semestre.

Grupo B

8† La universidad Faber está admitiendo aspirantes a los cursos del 2008. Hay cuatro objetivos para ellos, que se listan en orden de prioridad:

Objetivo 1 Los estudiantes que ingresen deben ser por lo menos 5 000.
Objetivo 2 Los estudiantes que ingresen deben tener por lo menos una calificación promedio de 640 en la prueba de aptitudes.
Objetivo 3 Por lo menos, 25% de los estudiantes que ingresen deben ser de otros estados.
Objetivo 4 Por lo menos, 2 000 estudiantes de los que ingresen no deben ser *nerds*.

Los aspirantes que recibe Faber se clasifican según la tabla 62. Formule un modelo de programación por objetivos prioritarios con el que se pueda determinar cuántos aspirantes de cada tipo deben ser admitidos. Suponga que todos los aspirantes que son admitidos deciden asistir a esta universidad.

9‡ Wivco encara la demanda siguiente de *globots* durante los próximos cuatro trimestres: trimestre 1, 13 *globots*; trimestre 2, 14 *globots*; trimestre 3, 12 *globots*; trimestre 4, 15 *globots*. Los *globots* se pueden fabricar con mano de obra en el horario regular o con mano de obra en tiempo extra. La capacidad de producción (cantidad de *globots*) y costos de producción durante los próximos cuatro trimestres se proporcionan en la tabla 63. Wivco ha establecido los objetivos siguientes en orden de importancia:

TABLA 61

Maestro	Mercadotecnia	Finanzas	Produción	Estadística
1	7	5	8	2
2	7	8	9	4
3	3	5	7	9
4	5	5	6	7

†Basado en Lee y Moore, "University Admissions Planning" (1974).
‡Basedo en Lee y Moore, "Production Scheduling" (1974).

TABLA 62

Lugar de residencia	Calificación en la prueba de aptitudes	Núm. Nerds	Núm. no-Nerds
Mismo estado	700	1500	400
Mismo estado	600	1300	700
Mismo estado	500	500	500
Otro estado	700	350	450
Otro estado	600	400	400
Otro estado	500	400	600

Objetivo 1 Cumplir la demanda de cada trimestre a tiempo.
Objetivo 2 El inventario al final de cada trimestre no puede ser mayor de 3 unidades.
Objetivo 3 El costo de producción total debe mantenerse por abajo de 250 dólares.

Desarrolle un modelo de programación por objetivos prioritarios que se pueda usar para determinar el programa de producción para los siguientes cuatro trimestres. Suponga que al principio del primer trimestre hay un *globot* en inventario.

10 La tienda de discos Ricky emplea ya 5 empleados de tiempo completo y tres empleados de medio tiempo. La carga de trabajo normal es de 40 horas a la semana para los empleados de tiempo completo y de 20 horas a la semana para los empleados de medio tiempo. Cada empleado de tiempo completo recibe 6 dólares por hora por trabajar hasta 40 horas a la semana y puede vender 5 discos por hora. Un empleado de tiempo completo que trabaja tiempo extra, recibe 10 dólares por hora. Los empleados de medio tiempo reciben 3 dólares por hora y pueden vender 3 discos por hora. A Ricky le cuesta 6 dólares comprar un disco y vende cada disco en 9 dólares. Ricky tiene gastos fijos a la semana de 500 dólares. Ha establecido los objetivos siguientes por semana, listados en orden de prioridad:

Objetivo 1 Vender por lo menos 1 600 discos por semana.
Objetivo 2 Tener una utilidad de por lo menos 2 200 dólares por semana.
Objetivo 3 Los empleados de tiempo completo deben trabajar cuando mucho 100 horas de tiempo extra.
Objetivo 4 Para aumentar el sentido de seguridad en el trabajo, se debe minimizar la cantidad de horas que cada empleado de tiempo completo trabaja después de las 40 horas.

Establezca un modelo de programación por objetivos prioritarios que se pueda usar para determinar cuántas horas por semana debe trabajar cada empleado.

TABLA 63

	Horario regular		Tiempo extra	
Trimestre	Capacidad	Costo/Unidad	Capacidad	Costo/Unidad
1	9	$4	5	$6
2	10	$4	5	$7
3	11	$5	5	$8
4	12	$6	5	$9

11 Gotham City pretende determinar el tipo y la ubicación de las instalaciones recreativas que se construirán en la década próxima. Se está pensando en cuatro tipos de instalaciones: campos de golf, albercas, gimnasios y canchas de tenis. Hay seis lugares para ello. Si se construye un campo de golf, tiene que ser en el sitio 1 o en el sitio 6. Otras instalaciones se pueden erigir en los sitios 2 a 5. El terreno disponible (en miles de pies cuadrados) en cada sitio se señala en la tabla 64.

El costo de la construcción de cada instalación (en miles de dólares), el mantenimiento anual (en miles de dólares) por cada instalación y el terreno (en miles de pies cuadrados) que requiere cada instalación se indican en la tabla 65.

La cantidad de días-usuario (en miles) por cada tipo de instalación depende del lugar donde se construya. Esta relación de dependencia se proporciona en la tabla 66.

a Considere el siguiente conjunto de prioridades:

Prioridad 1 Uso límite del terreno en cada sitio respecto al terreno disponible.
Prioridad 2 Los costos de construcción no deben exceder 1.2 millones de dólares.
Prioridad 3 Los días-usuario deben exceder 200 000.
Prioridad 4 Los costos de mantenimiento al año no deben ser mayores de 200 000 dólares.

Por lo que se refiere a este conjunto de prioridades, utilice la programación por objetivos prioritarios para determinar el tipo y ubicación de las instalaciones recreativas en Gotham City.

b Considere el siguiente conjunto de prioridades:

Prioridad 1 Uso del terreno límite en cada sitio respecto al terreno disponible.
Prioridad 2 La cantidad de días-usuario deben ser mayores que 200 000.
Prioridad 3 Los costos de construcción no deben exceder 1.2 millones de dólares.
Prioridad 4 Los costos de mantenimiento al año no deben sobrepasar 200 000 dólares.

Por lo que toca a este conjunto de prioridades, utilice la programación por objetivos prioritarios para determinar el tipo y ubicación de las instalaciones recreativas en Gotham City.[†]

12 Una pequeña compañía aeroespacial planea ocho proyectos:

Proyecto 1 Desarrollar una instalación de pruebas automatizadas.

TABLA **64**

	Sitio 2	3	4	5
Terreno	70	80	95	120

TABLA **65**

Sitio	Costo de construcción	Costo de mantenimiento	Terreno requerido
Golf	340	80	No relevante
Natación	300	36	29
Gimnasio	840	50	38
Canchas de tenis	85	17	45

[†]Basado en Taylor y Keown (1984).

TABLA **66**

Sitio	1	2	3	4	5	6
Golf	31	X	X	X	X	27
Natación	X	25	21	32	32	X
Gimnasio	X	37	29	28	38	X
Canchas de tenis	X	20	23	22	20	X

Proyecto 2 Asignar un código de barras a todo el inventario y maquinaria de la compañía.
Proyecto 3 Introducir un sistema CAD/CAM.
Proyecto 4 Comprar un torno y un sistema nuevos para eliminar rebabas.
Proyecto 5 Instituir el sistema de manufactura flexible.
Proyecto 6 Instalar una red de área local.
Proyecto 7 Desarrollar la simulación de inteligencia artificial.
Proyecto 8 Establecer una iniciativa de administración de calidad total.

Cada proyecto se clasificó según cinco atributos: rendimiento de la inversión (RDI), costo, mejoramiento de la productividad, trabajadores necesarios y grado de riesgo tecnológico. Los valores se dan en la tabla 67.

La compañía ha fijado los siguientes cinco objetivos (listados en orden de prioridad):

Objetivo 1 Alcanzar un rendimiento de la inversión de por lo menos 3 250 dólares.
Objetivo 2 Costo límite de 1 300 dólares.
Objetivo 3 Alcanzar un mejoramiento en la productividad de por lo menos 6.
Objetivo 4 Limitar la fuerza de trabajo a 108.
Objetivo 5 Limitar el riesgo tecnológico a un total de 4.

Utilice la programación por objetivos prioritarios para determinar qué proyectos se deben emprender.

13 Apenas fue elegido el nuevo presidente y ya se establecieron los objetivos económicos siguientes (listados en orden descendente de prioridad):

Objetivo 1 Equilibrar el presupuesto (esto significa ingresos por lo menos iguales a los costos).
Objetivo 2 Recortar los gastos cuando mucho en 150 mil millones de dólares.
Objetivo 3 Aumentar cuando mucho 550 mil millones de dólares en impuestos de los ricos.
Objetivo 4 Aumentar cuando mucho 350 mil millones de dólares en impuestos de los pobres.

En la actualidad, el gobierno gasta un millón de millones al año. El ingreso puede aumentar de dos maneras: mediante un impuesto a la gasolina y un impuesto sobre la renta. Usted debe determinar:

G = impuesto por galón (en centavos).
LTR = % de impuesto cargado en los primeros 30 000 dólares de renta.
HTR = % de impuesto cargado sobre cualquier ingreso obtenido por arriba de los 30 000 dólares.
C = recorte en gastos (en miles de millones).

Si el gobierno escoge G, LTR y HTR, entonces se elevan los ingresos que se muestran en la tabla 68 (en miles de millones). Naturalmente, el impuesto sobre los ingresos mayores de 30 000 dólares debe ser por lo menos igual al impuesto sobre los primeros 30 000 dólares de ingresos. Encuentre un mode-

TABLA 67

	Proyecto							
	1	2	3	4	5	6	7	8
ROI (dól.)	2 070	456	670	350	495	380	1 500	480
Costo (dól.)	900	240	335	700	410	190	500	160
Mejoramiento de la productividad	3	2	2	0	1	0	3	2
Fuerza de trabajo necesaria	18	18	27	36	42	6	48	24
Grado riesgo	3	2	4	1	1	0	2	3

TABLA 68

	Ingresos bajos	Ingreso alto
Impuesto a la gasolina	G	.5G
Impuesto sobre la renta hasta 30 000 dólares	20LTR	5LTR
Impuesto sobre la renta por arriba de los 30 000 dólares	0	15HTR

lo de programación por objetivos prioritarios para ayudar al presidente a lograr sus metas.

14 Las computadoras HAL deben determinar cuál de los siete proyectos de investigación y desarrollo (I y D) empezar. Para cada uno de los proyectos hay cuatro valores que son de interés:

a Valor presente neto (VNP en millones de dólares) del proyecto.

b Tasa de crecimiento anual en ventas generadas por el proyecto.

c Probabilidad de que el proyecto tenga éxito.

d Costo (en millones de dólares) del proyecto.

La información pertinente se proporciona en la tabla 69. HAL estableció los cuatro siguientes objetivos:

Objetivo 1 VPN = el valor presente neto de todos los proyectos seleccionados debe ser por lo menos de 200 millones de dólares.

TABLA 69

Proyecto	VNA (en millones)	Tasa de cecimoento anual	Probabiliad de éxito	Costo (en millones)
1	40	20	0.75	220
2	30	16	0.70	140
3	60	12	0.75	280
4	45	8	0.90	240
5	55	18	0.65	300
6	40	18	0.60	200
7	90	19	0.65	440

Objetivo 2 La probabilidad promedio de éxito para todos los proyectos elegidos debe ser por lo menos de 0.75.

Objetivo 3 La tasa de crecimiento promedio de todos los proyectos escogidos debe ser por lo menos de 15%.

Objetivo 4 El costo total de todos los proyectos seleccionados debe ser cuando mucho de mil millones.

Determine qué proyectos deben ser elegidos mediante programación por objetivos prioritarios para los conjuntos siguientes de prioridades:

Conjunto de prioridades 1 2>>>4>>>1>>>3
Conjunto de prioridades 2 1>>>3>>>4>>>2

4.17 Uso de Solver de Excel para solucionar PL

Excel tiene la capacidad de resolver problemas de programación lineal (y con frecuencia no lineal). En esta sección se ilustra cómo utilizar el Solver para Excel† para determinar la solución óptima para el problema de la dieta de la sección 3.4 y el ejemplo del inventario de la sección 3.10.

La clave para resolver un PL en una hoja de cálculo es establecer una que siga la pista a todo lo de interés (costos o utilidades, uso de recursos, etc.). Luego identifique las celdas de interés que puedan ser distintas. Éstas se denominan **celdas de cambio** (Changing Cells). Después de definir dichas celdas, identifique la celda que contiene su función objetivo como **celda blanco** (Target Cell). Enseguida identifique las restricciones e indique a Solver que resuelva el problema. En este momento, la solución óptima para el problema se colocará en la hoja de cálculo.

†Para activar Solver para Excel por primera vez, seleccione Tools (Herramientas) y luego Add-Ins (Complementos). Marque la casilla de Solver; esto hace que Excel abra Solver siempre que usted marque Tools (Herramientas) y luego Solver.

Uso de Solver para Excel para solucionar el problema de la dieta

Diet1.xls

En el archivo Diet1.xls se desarrolló un modelo en hoja de cálculo para el problema de la dieta (ejemplo 6 del capítulo. 3). Para empezar (véase figura 15) se introducen los encabezados por cada tipo de alimento en B3:E3. Se introducen los valores de ensayo para la cantidad de cada alimento consumido en el intervalo B4:E4. Por ejemplo, en la figura 15 se indica que se está considerando comer tres barras de chocolate, cuatro bolas de helado de crema de chocolate, cinco botellas de bebida de cola y seis rebanadas de pastel de queso con piña. Para ver si la dieta de la figura 15 es una dieta "óptima", se tiene que determinar su costo, así como las calorías, el chocolate, azúcar y grasa que proporciona. En el intervalo B5:E5 se introduce el costo por unidad de cada alimento disponible. Luego se calcula el costo de la dieta en la celda F5.

Se podría calcular el costo de la dieta en la celda F5 con la fórmula:

$$=B4 \cdot B5 + C4 \cdot C5 + D4 \cdot D5 + E4 \cdot E5$$

pero es más fácil introducir la fórmula

$$=\text{SUMPRODUCT(B\$4:E\$4, B5:E5)}$$

La función =SUMPRODUCT requiere dos intervalos como datos de entrada. La primera celda en el primer intervalo se multiplica por la primera celda del segundo intervalo; luego, la segunda celda del primer intervalo se multiplica por la segunda celda en el segundo intervalo; y así sucesivamente. Luego se suman todos estos productos. En esencia, la función =SUMPRODUCT copia la idea de los productos escalares de los vectores analizados en la sección 2.1. Por lo tanto, la función =SUMPRODUCT calcula el costo total como $(3)(50) + 4(20) + 5(30) + 6(80) = 860$ centavos en la celda F5.

Las calorías de cada alimento se introducen en el intervalo B6:E6; el contenido de chocolate en B7:E7; el contenido de azúcar en B8:E8, y el contenido de grasa en B9:E9. Al copiar la fórmula de F5 al intervalo de celdas F6:F9 ahora se calculan las calorías, chocolate, azúcar y grasa contenida en la dieta definida por los valores en B4:E4. Obsérvese que la función =SUMPRODUCT facilita la creación de varias restricciones al introducir una fórmula y usar el comando copy (copiar).

Las cantidades mínimas necesarias por día de cada nutriente se listan en el intervalo de celdas H6:H9. En la figura 15 se puede ver que la dieta actual es factible (cumple con las cantidades diarias necesarias de cada nutriente) y cuesta 8.70 dólares. A continuación se explica cómo usar Solver para hallar la solución óptima del problema de la dieta.

Paso 1 En el menú Tools (Herramientas), seleccione Solver. Aparecerá la ventana de diálogo de la figura 16.

Paso 2 Desplace el ratón hasta la porción de la celda blanco (Set Target Cell) de la ventana de diálogo, y dé un clic (o escriba la dirección de la celda) en la *target cell* (celda blanco) (costo total en la celda F5) y seleccione Min. De esta manera se le indica a Solver que debe minimizar el costo total.

Paso 3 Mueva el ratón hacia la parte de *By Changing Cells* (las celdas de cambio) de la ventana de diálogo y dé un clic en las celdas de cambio (B4:E4). Esto le indica a Solver que se puede cambiar la cantidad consumida de cada alimento.

Paso 4 Dé un clic en el botón *Add* para añadir las restricciones. Aparecerá la pantalla de la figura 17. Desplácese a la parte de la ventana de diálogo de *Add Constraint* y seleccione

	A	B	C	D	E	F	G	H
1			Feasible					
2			solution to Diet Problem					
3		Brownie	Choc IC	Cola	Pine Cheese	Totals		Required
4	Eaten	3	4	5	6			
5	Cost	50	20	30	80	860		
6	Calories	400	200	150	500	5750	>=	500
7	Chocolate	3	2	0	0	17	>=	6
8	Sugar	2	2	4	4	58	>=	10
9	Fat	2	4	1	5	57	>=	8

FIGURA 15

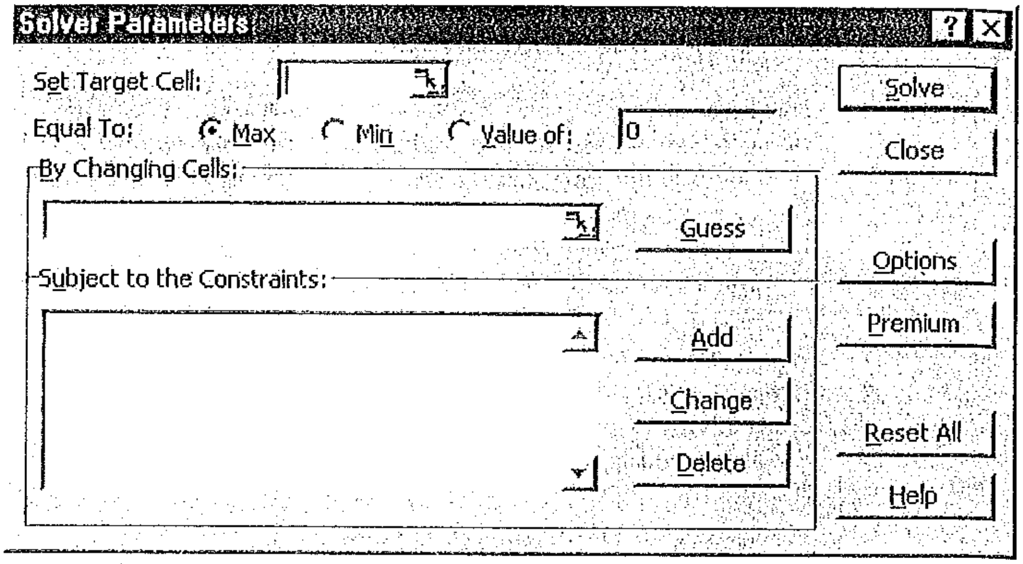


FIGURA 16

F6:F9. Luego muévase al cuadro con la flecha hacia abajo y seleccione >=. Por último, dé un clic donde dice *Constraints* y seleccione H6:H9. Escoja OK, porque ya no hay más restricciones. Si usted necesita agregar más restricciones, seleccione *Add.* Desde la ventana principal de Solver, usted puede cambiar una restricción si selecciona *Change*, o bien, borrar una limitación si selecciona *Delete.*

Ya se generaron cuatro restricciones. Solver asegurará que las *Changing Cells* se escojan de tal manera que F6>=H6, F7>=H7, F8>=H8, y F9>=H9. En pocas palabras, la dieta se escogerá para asegurar que se consuman suficientes calorías, chocolate, azúcar y grasa.

La ventana de Solver debe ser como la de la figura 18.

Paso 5 Antes de resolver el problema, es necesario indicarle a Solver que todas las *Changing Cells* deben ser no negativas. También se requiere indicarle que se tiene un modelo lineal. Si no se indica que el modelo es lineal, entonces Solver no sabrá que debe usar el método simplex para resolver el problema, y podría obtener una respuesta incorrecta. Se consiguen estos requisitos mediante la selección de *Options*. Entonces aparece la pantalla de la figura 19. Marque en ella el cuadro *Assume non-Negative*, lo cual segura que todas las celdas de cambio (*changing cells*) serán no negativas. Marque el cuadro *Assume Linear Model*, lo cual asegura que Solver utilizará el método simplex para resolver el PL. En ocasiones, Solver no identificará al PL como un modelo lineal si el PL tiene escalas poco apropiadas (un PL tanto con cantidades grandes como pequeñas presentes en la función objetivo, segundos miembros o las restricciones). Al marcar la casilla *Use Automatic Scaling* se reducen al mínimo las oportunidades de que un PL con escalas poco apropiadas sea interpretado como un modelo no lineal. Incidentalmente, *Max Time* es el tiempo máximo que Solver trabajará antes de avisar al usuario respecto a la terminación del procedimiento de solución. *Iterations* es el número máximo de pivoteos con simplex que Solver efectuará antes de preguntar al usuario si continúa con el procedimiento de la solución. El parámetro *Precision* señala cuánto "error" se tolera antes de decidir si una restricción no se satisface. Por ejemplo, con una precisión de 0.001, se consideraría que una celda de cambio (*chan-*

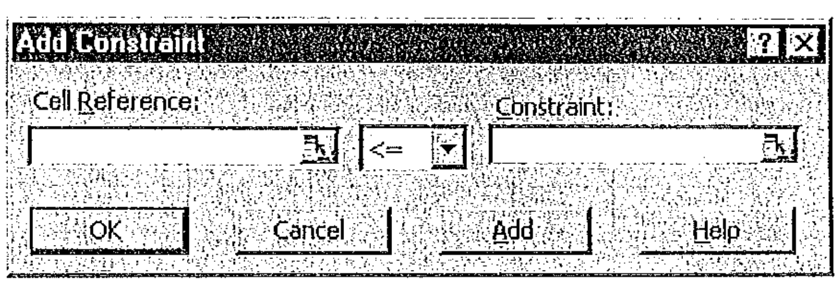


FIGURA 17

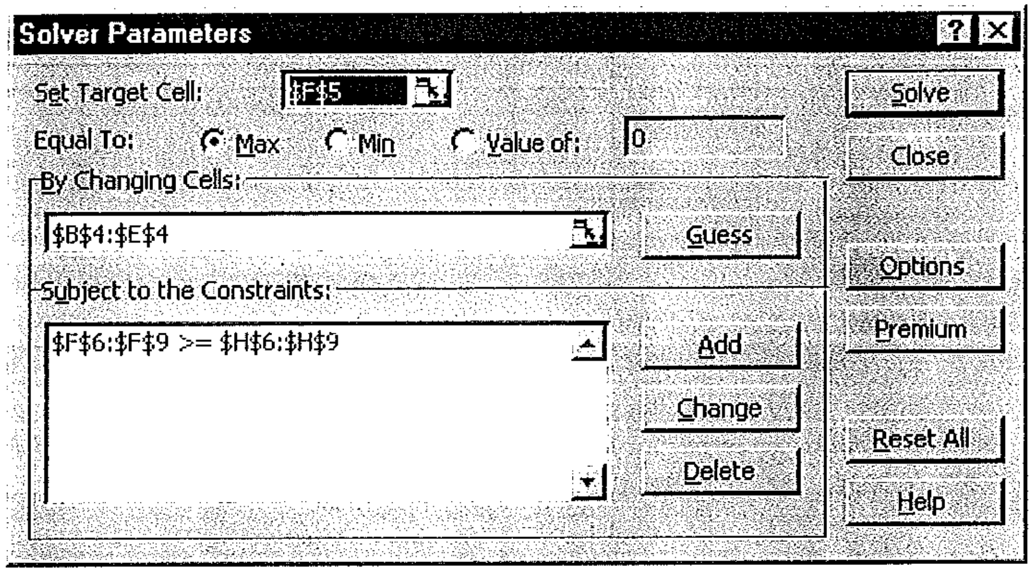


FIGURA **18**

ging cell) con un valor de -0.0009 satisface una restricción no negativa. Los parámetros tolerancia y convergencia se tratan en el capítulo 8.

Paso 6 Luego de seleccionar *OK* en el cuadro de opciones de Solver, se elige *Solve*. Solver obtiene la solución óptima mostrada en la figura 20.

Al igual que con LINDO, Solver señala que el costo mínimo es 90 centavos. El costo mínimo se consigue comiendo nada de barras de chocolate, aunque sí 3 onzas de helado de chocolate, una botella de bebida de cola y nada de pastel de queso con piña.

Solución del ejemplo de Sailco mediante Solver

Sailco.xls

Ahora se desarrollará una hoja de cálculo (Sailco.xls) para resolver el ejemplo de Sailco (ejemplo 12 del capítulo 13). Véase la figura 21. Se requiere seguir con atención el inventario inicial, el inventario final y los costos. Observe que para cada mes

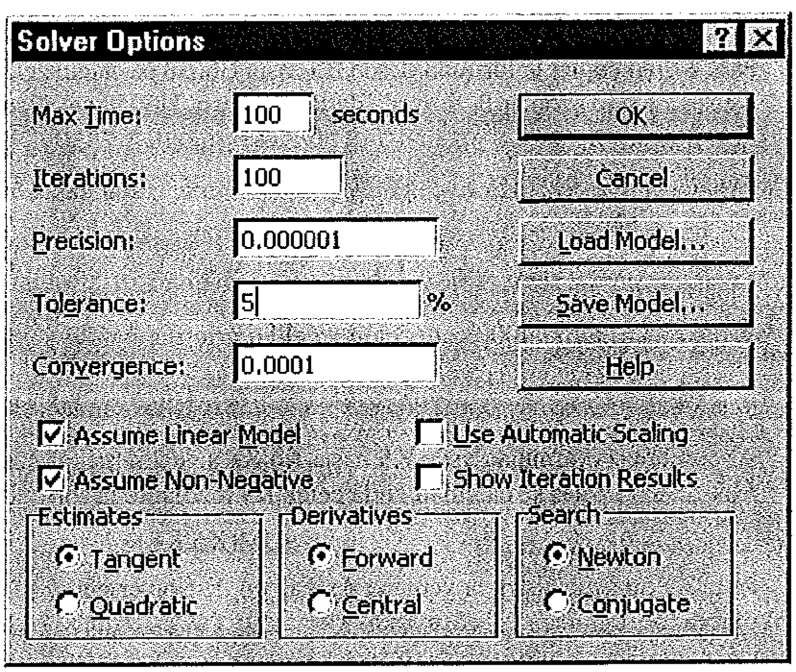


FIGURA **19**

FIGURA 20

	A	B	C	D	E	F	G	H
1			Optimal Solution					
2			to the Diet Problem					
3		Brownie	Choc IC	Cola	Pine Cheese	Totals		Required
4	Eaten	0	3	1	0			
5	Cost	50	20	30	80	90		
6	Calories	400	200	150	500	750	>=	500
7	Chocolate	3	2	0	0	6	>=	6
8	Sugar	2	2	4	4	10	>=	10
9	Fat	2	4	1	5	13	>=	8

FIGURA 21

	A	B	C	D	E	F	G	H	I	J	K
1			Optimal solution					RT unit cost	$ 400.00		
2			to Sailco problem					OT unit cost	$ 450.00		
3								Unit Holding cost	$ 20.00		
4	Month	Beg Inventory	OT Production	RT Production		RT Capacity	Demand	Ending Inventory			Monthly Cost
5	1	10	0	40	<=	40	40	10	>=	0	$ 16,200.00
6	2	10	10	40	<=	40	60	0	>=	0	$ 20,500.00
7	3	0	35	40	<=	40	75	0	>=	0	$ 31,750.00
8	4	0	0	25	<=	40	25	0	>=	0	$ 10,000.00
9										Total Cost	$ 78,450.00

$$\text{Costo mensual} = 400(\text{producción en horario regular}) + 450(\text{producción con tiempo extra}) + 20(\text{costo por retener las unidades})$$

$$\text{Inventario final} = \text{inventario inicial} + \text{producción mensual} - \text{demanda mensual}$$

Paso 1 Introducir los costos unitarios en I1:I3, las capacidades mensuales en horario regular en F5:F8, demandas en G5:G8 e inventario del mes 1 en B5.

Paso 2 Introducir los valores de ensayo de la producción de cada mes en horario regular y con tiempo extra en C5:D8.

Paso 3 Determinar el inventario final del mes 1 en H5 mediante la fórmula

$$=B5 + C5 + D5 - G5$$

Esto representa la siguiente relación:

$$\text{Inventario final} = \text{inventario al principio} + \text{producción mensual} - \text{demanda mensual}$$

Paso 4 Asignar el inventario inicial del mes 2 al inventario final del mes 1 al ingresar en la celda B6 la fórmula

$$=H5$$

Paso 5 Al copiar la fórmula desde B5 hasta B6:B8 se calcula el inventario inicial para los meses 2 a 4. Al copiar la fórmula desde H5 hasta H6:H8 se calcula el inventario final para los meses 2 a 4.

Paso 6 En la celda K5 se calcula el costo del mes 1 mediante la fórmula

$$=\$I\$1*D5 + C5*\$I\$2 + \$I\$3*H5$$

Lo anterior representa el hecho de que el costo de cada mes está dado por

$$\text{Costo mensual} = 400(\text{producción en el horario regular}) + 450(\text{producción con tiempo extra}) + 20(\text{costo por conservar las unidades})$$

Al copiar esta fórmula desde K5 hasta K6:K8 se calculan los costos para los meses 2 a 4. Los costos totales se calculan en la celda K9 mediante la fórmula

$$=SUM(K5:K8)$$

Paso 7 Se llena ahora la ventana de diálogo de Solver como se ilustra en la figura 22. El objetivo es minimizar el costo total (celda K9). Las *changing cells* son producción con tiempo extra y en horario regular (C5:D8). Es necesario asegurarse de que la producción de ca-

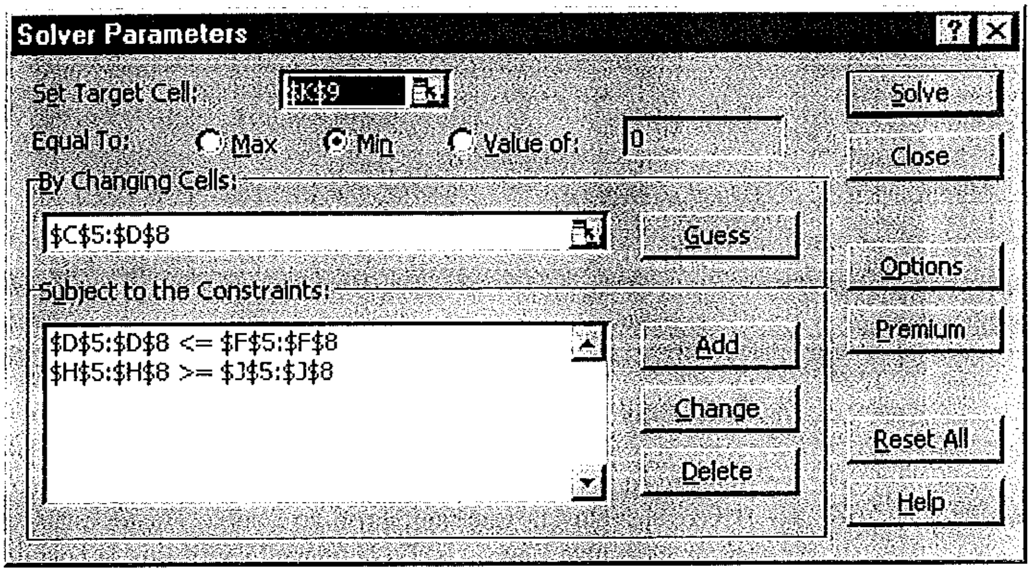


FIGURA 22

da mes en el horario regular es cuando mucho 40(D5:D8 <=F5:F8). Por último, si se obliga a que el inventario final de cada mes sea no negativo (H5:H8 > = J5:J8) se asegura que la demanda mensual se cumple a tiempo. Se marca *Assume Linear Model, Assume Non-Negative* y *Use Automatic Scaling* en *Options*. Después de elegir Solver se llega a la solución óptima que se ilustra en la figura 21. Se logra un costo mínimo de 78 450 dólares con la fabricación de 40 unidades elaboradas en el horario regular en los meses 1 a 3, 25 unidades producidas en horario regular durante el mes 4, 10 unidades producidas con tiempo extra durante el mes 2 y 35 unidades manufacturadas con tiempo extra durante el mes 3.

Uso del valor de opción

Recuerde que el costo mínimo era de 78 450 dólares en el problema de Sailco. Suponga que se desea determinar una solución que proporcione exactamente un costo de 90 000 dólares. Entonces se utiliza el valor de opción de Solver. Se llena el cuadro de diálogo de Solver como se muestra en la figura 23 (véase la hoja titulada *Cost of $90,000* en el archivo Sailco.xls).

Sailco.xls

Solver da la solución de la figura 24. Observe que Solver encontró una solución factible que tiene un costo total exactamente de 90 000 dólares.

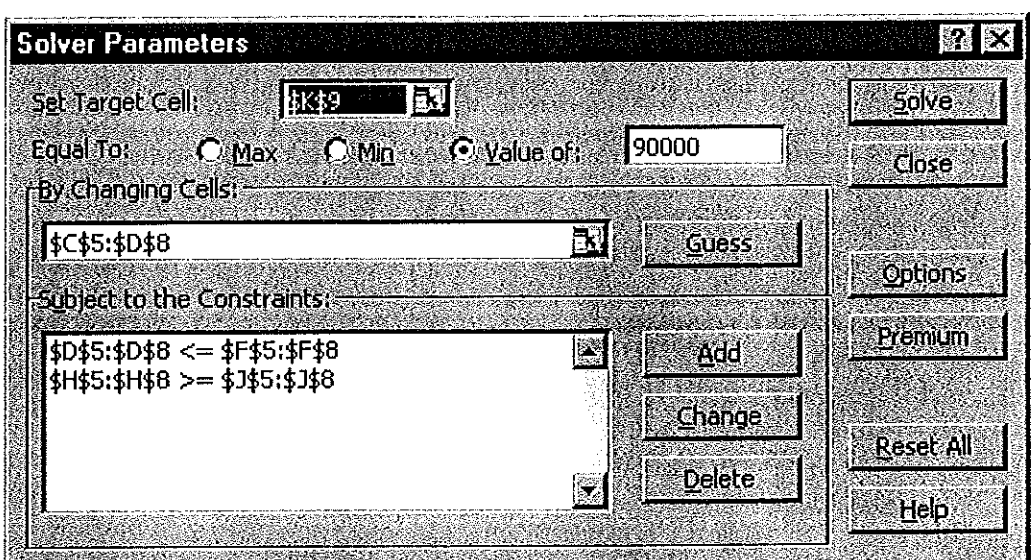


FIGURA 23

FIGURA 24

	A	B	C	D	E	F	G	H	I	J	K
1			Optimal solution					RT unit cost	$ 400.00		
2			to Sailco problem					OT unit cost	$ 450.00		
3								Unit Holding cost	$ 20.00		
4	Month	Beg Inventory	OT Production	RT Production		RT Capacity	Demand	Ending Inventory			Monthly Cost
5	1	10	179.090909	0	<=	40	40	149.0909091	>=	0	$ 83,572.73
6	2	149.0909	0	0	<=	40	60	89.09090909	>=	0	$ 1,781.82
7	3	89.09091	0	0	<=	40	75	14.09090909	>=	0	$ 281.82
8	4	14.09091	0	10.909091	<=	40	25	0	>=	0	$ 4,363.64
9										Total Cost	$ 90,000.00

Solver y PL no factibles

Bevco.xls

Recuerde que si por lo menos 36 mg de vitamina C se requieren, entonces el problema de Bevco (ejemplo 4 de este capítulo) es no factible. Hemos establecido este problema en Solver en el archivo Bevco.xls. En la figura 25 se ilustra la hoja de cálculo, y la ventana de Solver en la figura 26.

Cuando se elige Solver, aparece el mensaje que se presenta en la figura 27. Esto indica que el PL no tiene solución factible.

Solver y los PL no acotados

Breadco.xls

El ejemplo 3 de este capítulo es un PL no acotado. El archivo Breadco.xls (figura 28) contiene un planteamiento de este PL. En la figura 29 se presenta una ventana de Solver para el ejemplo de Breadco. Cuando se elige *Solve* aparece el mensaje de la figura 30.

	A	B	C	D	E	F
1						
2	Infeasible LP					
3						
4					Total Cost	
5		Soda	Juice		3 0	
6	Amount	0	10			
7	Unit cost	2	3			
8				Available		Needed
9	Sugar	0.5	0.25	2.5	>=	4
10	Vitamin C	1	3	30	>=	36
11	Total oz.	1	1	10	=	10

FIGURA 25

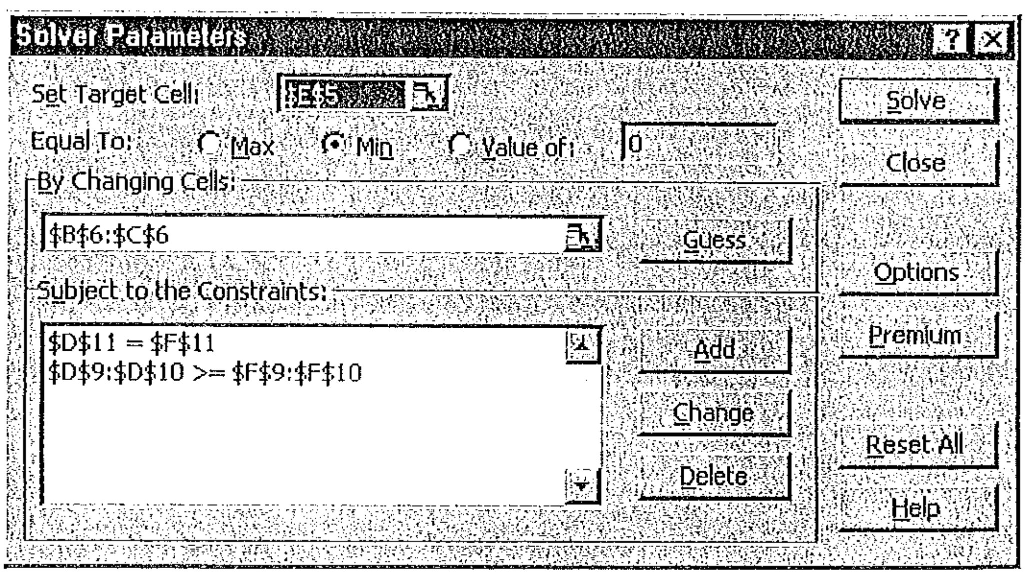


FIGURA 26

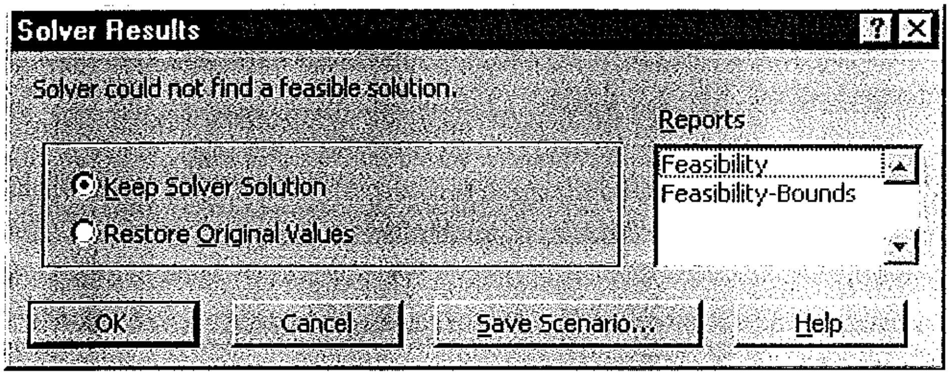


FIGURA **27**

	A	B	C	D	E	F	G	H
1	**Unbounded LP**							
2								
3			FB Baked	SD Baked	Yeast bought	Flour bought		Originally we have
4			5	0	0	20		
5		Price or cost	36	30	3	4		
6		Yeast needed	1	1				5
7		Flour needed	6	5				10
8								
9		Profit	100					
10								
11			Used		Available			
12		Yeast	5	<=	5			
13		Flour	30	<=	30			

FIGURA **28**

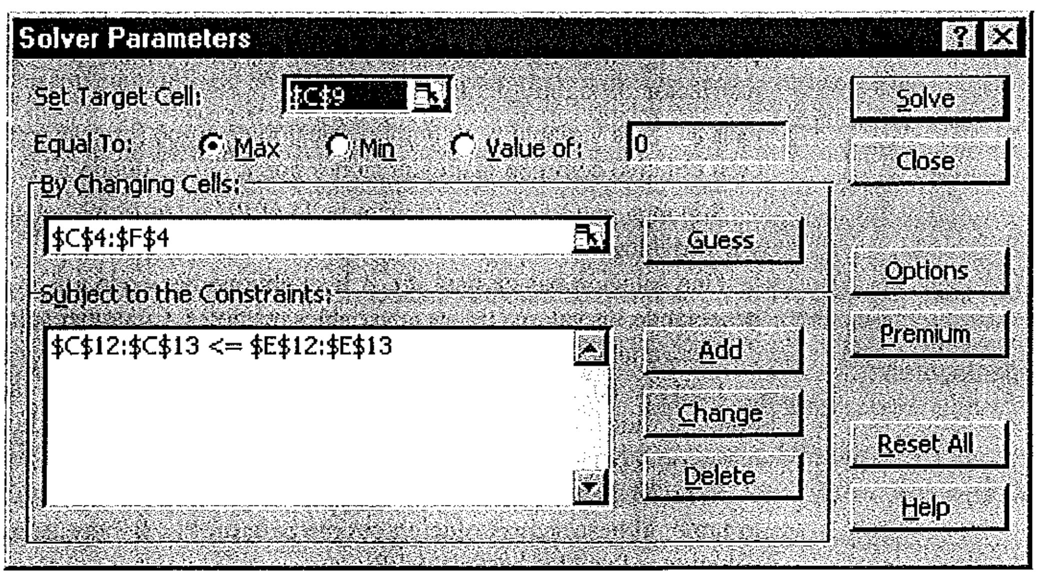


FIGURA **29**

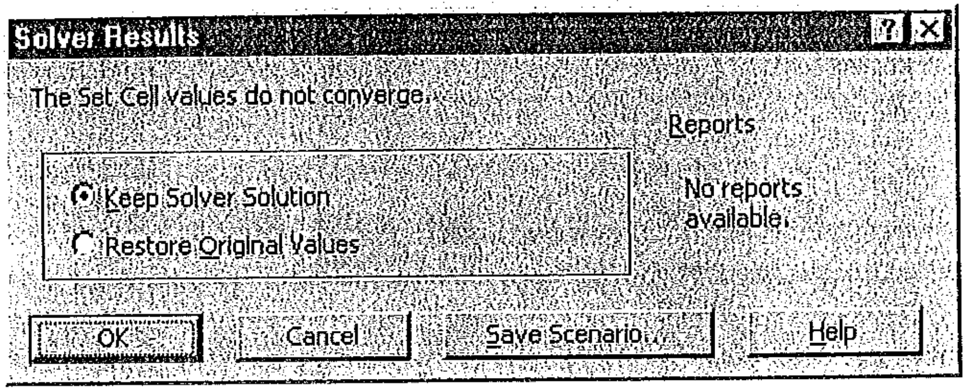


FIGURA 30

El mensaje "*Set Cell values do not converge*" ("los valores de las celdas no convergen") quiere decir que el PL es no acotado; es decir, hay valores de las celdas de cambio que satisfacen todas las restricciones y originan una utilidad arbitrariamente grande.

PROBLEMAS

Grupo A

Encuentre la solución óptima de los problemas siguientes con Solver para Excel.

1 Problema 2 de la sección 3.4.

2 Ejemplo 7 del capítulo 3.

3 Ejemplo 11 del capítulo 3.

4 Problema 3 de la sección 3.10.

5 Ejemplo 14 de la sección 3.12.

Grupo B

6 Problema 4 de la sección 3.11.

7 Problema 5 de la sección 3.11.

8 Problema 3 de la sección 3.12.

9 Problema 5 de la sección 3.12.

RESUMEN

Preparación de un PL para resolverlo mediante simplex

Un PL está en la **forma estándar** si todas las restricciones son restricciones de igualdad y todas las variables son no negativas. Para transformar un PL en forma estándar se efectúa lo siguiente:

Paso 1 Si la i-ésima restricción es una restricción $\leq$, entonces se convierte en una restricción de igualdad mediante la suma de una variable de holgura s_i y la restricción de signo $s_i \geq 0$.

Paso 2 Si la i-ésima restricción es una restricción $\geq$, entonces se convierte en una restricción de igualdad mediante la sustracción de una variable de excedente e_i y la suma de la restricción de signo $e_i \geq 0$.

Paso 3 Si la variable x_i no tiene restricción de signo (nrs), se reemplaza x_i tanto en la función objetivo como en las restricciones por $x_i' - x_i''$, donde $x_i' \geq 0$ y no $x_i'' \geq 0$.

Suponga que tras haber transformado el PL en la forma estándar tiene m restricciones y n variables.

Una solución básica para $A\mathbf{x} = \mathbf{b}$ se obtiene haciendo a $n - m$ variables iguales a 0 y despejando los valores de las m variables restantes. Cualquier solución básica en la cual todas las variables son no negativas es una **solución factible básica** (sfb) del PL.

Para cualquier PL hay un punto extremo único de la región factible del PL que corresponde a cada sfb. Asimismo, por lo menos una sfb corresponde a cada punto extremo de la región factible.

Si un PL tiene una solución óptima, entonces hay un punto extremo que es óptimo. Por lo tanto, al buscar una solución óptima para un PL, la búsqueda se podría restringir a las soluciones factibles básicas del PL.

El algoritmo simplex

Si el PL está en la forma estándar y es muy evidente una sfb, entonces el algoritmo simplex (para un problema de maximización) procede como sigue:

Paso 1 Si todas las variables no básicas tienen coeficiente no negativo en el renglón 0, entonces la sfb actual es óptima. Si algunas variables en el renglón 0 tienen coeficiente negativo, entonces elija la variable con el coeficiente más negativo en el renglón 0 para que entre a la base.

Paso 2 Para cada restricción en la cual la variable entrante tiene un coeficiente positivo, se calcula la razón siguiente:

$$\frac{\text{Segundo miembro de la restricción}}{\text{Coeficiente de la variable de entrada en la restricción}}$$

Cualquier restricción que alcance el valor más pequeño de esta razón es la ganadora de la prueba del cociente. Aplique OER para hacer de la variable entrante una variable básica en cualquier restricción que gane la prueba del cociente. Vuelva al paso 1.

Si el PL (en un problema de maximización) es no acotada, entonces con el paso del tiempo se alcanza un arreglo en el cual una variable no básica tiene un coeficiente negativo en el renglón 0 y coeficiente no positivo en cada restricción. De otro modo (exceptuando la ocurrencia rarísima de *los ciclos*), el algoritmo simplex encontrará una solución óptima para el PL.

Si una sfb no es muy evidente, entonces se tiene que usar el método de la gran M o el método simplex de dos fases para llegar a una sfb.

Método de la gran M

Paso 1 Modifique las restricciones de tal manera que el segundo miembro o lado derecho de cada una de ellas sea no negativo.

Paso 1′ Identifique las restricciones que son ahora (después del paso 1) restricciones $=$ o $\geq$. En el paso 3 se suma una variable artificial a cada una de estas restricciones.

Paso 2 Convierta cada restricción de desigualdad en la forma estandar.

Paso 3 Si (después de haber completado el paso 1) la restricción i es una restricción $\geq$ o $=$, entonces sume una variable artificial a_i y la restricción de signo $a_i \geq 0$.

Paso 4 Sea M un número positivo muy grande. Si el PL es un problema de minimización, entonces sume (por cada variable artificial) Ma_i a la función objetivo. Para un problema de maximización sume $-Ma_i$.

Paso 5 Puesto que cada variable artificial estará en la base inicial, se deben eliminar del renglón 0 antes de empezar el simplex. Si todas las variables artificiales son iguales a 0 en la solución óptima, entonces se ha encontrado la solución óptima del problema original. Si algunas variables artificiales son positivas en la solución óptima, entonces el problema original es no factible.

Método de las dos fases

Paso 1 Modifique las restricciones de tal manera que el segundo miembro de cada restricción sea no negativo.

Paso 1′ Identifique las restricciones que sean ahora (después del paso 1) restricciones $=$ o $\geq$. En el paso 3 se sumará una variable artificial a cada una de dichas restricciones.

Paso 2 Convierta las restricciones de desigualdad en la forma estándar.

Paso 3 Si (después del paso 1′) la restricción i es una restricción $\geq$ o $=$, entonces sume una variable artificial a_i y la restricción de signo $a_i \geq 0$.

Paso 4 Por lo pronto, ignore la función objetivo del PL original. Resuelva un PL cuya función objetivo sea min w' = (suma de todas las variables artificiales). Esto recibe el nombre de **PL de la Fase I.**

Como cada $a_i \geq 0$, luego de resolver el PL de la Fase I se tiene uno de los tres casos siguientes:

Caso 1 El valor óptimo de w' es mayor que cero. En este caso, el PL original no tiene solución factible.

Caso 2 El valor óptimo de w' es igual a cero y ninguna variable artificial está en la base óptima de la fase I. En este caso, elimine todas las columnas del tableau óptimo de la fase I que correspondan a las variables artificiales, y combine la función objetivo original con las restricciones del tableau óptimo de la fase I. Así se origina el **PL de la Fase II**. La solución óptima para este PL y el PL original son iguales.

Caso 3 El valor óptimo de w' es igual a cero y por lo menos una variable artificial está en la base óptima de la fase I. Es posible determinar la solución óptima del PL original en este caso si al final de la fase I se suprimen del arreglo óptimo de la fase I todas las variables artificiales no básicas y cualquier variable proveniente del problema original que tenga coeficiente negativo en el renglón 0 del tableau óptimo de la fase I.

Solución de problemas de minimización

Para resolver un problema de minimización mediante el simplex, elija como la variable entrante la variable no básica del renglón 0 cuyo coeficiente sea el más positivo. Un tableau, o forma canónica, es óptimo si cada variable del renglón 0 tiene coeficiente no positivo.

Soluciones óptimas alternativas

Si una variable no básica tiene coeficiente 0 en el renglón 0 de un tableau óptimo, y la variable no básica puede ser pivoteada en la base, el PL podría tener **soluciones óptimas alternativas.** Si dos soluciones factibles básicas son óptimas, entonces cualquier punto sobre el segmento de recta que une las dos soluciones factibles básicas óptimas es también una solución óptima para el PL.

Variables que no tienen restricción de signo (nrs)

Si se reemplaza una variable nrs x_i con $x_i' - x_i''$, la solución óptima del PL tendrá x_i', x_i'' o tanto x_i' como x_i'' iguales a cero.

PROBLEMAS DE REPASO

Grupo A

1 Mediante el algoritmo simplex determine *dos* soluciones óptimas del PL siguiente:

$$\begin{aligned} \max z &= 5x_1 + 3x_2 + x_3 \\ \text{s.a} \quad & x_1 + x_2 + 3x_3 \leq 6 \\ & 5x_1 + 3x_2 + 6x_3 \leq 15 \\ & x_3, x_1, x_2 \geq 0 \end{aligned}$$

2 Determine con el algoritmo simplex la solución óptima del PL siguiente:

$$\begin{aligned} \min z &= -4x_1 + x_2 \\ \text{s.a} \quad & 3x_1 + x_2 \leq 6 \\ & -x_1 + 2x_2 \leq 0 \\ & x_1, x_2 \geq 0 \end{aligned}$$

3 Utilice el método de la gran M y el método de las dos fases para encontrar la solución óptima del PL siguiente:

$$\begin{aligned} \max z &= 5x_1 - x_2 \\ \text{s.a} \quad 2x_1 + x_2 &= 6 \\ x_1 + x_2 &\leq 4 \\ x_1 + 2x_2 &\leq 5 \\ x_1, x_2 &\geq 0 \end{aligned}$$

4 Encuentre la solución óptima del PL siguiente por medio del algoritmo simplex:

$$\begin{aligned} \max z &= 5x_1 - x_2 \\ \text{s.a} \quad x_1 - 3x_2 &\leq 1 \\ x_1 - 4x_2 &\leq 3 \\ x_1, x_2 &\geq 0 \end{aligned}$$

5 Encuentre la solución óptima del PL siguiente por medio del algoritmo simplex:

$$\begin{aligned} \min z &= -x_1 - 2x_2 \\ \text{s.a} \quad 2x_1 + x_2 &\leq 5 \\ x_1 + x_2 &\leq 3 \\ x_1, x_2 &\geq 0 \end{aligned}$$

6 Encuentre la solución óptima del PL siguiente por medio del método de la gran M y el método de las dos fases:

$$\begin{aligned} \max z &= x_1 + x_2 \\ \text{s.a} \quad 2x_1 + x_2 &\geq 3 \\ 3x_1 + x_2 &\leq 3.5 \\ x_1 + x_2 &\leq 1 \\ x_1, x_2 &\geq 0 \end{aligned}$$

7 Aplique el algoritmo simplex para determinar *dos* soluciones óptimas del PL siguiente. ¿Cuántas soluciones óptimas tiene este PL? Encuentre una tercera solución óptima.

$$\begin{aligned} \max z &= 4x_1 + x_2 \\ \text{s.a} \quad 2x_1 + 3x_2 &\leq 4 \\ x_1 + x_2 &\leq 1 \\ 4x_1 + x_2 &\leq 2 \\ x_1, x_2 &\geq 0 \end{aligned}$$

8 Use el algoritmo simplex para determinar la solución óptima del PL siguiente:

$$\begin{aligned} \max z &= 5x_1 + x_2 \\ \text{s.a} \quad 2x_1 + x_2 &\leq 6 \\ x_1 - x_2 &\leq 0 \\ x_1, x_2 &\geq 0 \end{aligned}$$

9 Encuentre la solución óptima del PL siguiente por medio del método de la gran M y el método de las dos fases.

$$\begin{aligned} \min z &= -3x_1 + x_2 \\ \text{s.a} \quad x_1 - 2x_2 &\geq 2 \\ -x_1 + x_2 &\geq 3 \\ x_1, x_2 &\geq 0 \end{aligned}$$

10 Suponga que se pudieran manufacturar 10 tipos de muebles en el problema de los muebles Dakota. ¿Cuántos tipos de muebles (cuando mucho) se tendrían que fabricar para obtener una solución óptima?

11 Considere el PL siguiente:

$$\begin{aligned} \max z &= 10x_1 + x_2 \\ \text{s.a} \quad x_1 &\leq 1 \\ 20x_1 + x_2 &\leq 100 \\ x_1, x_2 &\geq 0 \end{aligned}$$

a Encuentre todas las soluciones factibles básicas de este PL.

b Demuestre que cuando el simplex se aplica para resolver el PL toda solución básica se tiene que examinar antes de encontrar la solución óptima.

Mediante la generalización de este ejemplo, Klee y Minty (1972) construyeron (para $n = 2, 3, \ldots$) un PL con n variables de decisión y n restricciones para las cuales el algoritmo simplex examina $2^n - 1$ soluciones factibles básicas antes de encontrar la solución óptima. Por lo tanto, existe un PL con 10 variables y 10 restricciones para las cuales el algoritmo simplex requiere $2^{10} - 1 = 1\,023$ pivoteos para hallar la solución óptima. Por fortuna estos PL "patológicos" rara vez se presentan en las aplicaciones prácticas.

12 Productco elabora tres productos. Todos requieren mano de obra, madera y pintura. Los recursos necesarios, precio unitario y costo variable (exclusivo de la materia prima) para cada producto se proporcionan en la tabla 70. Se dispone en la actualidad de 900 horas de mano de obra, 1 550 galones de pintura y 1 600 pies tablón de madera. Además, se puede comprar mano de obra adicional a 6 dólares la hora; la pintura extra costaría 2 dólares/galón y la madera adicional, 3 dólares/pie tablón. Para los dos conjuntos siguientes de prioridades utilice programación por objetivos prioritarios para determinar un plan de producción óptimo. Para el conjunto 1:

Prioridad 1 Obtener utilidad de por lo menos 10 500 dólares.
Prioridad 2 No comprar mano de obra adicional.
Prioridad 3 No comprar más pintura.
Prioridad 4 No comprar más madera.

Para el conjunto 2:

Prioridad 1 No adquirir más mano de obra.
Prioridad 2 Obtener utilidad de por lo menos 10 500 dólares.
Prioridad 3 No comprar más pintura.
Prioridad 4 No comprar más madera.

13 Los empleos en la Universidad de Indiana (UI) se clasifican según tres factores:

Factor 1 Complejidad de las funciones.
Factor 2 Instrucción necesaria.
Factor 3 Demanda mental, o visual, o ambas.

Para cada empleo en la UI, lo necesario para cada factor se califica según una escala de 1 a 4, en donde un 4 en el factor 1 representa alta complejidad de funciones; un 4 en el factor 2 significa alta escolaridad, y un 4 en el factor 3 quiere decir alta demanda mental, o visual, o ambas.

TABLA 70

Producto	Mano de obra	Madera	Pintura	Precio(dól.)	Costo variable (dól.)
1	1.5	2	3	26	10
2	3	3	2	28	6
3	2	4	2	31	7

La UI desea encontrar una fórmula para determinar el grado de cada empleo. Para hacerlo asignará un valor en puntos a la calificación de cada factor que requiere un empleo. Por ejemplo, suponga que el nivel 2 del factor 1 da un total de puntos de 10; el nivel 3 del factor 2 genera un total de puntos de 20 y que el nivel 3 del factor 3 origina un valor en puntos de 30. Entonces, un empleo con estos requisitos tendría un total de puntos de 10 + 20 + 30. El salario por hora del empleo es igual a la mitad de su total de puntos.

La universidad tiene dos objetivos (listados en orden de prioridad) al establecer los puntos dados a cada nivel de cada factor del empleo.

Objetivo 1 Cuando se incrementa el nivel de un factor en 1, los puntos se deben incrementar por lo menos en 10. Por ejemplo, el nivel 2 del factor 1 debe ganar por lo menos 10 puntos más que el nivel 1 del factor 1. El objetivo 1 es minimizar la suma de desviaciones respecto a estos requisitos.

Objetivo 2 Por lo que se refiere a los empleos de referencia de la tabla 71, el total de puntos real para cada empleo debe acercarse tanto como sea posible al total de puntos listado en la tabla. El objetivo 2 es minimizar la suma de las desviaciones absolutas de los totales de puntos respecto a las calificaciones deseadas.

Aplique la programación por objetivos prioritarios para alcanzar los totales de puntos apropiados. ¿Qué salario se debe pagar en un empleo con niveles de habilidad de 3 para cada factor?

14 Una clínica hospital de pacientes externos ejecuta cuatro tipos de operaciones. La utilidad por operación, así como los minutos de rayos X y el tiempo de laboratorio utilizados se dan en la tabla 72. La clínica tiene 500 habitaciones privadas y 500 cuartos de cuidados intensivos. Las operaciones tipo 1 y tipo 2 requieren que el paciente permanezca en el cuarto de cuidados intensivos durante un día, en tanto que para las operaciones tipo 3 y tipo 4 es necesario que el paciente permanezca en una habitación privada durante un día. Se practican todos los días por lo menos 100 operaciones de cada tipo en el hospital. El hospital tiene los objetivos siguientes:

Objetivo 1 Lograr una utilidad diaria de por lo menos 100 000 dólares.

Objetivo 2 Utilizar a lo sumo 50 h al día de tiempo de rayos X.

Objetivo 3 Utilizar a lo más 40 h al día de tiempo de laboratorio.

TABLA 71

	Nivel de factor			
Empleo	1	2	3	Calificación deseada
1	4	4	4	105
2	3	3	2	93
3	2	2	2	75
4	1	1	2	68

TABLA 72

	Tipo de operación			
	1	2	3	4
Utilidad (dólares)	200	150	100	80
Tiempo de rayos (minutos)	6	5	4	3
Tiempo de laboratorio (minutos)	5	4	3	2

El costo por desviación unitaria respecto a cada objetivo es como se indica:

Objetivo 1 Costo de 1 dólar por cada dólar con el que se incumpla el objetivo de la utilidad.

Objetivo 2 Costo de 10 dólares por cada hora con la que se incumpla el objetivo del tiempo de rayos X.

Objetivo 3 Costo de 8 dólares por cada hora por la que se incumpla el objetivo del laboratorio.

Establezca un modelo de programación por objetivos para minimizar el costo diario en que se incurre por el incumplimiento de los objetivos del hospital.

Grupo B

15 Considere un problema de maximización con el arreglo óptimo que aparece en la tabla 73. La solución óptima para este PL es $z = 10, x_3 = 3, x_4 = 5, x_1 = x_2 = 0$. Determine la sfb segunda en importancia para este PL. (*Sugerencia*: Demuestre que la segunda mejor solución tiene que ser una sfb que es un pivote alejado de la solución óptima.)

16 Un senderista está considerando llevar consigo dos tipos de objetos en un viaje por el campo. El objeto 1 pesa a_1 lb y el objeto 2 pesa a_2 lb. El senderista obtiene un beneficio de c_1 unidades con cada objeto tipo 1, y con cada objeto tipo 2 logra un beneficio de c_2 unidades. La mochila puede llegar a contener objetos que pesen a lo más b lb.

a Si el senderista puede llevar una cantidad fraccionaria de objetos en su viaje, formula un PL que maximice el beneficio.

b Demuestre que si

$$\frac{c_2}{a_2} \geq \frac{c_1}{a_1}$$

entonces, el senderista puede maximizar el beneficio llenando la mochila con $\frac{b}{a_2}$ objetos tipo 2.

c ¿Cuáles de las suposiciones de la programación lineal se incumplen con este planteamiento del problema del senderista?

17 Usted recibe el tableau mostrado en la tabla 74 para un problema de maximización. Proporcione las condiciones de las incógnitas a_1, a_2, a_3, b y c que hacen que los enunciados siguientes sean verdaderos:

a La solución actual es óptima.

b La solución actual es óptima y hay soluciones óptimas alternas.

c El PL es no acotado (en este inciso suponga que $b \geq 0$).

18 Suponga que se ha obtenido el tableau de la tabla 75 para un problema de maximización. Establezca condiciones sobre a_1, a_2, a_3, b, c, y c_2 que se requieren para que los enunciados siguientes sean verdaderos:

a La solución actual es óptima y hay soluciones óptimas alternas.

b La solución básica actual no es una solución factible básica.

TABLA 73

z	x_1	x_2	x_3	x_4	Ld
1	2	1	0	0	10
0	3	2	1	0	3
0	4	3	0	1	5

TABLA 74

z	x_1	x_2	x_3	x_4	x_5	ld
1	$-c$	2	0	0	0	10
0	-1	a_1	1	0	0	4
0	a_2	-4	0	1	0	1
0	a_3	3	0	0	1	b

TABLA 75

z	x_1	x_2	x_3	x_4	x_5	x_6	ld
1	c_1	c_2	0	0	0	0	10
0	4	a_1	1	0	a_2	0	b
0	-1	-5	0	1	-1	0	2
0	a_3	-3	0	0	-4	1	3

c La solución básica actual es una sfb degenerada.

d La solución básica actual es factible, pero el PL es no acotado.

e La solución básica actual es factible, pero el valor de la función objetivo puede ser mejorado al reemplazar x_6 como una variable básica con x_1.

19 Suponga que estamos resolviendo un problema de maximización y que la variable x_r casi deja la base.

a ¿Cuál es el coeficiente de x_r en el renglón 0 actual?

b Demuestre que después de que el pivoteo actual se ejecuta, el coeficiente de x_r en el renglón 0 no puede ser menor que cero.

c Explique por qué una variable que ya dejó la base en un pivoteo dado no puede volver a entrar a la base en el siguiente pivoteo.

20 Una compañía de autobuses opina que necesitará las cantidades siguientes de conductores durante cada uno de los cinco años siguientes: año 1, 60 conductores; año 2, 70 conductores; año 3, 50 conductores; año 4, 65 conductores; año 5, 75 conductores. Al inicio de cada año, la compañía tiene que decidir cuántos conductores se deben contratar o despedir. Cuesta 4 000 dólares contratar un conductor y 2 000 dólares despedir a uno. El salario de un conductor es de 10 000 dólares al año. Al empezar el año 1, la compañía tiene 50 conductores. Un conductor contratado al inicio del año puede ayudar a cumplir las necesidades del año actual y se le paga salario completo por el año actual. Determine un PL que minimice el salario y los costos de la contratación y del despido en los siguientes cinco años.

21 Los Zapateros de América pronostican la demanda siguiente para cada uno de los seis meses próximos: mes 1, 5 000 pares; mes 2, 6 000 pares; mes 3, 5 000 pares; mes 4, 9 000 pares; mes 5, 6 000 pares; mes 6, 5 000 pares. Un zapatero hace un par de zapatos en 15 minutos. Todos los zapateros trabajan 150 horas por mes más hasta 40 horas por mes de tiempo extra. Un zapatero recibe un salario regular de 2 000 dólares al mes más 50 dólares por hora de tiempo extra. Al inicio de cada mes, la empresa puede contratar o despedir trabajadores. A la compañía le cuesta 1 500 dólares contratar un trabajador y 1 900 dólares despedir a uno. El costo por mes por retener un par de zapatos es 3% del costo de producción de un par de zapatos con mano de obra y horario regular. (La materia prima para un par de zapatos cuesta 10 dólares.) Plantee un PL con el que se puedan minimizar los costos para cumplir (a tiempo) las demandas de los próximos seis meses. Al inicio del mes 1 la empresa tiene 13 trabajadores.

22 El condado de Monroe pretende determinar dónde ubicar la estación de bomberos del condado. Las ubicaciones de las cuatro ciudades principales del condado se dan en la figura 31. La ciudad 1 está en (10, 20); la ciudad 2 está en (60, 20); ciudad 3 está en (40, 30); ciudad 4 está en (80, 60). El promedio de incendios en la ciudad 1 es de 20 por año; el de la ciudad 2 es de 30; el de la ciudad 3 es de 40; el de la ciudad 4 es de 25 incendios. El condado quiere construir la estación de bomberos en un lugar que minimice la distancia promedio que una motobomba debe recorrer para acudir al incendio. Como la mayoría de caminos va de oriente a poniente o de norte a sur, se supone que la motobomba sólo puede hacer lo mismo. Por lo tanto, si la estación de bomberos se ubicara en (30, 40) y un incendio ocurriera en la ciudad 4, la motobomba tendría que recorrer $(80 - 30) + (60 - 40) = 70$ millas hasta el incendio. Aplique la programación lineal para determinar dónde se debe ubicar la estación de bomberos. (*Sugerencia*: Si la estación de bomberos se ubicara en el punto (x, y) y hay una ciudad en el punto (a, b) defina las variables e, w, n, s (este, oeste, norte, sur) que satisfacen las ecuaciones $x - a = w - e$ y $y - b = n - s$. Debe ser fácil obtener la formulación del PL correcto.)

23† Durante la temporada de futbol americano de 1972, los Delfines de Miami, los Bills de Buffalo y los Jets de Nueva York jugaron los partidos mostrados en la tabla 76. Suponga que según las bases de estos partidos queremos clasificar a los tres equipos. Sea M = la clasificación de Miami, J = la clasificación de los Jets y b = la clasificación de los Bills. Dados los valores de M, J y B, usted pronosticaría que cuando, por ejemplo, los Bills jugaran contra Miami, era de esperarse que Miami ganaría por $M - B$ puntos. Por lo tanto, su predicción habría sido errónea para el primer partido Miami-Bills por $|M - B - 1|$ puntos. Muestre cómo se puede aplicar la programación lineal para determinar la clasificación de cada equipo que minimice la suma de los errores de pronóstico para todos los juegos.

Al terminar la temporada, este método se usa para determinar las clasificaciones del futbol colegial y el basquetbol colegial. ¿Qué problemas se pueden prever si este método se usara para clasificar equipos al principio de la temporada?

24 Durante los cuatro trimestres próximos, Dorian Auto debe cumplir (a tiempo) la demanda siguiente de automóviles: trimestre 1, 4 000; trimestre 2, 2 000; trimestre 3, 5 000; trimestre 4, 1 000. Al inicio del trimestre 1 hay 300 automóviles en existencia y la compañía tiene la capacidad de producir a lo más 3 000 automóviles por trimestre. La compañía puede cambiar, al empezar cada trimestre, la capacidad de producción en un automóvil. Aumentar cada trimestre la capacidad de produc-

FIGURA 31

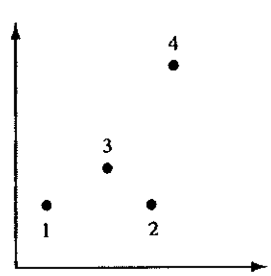


TABLA 76

Miami	Bills	Jets
27	—	17
28	—	24
24	23	—
30	16	—
—	24	41
—	3	41

†Basado en Wagner (1954).

ción cuesta 100 dólares. Cuesta 50 dólares por trimestre mantener un automóvil de la capacidad de producción (incluso si éste no se usa durante el trimestre actual). El costo variable por fabricar un automóvil es de 2 000 dólares. El costo por retener un automóvil es de 150 dólares por unidad, y se fija contra el inventario final de cada trimestre. Es necesario que al final del trimestre 4 la capacidad de la planta sea de por lo menos 4 000 automóviles. Plantee un PL con el que se minimice el costo total en que se incurre durante los próximos cuatro trimestres.

25 Ghostbusters, Inc., exorciza, es decir, libra a uno de los fantasmas. La compañía recibirá durante los siguientes tres meses la siguiente cantidad de llamadas telefónicas de personas que desean desembarazarse de sus fantasmas: enero, 100 llamadas; febrero, 300 llamadas; marzo, 200 llamadas. La empresa cobra 800 dólares por cada fantasma exorcizado durante el mes en que el cliente llama. No es necesario atender las llamadas durante el mes en que son hechas, pero si se atiende una llamada un mes después de que es hecha, entonces la compañía pierde 100 dólares en la clientela futura, y si una llamada se atiende a los dos meses de haber sido hecha, la firma pierde 200 dólares en la clientela. Cada empleado de Ghostbusters puede exorcizar 10 fantasmas al mes. Cada empleado recibe un salario de 4 000 dólares por mes. Al empezar enero, la compañía tiene 8 trabajadores. Los empleados pueden ser contratados o capacitados (en el tiempo 0) a un costo de 5 000 dólares por trabajador. El despido de trabajadores cuesta 4 000 dólares por empleado. Establezca un PL con el que se pueda maximizar la utilidad de Ghostbusters (ingresos menos costos) en los tres meses próximos. Suponga que todas las llamadas tienen que ser atendidas a fines de marzo.

26 Carco utiliza robots para manufacturar automóvil. La demanda siguiente se debe cumplir (no necesariamente a tiempo, pero todos los automóviles deben ser entregados al finalizar el trimestre 4): trimestre 1, 600; trimestre 2, 800; trimestre 3, 500; trimestre 4, 400. Carco tiene dos robots al empezar el trimestre. Los robots pueden comprarse al inicio de cada trimestre, pero sólo un máximo de dos por trimestre. Cada robot puede producir 200 automóviles por trimestre. La compra de un robot cuesta 5 000 dólares. Cada robot incurre en un costo de 500 dólares por mantenimiento cada trimestre (incluso si no se utiliza para fabricar automóviles). Los robots se pueden vender también al inicio de cada trimestre en 3 000 dólares. Al final de cada trimestre se incurre en un costo de 200 dólares por automóvil que se conserva. Si se acumula alguna de las demandas, entonces se incurre en un costo de 300 dólares por automóvil por cada trimestre que la demanda deja de surtirse.

Al final del trimestre 4, Carco debe tener por lo menos dos robots. Determine un PL con el que se minimice el costo total en que se incurre al cumplir la demanda de automóviles de los próximos cuatro trimestres.

27 Suponga que hemos encontrado un tableau óptimo para un PL, y que la sfb para dicho tableau es no degenerada. También suponga que hay una variable no básica en el renglón 0 con coeficiente cero. Demuestre que el PL tiene más de una solución óptima.

28 Suponga que la sfb para un tableau óptimo es degenerada, y que una variable no básica en el renglón 0 tiene coeficiente cero. Muestre con un ejemplo que cualquiera de los dos casos siguientes puede ocurrir:

Caso 1 El PL tiene más de una solución óptima.
Caso 2 El PL tiene una única solución óptima.

29 Usted es el alcalde de Gotham City, por lo que debe determinar una política fiscal para la ciudad. Se utilizan cinco tipos de impuestos para conseguir ingresos:

a Impuestos sobre la propiedad. Sea p = tasa porcentual del impuesto sobre la propiedad.

b Un impuesto sobre las ventas de todos los productos, excepto alimentos, medicamentos y bienes duraderos. Sea s = tasa porcentual del impuesto sobre las ventas.

c Un impuesto sobre la venta de bienes duraderos. Sea d = tasa porcentual del impuesto sobre las ventas de bienes duraderos.

d Un impuesto sobre la venta de gasolina. Sea g = tasa porcentual del impuesto sobre la venta de gasolina.

e Un impuesto sobre la venta de alimentos y medicamentos. Sea f = impuesto sobre la venta de alimentos y medicamentos.

La ciudad consta de tres grupos de personas: bajos ingresos (BI), ingresos medios (IM) y altos ingresos (AI). La cantidad de ingresos (en millones de dólares) proveniente de cada grupo al fijar un impuesto particular en un nivel de 1% se proporciona en la tabla 77.

Por ejemplo, un impuesto de 3% sobre las ventas de bienes duraderos generará 360 millones de dólares provenientes de las personas de bajos ingresos. Su política fiscal debe satisfacer las siguientes:

Restricción 1 La carga fiscal sobre las personas de IM no puede sobrepasar 2.8 miles de millones de dólares.
Restricción 2 La carga fiscal sobre las personas de AI no puede sobrepasar 2.4 miles de millones de dólares.
Restricción 3 Los ingresos totales reunidos tienen que exceder el nivel actual de 6.5 miles de millones de dólares.
Restricción 4 s tiene que estar entre 1 y 3%.

Dadas estas restricciones, el alcalde estableció los tres objetivos siguientes:

Objetivo P Mantener la tasa de impuestos sobre la propiedad menor a 3%.
Objetivo BI Limitar la carga fiscal sobre las personas de BI a 2 mil millones de dólares.
Objetivo suburbios Si la carga fiscal resulta demasiado alta, 20% de las personas BI, 20% de las personas IM y 40% de las personas de AI podrían cambiarse a las afueras de la ciudad. Suponga que esto sucederá si su carga fiscal total excede 1.5 mil millones de dólares. Para desanimar este éxodo, el objetivo suburbios es mantener la carga fiscal total sobre estas personas abajo de 2.5 mil millones de dólares.

Aplique la programación lineal para determinar una política fiscal óptima si los objetivos del alcalde tienen el siguiente conjunto de prioridades:

LI >>> P >>> Suburbios†

TABLA 77

	p	s	d	g	f
BI	900	300	120	30	90
IM	1 200	400	100	20	60
AI	1 000	250	60	10	40

†Basado en Chrisman, Fry, Reeves, Lewis, y Weinstein (1989).

APÉNDICE A Comandos de los menús e instrucciones de LINDO

Comandos de los menús

Se puede tener acceso a los comandos u órdenes de LINDO desde un menú conveniente similar a los de otros programas de Windows. El menú principal está formado por seis submenús a lo largo de la parte superior de la pantalla que señalan los diferentes comandos. Cuando usted da un clic en uno de los submenús —File, Edit, Solve, Reports, Window o Help, aparece un menú que se extiende hacia abajo con los diversos comandos. Usted puede seleccionar los comandos de la misma manera que en los programas de Windows dando un clic sobre la orden con el ratón o escribiendo la letra subrayada en el nombre del comando cuando el submenú apropiado está resaltado. Muchos comandos tienen también asignada una combinación de teclas (F2, Ctrl+Z, etc.). Como una facilidad más, también se puede tener acceso a algunos de los comandos que se usan con mayor frecuencia por medio de un icono que se localiza en una barra de herramientas en la parte superior de la pantalla. Diversos comandos de menús se describen brevemente y se da la lista de las combinaciones de teclas aplicables y los iconos en las secciones siguientes.

Menú Archivo (File Menu)

Los comandos del menú *File* le permiten a usted manipular en varios modos sus archivos de datos de LINDO. Puede usar este menú para abrir, cerrar, guardar e imprimir archivos, así como ejecutar varias tareas exclusivas de LINDO. A continuación se presenta una descripción de los comandos de *File*.

COMANDO			DESCRIPCIÓN
New	F2		Crea una ventana nueva para introducir datos.
Open	F3		Abre un archivo existente. Los cuadros de diálogo le permiten seleccionar entre varios tipos de archivos y ubicaciones.
View	F4		Abre un archivo existente sólo para verlo. No se pueden efectuar cambios en el archivo.
Save	F5		Guarda la ventana. Usted puede guardar los datos introducidos (un modelo), una Reports window o una ventana de comandos. La información se puede guardar en los formatos siguientes: *.LTX, un formato de texto que puede ser editado con *software* de procesador de palabras; *.LPK, para guardar modelos compilados en un formato "empaquetado", pero sin formato especial ni comentarios; y *.MPS, el formato estándar de la industria de máquinas independientes para transferir problemas de PL en LINDO a otro *software* para PL.
Save As...	F6		Guarda la ventana activa con un nombre de archivo específico. Es útil para dar un nuevo nombre a un archivo revisado y conservar intacto el archivo original.
Close	F7		Cierra la ventana activa. Si la ventana contiene información nueva, se le preguntará a usted si desea guardar los cambios.
Print	F8		Envía la ventana activa a la impresora.
Printer Setup...	F9		Selecciona la impresora y varias opciones para el formato de impresión.
Log Output...	F10		Envía toda la actividad posterior de la pantalla, que normalmente se enviaría a la *Report window*, a un archivo de texto. Cuando usted especificó una ubicación del archivo log, una marca aparecerá en el menú *File* en la línea *Log Output*. Para inhabilitar *Log Output* seleccione simplemente de nuevo el comando.

COMANDO	DESCRIPCIÓN
Take Commands... F11	"Toma" un *batch file* de LINGO con comandos y texto para operación automatizada. Se puede poner un modelo en la memoria, resolverlo, y la solución es colocada en la ventana *Reports* y guardada en un archivo. Si usted usa el comando *Batch* antes del inicio del texto del modelo, el modelo y los comandos contenidos en el archivo, así como la solución se verán en la ventana *Reports*.
Basis Read F12	Recupera una solución de un modelo que se guardó usando el comando *Basis Save*.
Basis Save Shift+F2	Guarda la solución para el modelo activo en disco con un nombre de archivo específico.
Title Shift+F3	Muestra el título del modelo activo, si se ha incluido uno con la instrucción opcional *Title* en el modelo.
Date Shift+F4	Abre una ventana *Reports* y muestra fecha y hora actuales según el reloj de su computadora.
Elapsed Time Shift+F5	Abre una ventana *Reports* y muestra el tiempo total transcurrido en su sesión actual de LINDO.
Exit Shift+F6	Cierra LINDO.

Menú Edit

Los comandos del menú *Edit* le permiten efectuar tareas básicas de edición comunes a la mayoría de aplicaciones de Windows, así como ejecutar varias tareas exclusivas de LINDO. Sigue una descripción de los comandos de *Edit*.

COMANDO	DESCRIPCIÓN
Undo Ctrl+Z	Deshace la última acción.
Cut Ctrl+X	Corta el texto seleccionado y lo coloca en el portapapeles para pegarlo.
Copy Ctrl+C	Copia el texto seleccionado en el portapapeles para pegarlo.
Paste Ctrl+V	Inserta o pega el contenido del portapapeles en el punto de inserción.
Clear Delete	Borra el texto seleccionado, pero no lo coloca en el portapapeles.
Find/Replace... Ctrl+F	Busca la ventana activa para encontrar el texto seleccionado y reemplazarlo con el texto introducido en el cuadro "Replace with".
Options Alt+O	Permite ver y cambiar varios parámetros usados en las sesiones de LINDO.
Go To Line... Ctrl+T	Le permite mover el cursor a cualquier línea en la ventana activa.
Paste Symbol... Ctrl+P	Le permite pegar nombres de variables y símbolos reservados en la ventana activa.
Select All Ctrl+A	Selecciona todo lo que haya en la ventana activa para cortarlo o copiarlo.
Clear All	Borra el contenido total de la ventana activa.
Choose New Font	Selecciona una nueva fuente para el texto en la ventana activa.

Menú Solve

Los comandos del menú *Solve* se usan después de que usted introdujo datos y están listos para obtener una solución. Enseguida se presenta la descripción de los comandos.

COMANDO	DESCRIPCIÓN
Solve Ctrl+S	Envía el modelo de la ventana activa al Solver de LINDO para obtener la solución.
Compile Model Ctrl+E	Traduce el modelo al formato aritmético requerido por el Solver de LINDO. Los modelos también se compilan en forma automática cuando usted usa el comando *Solve*.
Debug Ctrl+D	Ayuda a determinar problemas con modelos no factibles y no acotados. Se pueden identificar conjunto (renglones) suficientes y necesarios, como las restricciones cruciales, aquellas que hacen factible un modelo no factible si se eliminaran del modelo.
Pivot Ctrl+N	Hace que LINDO ejecute el paso siguiente en el proceso de la solución, con lo que es posible que los problemas de programación lineal se resuelvan paso a paso.
Preemptive Goal Ctrl+G	Ejecuta la optimización Lexico (una forma de programación por objetivos) en un modelo.

Menú Reports

Los comandos del menú *Reports* permiten a usted especificar cómo generar los informes con LINDO. Las descripciones de los comandos de *Reports* se presentan enseguida.

COMANDO	DESCRIPCIÓN
Solution Alt+0	Abre el cuadro de diálogo de *Solutions Report Options*, el cual le facilita especificar cómo quiere usted que aparezca su informe.
Range Alt+1	Crea un informe del intervalo, o análisis de sensibilidad, para la ventana activa con el modelo.
Parametrics Alt+2	Ejecuta un análisis paramétrico sobre el segundo miembro o lado derecho de una restricción.
Statistics Alt+3	Despliega las variables estadísticas para el modelo de la ventana activa.
Peruse Alt+4	Se usa para ver informes o partes seleccionadas de la estructura o solución del modelo actual.
Picture Alt+5	Representa el modelo actual en la forma de matriz. Los coeficientes no cero de la matriz se podrían representar como texto o como gráficos.
Basis Picture Alt+6	Muestra un informe con formato de texto con un "panorama" de la base actual, y ordena los renglones y las columnas según la última inversión o triangulación ejecutada por el Solver. El informe *Basis Picture* se envía a la ventana *Reports*.
Tableau Alt+7	Muestra el tableau simplex para el modelo activo. Facilita la observación del algoritmo simplex en cada paso.
Formulation Alt+8	Muestra su modelo completo o en partes específicas en la ventana *Reports*.
Show ColumnAlt+9	Muestra una columna seleccionada sin el resto del modelo.
Positive Definite	Busca una garantía de optimalidad global en un modelo cuadrático.

Menú Window

Los comandos del menú *Window* permiten que usted ajuste ventanas activas de comandos y de estado, y organice el despliegue de varias ventanas. A continuación se describen los comandos de *Window*.

COMANDO	DESCRIPCIÓN
Open Command Window Alt+C	Proporciona acceso a la interfaz de la línea de comandos de LINDO, donde usted podría introducir comandos en el indicador de dos puntos.
Open Status Window	Abre la ventana *Solver Status* de LINDO, la cual despliega información acerca del estado del optimizador, tal como un número de iteraciones o tiempo transcurrido. Esta ventana aparece también cuando usted selecciona *Solve* en el menú *Solve*.
Send to Back Ctrl+B	Envía la ventana más al frente hasta atrás.
Cascade Alt+A	Acomoda todas las ventanas abiertas en forma de cascada desde la parte superior izquierda hasta la parte inferior derecha; la ventana activa está en la parte superior.
Tile Alt+T	Acomoda todas las ventanas abiertas de tal manera que ocupen el espacio equivalente dentro de la ventana del programa.
Close All Alt+X	Cierra todas las ventanas activas.
Arrange Icons Alt+I	Desplaza los iconos que representan ventanas minimizadas, de tal manera que se acomodan en la parte inferior de la pantalla.
List of Windows	En la parte inferior del menú Window se despliega una lista de las ventanas abiertas. Se marca la ventana activa.

Menú Help

Los comandos del menú *Help* proporcionan acceso a la ayuda en línea de LINDO. Se describen en seguida los comandos de *Help*.

COMANDO	DESCRIPCIÓN
Contents F1	Muestra el contenido de la sección de ayuda. El segundo icono (el de la flecha y el signo de interrogación) posibilita la ayuda sensible al contexto, donde el indicador del cursor cambia a un signo de interrogación y la ayuda se le proporciona específicamente según el comando seleccionado.
Search for Help on... Alt+F1	Busca la sección de ayuda para una palabra o un tema.
How to Use Help Ctrl+F1	Proporciona ayuda para aprender a usar el sistema de ayuda en línea.
About LINDO...	Muestra la pantalla inicial de arranque con información general acerca de LINDO.

Instrucciones de modelado opcionales

Aparte de los elementos básicos de un modelo, LINDO reconoce varias instrucciones opcionales que podrían aparecer después de la instrucción END. Estas instrucciones significan capacidades mayores de modelado, tal como asignar límites adicionales a las variables. La descripción de estas instrucciones se proporciona enseguida.

INSTRUCCIÓN	DESCRIPCIÓN
FREE <Variable>	Elimina todos los límites de la variable, por lo cual ésta asume cualquier valor real: positivo o negativo.
GIN <Variable>	Una variable queda restringida a ser un entero general (es decir, en el conjunto de los enteros no negativos).
INT <Variable>	Una variable queda restringida a ser un entero binario (es decir, 0 o 1).
SLB <Variable> <Value>	Establece un límite inferior simple para una variable (es decir, SLB X 10 requeriría que *X* fuera mayor que 10 o igual a 10).

SUB <Variable> <Value> Establece un límite superior simple para una variable (es decir, SUB X 10 requeriría que *X* fuera menor que 10 o igual a 10).

QCP <Constraint> Señala la primera restricción "real" en un modelo de programación cuadrática.

TITLE <Title> Permite que usted asigne un título a su modelo. El título se despliega mediante el comando *Title* del menú *File*.

APÉNDICE B Inicio con LINGO

Bienvenido a la parte de LINGO de este libro. El apéndice proporciona información básica breve sobre LINGO y le ayuda a instalar el *software*. Las características del software y cómo aplicarlo a los problemas muestra se explican en los capítulos posteriores.

¿Qué es LINGO?

LINGO es un paquete de *software* interactivo que se usa para resolver problemas de programación lineal, por enteros y no lineal. Es posible aplicarlo en situaciones similares en las que se utiliza LINDO, pero ofrece mayor flexibilidad en términos de cómo se expresan los modelos. Al contrario de LINDO, LINGO permite usar paréntesis y variables en el segundo miembro de una ecuación. Por lo tanto, las restricciones se escriben en la forma original y no se tiene que volver a escribirlas con constantes en el lado derecho. LINGO es capaz también de generar modelos grandes con relativamente pocas líneas de datos. Asimismo, el programa proporciona una gran biblioteca de funciones matemáticas, estadísticas y probabilísticas, y tiene mayor aptitud para leer información de archivos externos y hojas de cálculo.

Fundamentos de LINGO

De una manera muy parecida a LINDO, LINGO se puede usar para resolver problemas en forma interactiva con la ayuda del teclado, o bien, resolver problemas usando archivos creados en otro lado, en un programa independiente o como parte de un programa integrado que contiene códigos personalizados y bibliotecas de optimización de LINGO. Este apéndice se centra principalmente en el primer método, es decir, en resolver problemas de manera interactiva. Mayor información acerca de los otros métodos se encuentra en LINDO Systems, Inc.

Introducir un modelo en la versión de LINGO para Windows es similar a escribir en un formato del procesador de palabras de Windows: usted escribe simplemente los datos del modelo como los escribiría si resolviera en forma manual el problema. La ventana interior que se llama al inicio "*untitled*" (sin título) está lista para aceptar datos LINGO también posee comandos u órdenes básicas de edición para cortar, copiar y pegar texto. Esta herramienta, y otras características se encuentran en los comandos de *Window* tratados en el apéndice C.

Los elementos requeridos de LINGO son similares a los de LINDO. LINGO requiere también un objetivo, una o más variables y una o más restricciones. Pero al contrario que LINDO, las restricciones de LINGO *no* están precedidas por ningún término especial tal como SUBJECT TO (sujeto a) o SUCH THAT (tal como).

La sintaxis de LINGO es similar a la de LINDO, con las diferencias siguientes:

- Las instrucciones de LINGO terminan con ; (punto y coma).
- LINGO incluye operadores matemáticos adicionales, como se explica en el apéndice C. Se requiere un asterisco para denotar multiplicaciones.
- A veces se incluyen paréntesis para definir el orden de las operaciones matemáticas si así lo desea usted.
- Los nombres de las variables pueden tener hasta 32 caracteres.

APÉNDICE C Comandos de los menús y funciones de LINGO

Comandos de los menús

Se puede tener acceso a los comandos u órdenes de LINGO desde un menú conveniente similar a los de los otros programas de Windows. El menú principal comprende cinco submenús acomodados en la parte superior de la pantalla, en los que se listan los diversos comandos. Cuando usted da un clic en uno de los submenús (File, Edit, LINGO, Window o Help) aparece un menú con los distintos comandos. Usted puede seleccionar los comandos de la misma manera que lo hace en la mayoría de los programas de Windows: haciendo clic en el comando con el ratón o escribiendo la letra que está subrayada en el nombre del comando cuando el submenú apropiado está remarcado. Muchos comandos también tienen asignada una combinación de teclas que funcionan como orden corta (F2, Ctrl+Z, etc.). Otra ventaja más es que también se puede tener acceso a algunos de los comandos más usados mediante un icono localizado en una barra de herramientas en la parte superior de la pantalla.

Menú File

Los comandos del menú File (archivo) le permiten manejar sus archivos de datos en LINGO de distintas maneras. Usted puede usar este menú para abrir, cerrar, guardar e imprimir archivos, así como para ejecutar varias tareas únicas con LINGO. Enseguida se describen los comandos de File.

COMANDO			DESCRIPCIÓN
New	F2		Crea una nueva ventana para ingresar datos.
Open	F3		Abre un archivo existente. Mediante los cuadros de diálogo usted puede seleccionar de entre varios tipos de archivos y ubicaciones.
Save	F4		Guarda la ventana activa. Usted puede guardar los datos de entrada (un modelo), una *reports window* (ventana de informes) o una ventana de comandos.
Save As...	F5		Guarda la ventana activa con un nombre de archivo específico. Es útil para volver a asignar un nombre a un archivo revisado y conservar intacto el archivo original.
Close	F6		Cierra la ventana activa. Si la ventana contiene nuevos datos, entonces se le preguntará a usted si desea guardar los cambios.
Print	F7		Envía la ventana activa a su impresora.
Printer Setup...	F8		Selecciona la impresora y varias opciones para el formato de impresión.
Log Output...	F9		Envía toda la actividad posterior de la pantalla, que normalmente se enviaría a la ventana *Reports*, a un archivo de texto. Si usted especifica una ubicación *log file*, aparece una marca en el menú *File* en la línea de *Log Output*. Para inactivar Log Output, seleccione simplemente el comando de nuevo.
Take Commands	F11		"Toma" un *batch file* de LINGO con comandos y texto del modelo para operación automatizada. Se puede poner un modelo en la memoria, resolverlo, y la solución es colocada en la ventana *Reports* y guardada en un archivo. Si usted usa el comando *BATCH* antes del inicio del texto del modelo, el modelo y los comandos contenidos en el archivo, así como la solución, se verán en la ventana *Reports*.
Import LINDO file	F12		Abre un archivo que contiene un modelo de LINDO en el formato TAKE de LINDO, y traduce el modelo en un formato que LINGO acepta.
Exit	F10		Cierra LINGO.

Menú Edit

Los comandos del menú Edit permiten a usted ejecutar tareas básicas de edición, comunes a la mayoría de aplicaciones para Windows, así como efectuar varias tareas que son exclusivas para LINGO. A continuación se describen los comandos Edit.

COMANDO		DESCRIPCIÓN
Undo	Ctrl+Z	Deshace la última acción.
Cut	Ctrl+X	Corta el texto seleccionado y lo coloca en el portapapeles para pegarlo.
Copy	Ctrl+C	Copia el texto seleccionado en el portapapeles para pegarlo.
Paste	Ctrl+V	Inserta o pega el contenido del portapapeles en el punto de inserción.
Clear	Delete	Borra el texto seleccionado, pero no lo coloca en el portapapeles.
Find/Replace...	Ctrl+F	Busca la ventana activa para encontrar el texto seleccionado y reemplazarlo con el texto introducido en el cuadro "*Replace with*".
Go To Line...	Ctrl+T	Usted puede mover el cursor a cualquier línea especificada de la ventana activa.
Match Parenthesis	Ctrl+P	Encuentra el paréntesis cerrado que corresponde al paréntesis abierto seleccionado.
Paste Function		Pega las funciones incorporadas en LINGO en el punto de inserción actual. Después de seleccionar este comando aparece otro submenú con las distintas categorías de las funciones
Select All	Ctrl+A	Selecciona toda la ventana activa para cortarla o copiarla.
Choose New Font		Selecciona una nueva fuente para el texto en la ventana activa.

Menú LINGO

Los comandos del menú LINGO se usan después de que usted ya introdujo datos y están listos para obtener una solución. Siguen las descripciones de los comandos de LINGO:

COMANDO		DESCRIPCIÓN
Solve	Ctrl+S	Envía el modelo que se encuentra en la ventana activa al Solver de LINGO.
Solution...	Ctrl+O	Abre el cuadro de diálogo *Solution Report Options* (Opciones para mostrar la solución), el cual permite a usted especificar cómo quiere que aparezca su solución.
Range	Ctrl+R	Despliega un informe de intervalos, el cual le muestra dentro de qué valores usted puede cambiar coeficientes sin modificar los valores óptimos.
Look...	Ctrl+L	Despliega todo el modelo o las líneas seleccionadas.
Generate...	Ctrl+S	Crea otra versión del modelo actual en formato algebraico, de LINDO o MPS. Se puede usar para numerar renglones y desplegar el modelo en un formato más fácil de leer. El comando GEN proporciona una capacidad similar desde la ventana de comandos.
Export to Spreadsheet	Ctrl+E	Exporta valores de variables seleccionadas a intervalos nombrados en una hoja de cálculo. Primero se tiene que crear una hoja de cálculo con intervalos dimensionados para que se puedan acomodar en ellos los valores exportados. Los intervalos *tienen* que contener números. Al seleccionar este comando se abrirá un cuadro de diálogo que pide la plantilla y las

	hojas de trabajo (nombres del archivo de la hoja de cálculo), variables por exportar y el intervalo para el cual se exportarán los valores. Las variables y el intervalo se introducen por pares y se añaden a la lista de pares de variable e intervalo dando un clic en el botón de agregar.
Options Alt+O 	Le permite ver y modificar varios parámetros usados en las sesiones de LINGO.
Workspace Limit Ctrl+S	Asigna memoria a LINGO. Si usted elige "None", (Ninguna), LINGO usará toda la memoria disponible.

Menú Window

Los comandos del menú *Window* le permiten ajustar cualquier comando abierto y ventanas de estado, así como organizar el despliegue de varias ventanas. Enseguida se proporcionan las descripciones de los comandos de *Window*.

COMANDO	DESCRIPCIÓN
Open Command Window	Permite el acceso a la interfaz de la línea de comandos de LINGO, donde usted puede introducir comandos en el indicador : (dos puntos).
Open Status Window	Abre la ventana de estado del Solver para LINGO, la cual muestra información respecto al estado del optimizador, tal como número de iteraciones y tiempo transcurrido. Esta ventana aparece también cuando selecciona *Solve* en el menú de LINGO:
Send to Back  Alt+B	Envía la ventana del frente hasta atrás.
Close All Alt+X 	Cierra todas las ventanas activas.
Cascade Alt+A	Acomoda todas las ventanas en forma de cascada desde la parte superior izquierda a la inferior derecha; la ventana activa queda en la parte de encima.
Tile Alt+T	Acomoda todas las ventanas abiertas de tal manera que ocupen el espacio equivalente dentro de la ventana del programa.
Arrange Icons Alt+I	Desplaza los iconos que representan ventanas minimizadas de tal manera que se acomodan en la parte inferior de la pantalla.
List of Windows	En la parte inferior del menú Window se despliega una lista de las ventanas abiertas. Resalta la ventana activa.

Menú Help

Los comandos del menú *Help* permiten tener acceso a la ayuda en línea de LINGO. Se describen a continuación los comandos de *Help*.

COMANDO	DESCRIPCIÓN
Contents   F1	Muestra el contenido de la sección de ayuda. El segundo icono (el de la flecha y el signo de interrogación) posibilita la ayuda sensible al contexto; el indicador del cursor cambia a un signo de interrogación y la ayuda se le proporciona específicamente según el comando seleccionado.
Search for Help on... Alt+F1	Busca la sección de ayuda para una palabra o un tema.
How to Use Help Ctrl+F1	Proporciona ayuda para aprender a usar el sistema de ayuda en línea.
About LINGO...	Muestra la pantalla inicial de arranque con información general acerca de LINGO.

Funciones

LINGO tiene siete funciones principales: operadores estándar, importación de archivos, finanzas, matemáticas, iteraciones en conjuntos, dominio de variable y probabilidad, y una variedad de otras funciones. La mayoría de estas funciones está disponible por medio de los comandos del menú. El *software* de LINGO incluye una descripción detallada de sus funciones en pantallas de ayuda en línea; por lo tanto, sólo se proporciona aquí una descripción breve de las funciones de LINGO.

Operadores estándar

Entre estos operadores se encuentran los aritméticos (es decir., ^, *, /, +, y −), operadores lógicos (#EQ#, #NE#, #GT#, #GE#, #LT#, y #L3#) para determinar la calidad del conjunto y operadores de igualdad-desigualdad (<, =, >, <=, y >=) para especificar si el primer miembro de una expresión debe ser menor que, igual a, o mayor que el segundo miembro. Estos operadores constituyen algunas de las funciones más básicas disponibles en LINGO. Obsérvese que los símbolos "mayor que" y el "menor que" (> y <) se interpretan como desigualdades "sueltas" [es decir, mayor que o igual a (≥) y menor que o igual a (≤), respectivamente]. Usted escriba simplemente estos operadores en el teclado, en lugar de "jalarlos" desde un comando de *Window*.

Funciones de importación de archivos

Estas funciones le permiten importar texto y datos de fuentes externas. La función @FILE le permite importar texto o datos desde un archivo en ASCII, y la función @IMPORT le permite importar datos sólo de una hoja de cálculo.

Funciones de finanzas

Entre estas funciones están @FPA(I,N), la cual da el valor presente de una anualidad, y la función @FPL(I,N), la cual regresa el valor presente de un valor global de *N* periodos de $1 a partir de ahora si la tasa de interés es *I* por periodo. *I* no es un porcentaje, sino un número no negativo que representa la tasa de interés.

Funciones matemáticas

Comprenden las funciones generales y trigonométricas siguientes: @ABS(X), @COS(X), @EXP(X), @LGM(X), @LOG(X), @SIGN(X), @SIN (X), @SMAX(list), @SMIN(list), @TAN(X). Se pueden utilizar combinaciones de las tres funciones trigonométricas básicas (seno, coseno y tangente) para obtener otras funciones trigonométricas.

Funciones de iteraciones en conjuntos

Comprenden @FOR (set_name : constraint_expressions), @MAX (set_name : expression), @MIN (set_name : expression) y @SUM (set_name : expression). Estas funciones operan sobre un conjunto completo, y producen un solo resultado en todos los casos, excepto con la función @FOR, la cual genera restricciones independientemente de cada elemento del conjunto.

Funciones del dominio de la variable

Estas funciones fijan restricciones adicionales sobre variables y atributos. Comprenden las siguientes: @BND(L, X, U), @BIN(X), @FREE(X) y @GIN(X).

Funciones de probabilidad

LINGO posee capacidades estadísticas comunes con sus funciones de probabilidad: @PSN(X), @PSL(X), @PPS(A,X), @PPL(A,X), @PBN(P,N,X), @PHG(POP,G,N,X), @PEL(A,X), @PEB(A,X), @PFS(A,X,C), @PFD(N,D,X), @PCX(N,X), @PTD(N,X) y @RAND(X).

Otras funciones

Otras funciones que posee LINGO son @IN(set_name, set_element), @SIZE(set_name), @WARN('text',condition), @WRAP(I,N) y @USER. Estas funciones ofrecen una variedad de capacidades además de las de las categorías ya mencionadas.

BIBLIOGRAFÍA

Hay muchos textos sobre programación lineal, sin olvidar los siguientes:

Bazaraa, M. y J. Jarvis. *Linear Programming and Network Flows*. Nueva York: Wiley, 1990.

Bersitmas, D. y J. Tsitsiklis. *Introduction to Linear Optimization*. Belmont, Mass.: Athena Publishing, 1997.

Bradley, S., A. Hax y T. Magnanti. *Applied Mathematical Programming*. Reading, Mass.: Addison-Wesley, 1977.

Chvàtal, V. *Linear Programming*. San Francisco: Freeman, 1983.

Dantzig, G. *Linear Programming and Extensions*. Princeton, N.J.: Princeton University Press, 1963.

Gass, S. *Linear Programming: Methods and Applications*, 5a´. ed. Nueva York: McGraw-Hill, 1985.

Luenberger, D. Linear and Nonlinear Programming, 2a ed. Reading, Mass.: Addison-Wesley, 1984.

Murty, K. *Linear Programming*. Nueva York: Wiley, 1983.

Nash, S. y A. Sofer. *Linear and Nonlinear Programming*. Nueva York: McGraw-Hill, 1995.

Nering, E. y A. Tucker. *Linear Programs and Related Problems*. Nueva York: Academic Press, 1993.

Simmons, D. *Linear Programming for Operations Research*. Englewood Cliffs, N.J.: Prentice Hall, 1972.

Simonnard, M. *Linear Programming*. Englewood Cliffs, N.J.: Prentice Hall, 1966.

Wu, N. y R. Coppins. *Linear Programming and Extensions*. Nueva York: McGraw-Hill, 1981.

Bland, R. "New Finite Pivoting Rules for the Simplex Method", *Mathematics of Operations Réséarch*, 2(1977):103-107. Explica un enfoque sencillo e ingenioso para evitar los ciclos.

Dantzig, G. y N. Thapa. *Linear Programming*. Nueva York: Springer-Verlag, 1997.

Karmarkar, N. "A New Polynomial Time Algorithm for Linear Programming", *Combinatorica* 4(1984):373-395. El método de Karmarkar para resolver PL.

Klee, V. y G. Minty. "How Good Is the Simplex Algorithm?" En *Inequalities-III*. Nueva York: Academic Press, 1972. Describe los PL para los cuales el método simplex examina cada solución factible básica antes de encontrar la solución óptima.

Kotiah, T. y N. Slater. "On Two-Server Poisson Queues with Two Types of Customers", *Operations Research* 21(1973):597-603. Describe una aplicación real que genera un PL en el cual surgen los ciclos.

Love, R. y L. Yerex. "An Application of a Facilities Location Model in the Prestressed Concrete Industry", *Interfaces* 6(no.4, 1976):45-49.

Papadimitriou, C. y K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Englewood Cliffs, N.J.: Prentice Hall, 1982. Más análisis de algoritmos de tiempo polinómico y tiempo exponencial.

Schrage, L. *User's Manual for LINDO*. Palo Alto, Calif.: Scientific Press, 1990. Proporciona detalles acerca de LINDO.

Schrage, L. *User's Manual for LINGO*. Chicago, Ill.: LINDO Systems Inc., 1991. Ofrece detalles sobre LINGO.

Schrage, L. *User's Manual for What's Best*: Chicago, Ill.: LINDO Systems Inc., 1993. Proporciona detalles acerca de qué es mejor.

Wagner, H. "Linear Programming Techniques for Regression Analysis", *Journal of the American Statistical Association* 54(1954):206-212.