63
2 ej.

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE ECONOMIA

PAQUETE DE PROGRAMACION LINEAL "PROLIN"
(CON APLICACIONES A LA ECONOMIA)

T E S I S

QUE PARA OBTENER EL TITULO DE:

LICENCIADO EN ECONOMIA

Presentan:

JUAN MANUEL MUÑOZ ARAUJO
ALEJANDRO REYES GUERRERO

Cd. Universitaria 1992

I N D I C E

Pag.

I N T R O D U C C I O N

El siguiente documento está orientado a describir una herramienta de fácil manejo, para resolver problemas, específicamente para el ámbito económico, utilizando para ello la técnica de Programación Lineal; de tal forma que quienes realicen Programación Lineal centren su atención en la elaboración de la Función Objetivo y las restricciones que componen el Modelo de una manera completa sin que estén preocupados por la complejidad de las operaciones o por el gran número de restricciones que puedan surgir.

Así, este apartado tiene como finalidad mostrar el desarrollo del mismo documento. El estudiante tendrá a su alcance una herramienta más para poder comprender como encaja la Programación Lineal dentro del campo de la economía, así como una idea general de las aplicaciones de la técnica en la actualidad, los alcances y limitaciones de la misma. Dicha herramienta permitirá el apoyo a la docencia, para que los cursos de Programación Lineal e Investigación de Operaciones sean mas prácticos.

En el Capítulo I, se da la ubicación de la Programación Lineal, así como sus antecedentes históricos.

Para el Capítulo II, se mostrará en forma muy general la descripción de como se puede resolver un problema de asignación de recursos, a partir de un problema dado, utilizando para ello la técnica de Programación Lineal, bajo sus diferentes métodos de solución, tales como: Método Gráfico, Simplex y Dual-Simplex.

Este capítulo también describe cual fué el procedimiento para desarrollar el diseño del Paquete de Programación Lineal Titulado "PROLIN", que tiene como objetivo desarrollar problemas de Programación Lineal de una manera muy sencilla y fácil de interpretar los resultados.

Se incluye un pequeño glosario de términos que le permitirán al usuario comprender la interpretación de la salida de resultados en la solución de un problema de Programación Lineal, ya sea utilizando el Paquete PROLIN u otros; así como las diferentes aplicaciones que se pueden dar utilizando la Programación Lineal en los distintos sectores de la economía.

El Capítulo III tratará de abocarse a la descripción del manejo del Paquete, de tal forma que resulte de fácil comprensión para el usuario. Además, se hace mención de las limitantes con las que cuenta el paquete "PROLIN", así como sus ventajas.

"PROLIN" es una herramienta de autoaprendizaje; restringida tan sólo a 10 variables y 10 restricciones; se maneja bajo el procedimiento del Tableau de Tucker, basado sobre el Método Dual-Simplex, donde las variables deben cumplir con la propiedad de no-negativas y los resultados serán interpretados por el usuario de acuerdo a su objetivo.

También se añadira el diskette del Paquete "PROLIN", como así el programa Fuente del mismo.

Una vez mencionado el manejo del paquete "PROLIN" en el Capítulo anterior, se hace referencia que en la actualidad existen varios paquetes que permiten resolver problemas de Programación Lineal; es por ello que en el Capítulo IV, de manera general, se hace una descripción de dos paquetes (LINDO y QSB), que le permiten al usuario obtener la solución a sus modelos planteados.

Cabe mencionar que esta descripción de los paquetes anteriores es tan sólo de manera esquemática, dado que si un usuario desea conocer más a fondo el manejo del paquete, se le recomienda utilizar los manuales correspondientes.

En el último capítulo se dan una serie de ejemplos de tipo económico, los cuales podrá utilizar el estudiante para reforzar sus conocimientos de Programación Lineal, estos ejemplos además le permitirán al usuario saber dar una interpretación económica de los resultados obtenidos en la utilización de los diferentes paquetes de cómputo. También se mostrarán las gráficas de algunos problemas y las salidas de resultados de los distintos paquetes utilizados en el documento.

I. ANTECEDENTES DE LA PROGRAMACION LINEAL

Muchos problemas en los negocios conciernen esencialmente con la asignación de recursos limitados como son: dinero, personal, materiales, máquinas, espacios y tiempo; para maximizar de alguna manera los rendimientos o minimizar los costos. Dicho problema económico se puede resolver a través de una técnica matemática llamada Programación Lineal, la cual tiene sus orígenes en el modelo de Insumo Producto desarrollado por W. W. Leontief; sin embargo, su versión actual se considera de un origen más reciente. El estado de su conocimiento se atribuye a G.D. Dantzing, quien introdujo el método Simplex como un procedimiento sistemático para la solución de problemas idóneos con el uso de la Programación Lineal.

No todo tipo de problema sobre la asignación óptima de recursos puede ser resuelto a través de la programación lineal, sin embargo algunos de ellos por sus características, permiten aplicar dicha técnica tal es el caso de: (producción, transporte, etc.), ya que pueden ser formuladas a través de ésta y cumplen además con los supuestos necesarios en los que se basa dicha herramienta. La primera técnica matemática, ampliamente aceptada en el campo, conocida como el Método Simplex de Programación Lineal, fue desarrollada en 1947 por el matemático norteamericano George B. Dantzing. Desde entonces las nuevas técnicas de aplicación y de Investigación de Operaciones han tenido un gran progreso. Dicho progreso se debe en gran parte al desarrollo paralelo de la computadora. En efecto, si no hubiera sido por la computadora, los problemas de Programación Lineal y la Investigación de Operaciones no habrían adquirido el estado actual promisorio en todas las clases de ámbito[1].

En el campo de la Economía es un problema cotidiano el que se refiere a la distribución de recursos limitados, donde la distribución de recursos escasos es un problema de decisiones, pues implica la acción de preferir una alternativa de uso entre una gama infinita de posibilidades. Tales decisiones están basadas en consideraciones técnicas y prácticas que proporcionen elementos suficientes para poder tomar la mejor decisión. Dicha decisión posee una solución óptima, ya que permite aplicarle métodos matemáticos (aún sin número de ecuaciones) al problema de tipo económico cuantitativo. Esto permite hacer más científica a la economía, donde a dicha técnica matemática se le conoce con el nombre de Investigación de Operaciones o Teoría de las Decisiones, cuyo objetivo principal es la determinación de soluciones óptimas de los problemas económicos mediante métodos matemáticos. A tal grupo de técnicas pertenece la Programación Lineal; el problema que resuelve, en su aspecto general, es el que se refiere a determinar la combinación de recursos, o sea la necesidad de aprovechar al máximo el uso de los recursos que cada vez son más limitados, con el propósito de maximizar la

[1] PAG. No. 2 CAOLT I. INVESTIGACION DE UPER TAHA HAMAY-TAHA.

utilidad ya sea medida en términos monetarios o en bienestar.

Una economía tiene a su disposición en un momento cualquiera, una cantidad dada de varios factores productivos y cierto número de tareas a las que aquellas se pueden destinar. Estos factores productivos pueden ser asignados a las diferentes tareas, por lo general de muchas maneras distintas, con resultados diversos, como se señala este tipo de problemas es de los más frecuentes en el análisis económico, es decir el análisis de las particularidades que presenta la mejor asignación de factores a casos como éste. Dicha combinación óptima puede determinarse por medio de la Programación Lineal, cuyo objetivo es la maximización de las utilidades, ahora bien tales utilidades dependen del número que de cada artículo se produzca. El problema, desde el punto de vista matemático, consiste en obtener el valor máximo de una función considerada por desigualdades, empero, lo que se ha dicho para el caso de las utilidades o el producto, es válido también para conceptos tales como costos; sin embargo, en ese caso el objetivo será minimizarlos.

Como podemos darnos cuenta, la Programación Lineal se puede aplicar aún sin número de sectores económicos donde exista el problema de la asignación de recursos limitados y donde se desea maximizar los recursos y minimizar los costos.

Existen varios métodos para el cálculo de programas lineales, gran parte de los cuales requiere de conocimientos matemáticos. Aquí sólo se limitará la exposición del método Simplex y su Dual. Dicha solución es una tarea sencilla a partir de la existencia de programas de cómputo.

Cabe señalar que no todo tipo de problemas sobre asignación óptima de recursos puede ser resuelto a través de la Programación Lineal, ya que el uso de la Programación Lineal requiere de una formulación especial del problema y la consideración de un conjunto de supuestos. La aplicación de dicha técnica matemática a problemas económicos productivos de la empresa, se basa en su desarrollo, en la maximización del beneficio y en la minimización de los costos a través de la combinación óptima seleccionada, además dicha técnica de programación lineal deberá tomar en cuenta:

1.- El problema de la selección entre los diversos procesos cuando los factores son limitados, disponibles en cualquier cantidad deseada a un precio constante.

2.- El problema de la elección entre diversos procesos cuando ciertos factores de producción son limitados y se dispone de ellos hasta cierto tipo de cantidad y a un mismo precio.

Problemas cuyas características sean los puntos anteriores, se pueden resolver a través de sistemas de desigualdades y ecuaciones lineales, cuya finalidad es establecer en el nivel óptimo del proceso.

Los problemas que tratan con ecuaciones y desigualdades, son resueltos por medio de la Programación Lineal, que es una herramienta matemática de análisis de fenómenos económicos en los que interviene un cierto número de variables "no negativas" ligadas entre sí por relaciones independientes de carácter lineal. Estas relaciones forman un sistema de ecuaciones lineales, que constituyen las restricciones y reflejan los objetivos del fenómeno estudiado, se asocia a estas restricciones una función económica también de primer grado, que se llama Función Objetivo por lo que la Programación Lineal es un método matemático que permite optimizar (maximizar o minimizar) la función económica o función objetivo, respetando las restricciones.

II. MARCO TEORICO

2.1. Significado de la Programación Lineal.

Dentro del campo de la "Investigación de Operaciones", la Programación Lineal ha figurado como una de las técnicas de mayor uso y aplicación práctica, en la optimización de un programa a desarrollar en base a ciertas medidas restrictivas, que generalmente se presentan como recursos escasos o difíciles de lograr.

Desde un punto de vista puramente matemático, la "Programación Lineal", es una metodología que se utiliza en la solución de problemas en los que se desea maximizar o minimizar una función lineal de una o más variables, llamada "Función Objetivo", sujeta a ciertas limitaciones (restricciones) que se pueden representar como desigualdades o igualdades de funciones lineales de las variables.

En lo que toca al presente trabajo a desarrollar, interesa la aplicación de la programación lineal en el aspecto económico, por lo tanto, la citada metodología se utilizará como una herramienta para resolver problemas de esta naturaleza.

Es por ello, que hablando en términos económicos, "La Programación Lineal es una técnica empleada para distribuir un grupo de recursos limitados, que se asignan a un número de actividades", es decir, el papel fundamental que juega esta herramienta matemática dentro de la economía, es el de economizar al máximo los recursos escasos disponibles, determinando el programa óptimo más conveniente para su aplicación en algún sector de la economía.

Entre las herramientas más útiles para estudiar los sistemas que se presentan en la economia, se encuentran los métodos de optimización. Dentro de éstos, está la **programación matemática**, que pretende encontrar el valor óptimo del objetivo del sistema, sujetándose a una serie de restricciones que surgen de las relaciones que existen entre sus entidades.

Una de sus técnicas más importantes y más utilizadas es la **programación lineal** , que recibe este nombre porque todas sus relaciones funcionales se pueden expresar como ecuaciones lineales.

2.2. Aplicaciones Actuales de la Programación Lineal.

Actualmente la Programación Lineal, al parecer, tiene mayor importancia en la industria que en cualquier otra rama de la economía. Por tanto, es de mayor aplicación en los países industrializados que en aquellos en vías de desarrollo.[2]

2.2.1. Sector Agropecuario.

En lo que toca a este sector de la economía del país, la Programación Lineal se aplica en dos campos definidos: en la economía y en la administración agrícolas. En la primera, trata de resolver problemas agrícolas macroeconómicos, en

[2] VICTOR RIOS G. ."INVESTIGACION DE OPERACIONES", PAG 24-27

tanto que en la segunda se interesa en los problemas de las granjas individuales. Algunas de sus aplicaciones en este caso son las siguientes:

a) Resuelve problemas de como lograr una mejor distribución de tierra para cultivos, con la finalidad que los rendimientos sean óptimos en base a una disponibilidad de recursos escasos.

b) Trata de optimizar el uso de los recursos, de tal forma que se minimicen los costos y se incrementen las utilidades. Por ejemplo:

1) Un ganadero que cría ganado para engorda, debe decidir qué tipo de pasto proporcionará al ganado, y qué plan de engorda debe seguir, con objeto de obtener el máximo beneficio.

2) Un agricultor debe decidir qué cultivos plantar durante la siguiente temporada, tomando en cuenta la cantidad de agua de que dispone; la tierra cultivable , y alguna otra restricción en cuanto a cantidades que debe proporcionar de uno de los cultivos obligado por un contrato, con objeto de obtener el máximo beneficio monetario.

2.2.2. Sector Industrial.

En lo que respecta a este sector, es donde mayor aplicación ha tenido la Programación Lineal por resultar de suma utilidad, puesto que las formas de producción no son tan variables como en la agricultura. Concretamente la técnica resuelve problemas de:

a) Como lograr mejor empleo de la maquinaria para que rinda una mayor productividad al menor costo posible.

b) La mejor manera de asignar a equipos de trabajo, para que al mínimo tiempo posible incremente la producción.

c) La forma de optimizar el aprovechamiento de materias primas, en la elaboración de productos manufacturados, etc.

Como ejemplos tenemos:

1) Una fábrica de alimento para pollos, produce dos marcas que se hacen con harina de pescado y nutriente. Tiene además capacidad limitada en el empaque de uno de sus productos. Debe decidir como emplear esta materia prima y su capacidad, en la fabricación, para obtener el máximo beneficio posible.

2.2.3. Sector Servicios.

Dentro de este sector terciario de la economía del país, la Programación Lineal se aplica a problemas como son:

a) Problemas de transporte de mercancías, señalando la forma y medio más conveniente para transportar las mercancías, de los centros de producción a los centros de consumo, o sea maximizar los beneficios al menor costo posible.

b) Problemas de trasbordo, que es un caso general del transporte, por tanto; la finalidad es minimizar los costos de transporte para lograr maximizar los beneficios.

Tenemos los siguientes ejemplos:

1) La CFE debe decidir cómo distribuir el carbón de que dispone entre las termoeléctricas que alimentan al Valle de México, sujetas a las restricciones de cantidad de carbón y a la demanda y eficiencia de las termoeléctricas, de manera que

incurra en el mínimo costo de producción, de carbón y transporte.

2) PEMEX debe decidir cómo encauzar el petróleo crudo que llega a una de sus refinerías tomando en cuenta la capacidad y eficiencia de los procesos, así como la demanda de los productos terminados, de manera que se minimice el costo de la operación.

Después de haber hecho un breve bosquejo de algunas de las aplicaciones de la Programación Lineal, se demuestra lo indespensable que resulta ser la técnica en la solución de problemas económicos para la optimización en el uso de los recursos; tanto en la iniciativa privada como en el sector público, esta herramienta ha sido aplicada a una gran variedad de problemas de optimación, obteniéndose resultados favorables en beneficio del crecimiento en la economía del país.

2.3. Limitaciones de la Programación Lineal.[9]

La Programación Lineal como toda herramienta, está sujeta a ciertas limitaciones que a continuación mencionamos:

PROPORCIONALIDAD .- Debido a que la función objetivo y restricciones son lineales, ésto implica que la medida y efectividad de los recursos usados debe ser proporcional al nivel de cada actividad por separado.

ACUMULACION .- Esto se da cuando exite alguna interacción independiente de que la medida de efectividad y el uso de los

[9] IDEM PAG 27

recursos se comporte linealmente. Esto resulta de la proporcionalidad, es decir, que los cambios que se hagan en la aplicación de los recursos, de la misma manera se harán en las actividades a desarrollar de tal forma que los incrementos o decrementos realizados sean proporcionales.

DIVISIBILIDAD.- No es entera; es decir, las actividades pueden realizarse en cualquier extensión positiva mientras se disponga de recursos existentes.

DETERMINISTICA.- Se refiere a la existencia de un número finito de actividades disponibles, así como de la cantidad de recursos con que se cuenta.

Las limitaciones antes enumeradas se pueden considerar como endógenas, es decir, son propias de la técnica de la Programación Lineal; pero de la misma forma, existen limitaciones exógenas, dadas por los cambios constantes de la economía del país, que en un momento dado, traen consigo modificaciones en los programas de producción.

2.4. El Problema de la Programación Lineal.

El problema general de la Programación Lineal está representado por una serie de ecuaciones e inecuaciones formadas por un número finito de variables.

Para la solución específica a problemas de programación lineal, utilizamos al Método Gráfico cuando se trate de un problema teórico de dos variables y por el Método Simplex cuando intervienen más de dos variables; de igual forma se estará en capacidad de dar a la solución obtenida la interpretación económica correspondiente.

Un programa lineal puede ser del tipo Maximización ó Minimización, las Restricciones pueden ser del tipo $(\leq,=,\geq)$ y las variables pueden ser o no-negativas o irrestrictas en el signo.

Un Modelo de Programación Lineal se define:[4]

$$\text{MAX o MIN } Z = c_1X_1 + c_2X_2 + \ldots + c_nX_n \;\Big\}\; \text{FUNCION OBJETIVO}$$

sujeto a

$$a_{11}X_1 + a_{12}X_2 + \ldots + c_{1n}X_n (\leq,=,\geq) b_1$$

$$a_{21}X_1 + a_{22}X_2 + \ldots + c_{2n}X_n (\leq,=,\geq) b_2 \text{ RESTRICCIONES } a_{m1}X_1 + a_{m2}X_2 + \ldots$$

$$+ c_{mn}X_n (\leq,=,\geq) b_m \quad X_1 \geq 0, X_2 \geq 0, \ldots, X_n \geq 0$$

donde c_j, b_i y a_{ij} $(i=1,2,\ldots m; j=1,2,\ldots n)$ son constantes determinadas por la tecnologia del problema y x_j son las variables de decisión. Unicamente un signo $(\leq,=,\geq)$ ocurre para cada restricción no obstante que todas las variables son declaradas no-negativas. La variable inrrestricta de no-negatividad es esencial para el desarrollo del metodo de solucion para programacion lineal.

Los modelos de programación lineal a menudo representan problemas de "asignación" en los cuales los recursos limitados se asignan a un número de actividades. en función de la formulación anterior los coeficientes c_i, a_{ij} y b_i se interpretan físicamente como siguen: Si b_i es la cantidad

[4] HAMDY A. TAHA,"INVESTIGACION DE OPERACIONES" PAG 30

disponible de recurso i, entonces a_{ij} es la cantidad de recurso i que debe asignarse a cada unidad de la actividad j. El valor por unidad de la actividad j es igual a c_j.

Debido a que los modelos de programación lineal se presentan de una variedad de formas [max o min para la funcion objetivo y (≤, =, ≥) para las restricciones] es necesario modificar estas formas para que se ajusten al procedimiento de solución. Introduciendo dos formas para este propósito: La forma canónica y la forma estándar. Las cuales a continuación se presentan:

2.4..1. Forma Canónica.[5]

El problema de Programción Lineal general definido anteriormente, puede ser puesto siempre de la forma siguiente, la cual se conocerá como la forma canónica.

$$\text{MAXIMIZAR } X_0 = \sum_{j=1}^{n} c_j x_j$$

SUJETO

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i, \quad i=1,2,..,m$$

$$x_j \geq 0 \quad j=1,2,..,n$$

Las caraterísticas de esta forma son:

1) Todas las variables de decisión son no-negativas.

2) Todas las restricciones son del tipo (menor o igual).

3) La Función objetivo es del tipo de Maximización.

Un problema de Programción Lineal puede ponerse en la

[5] IDEM PAG 31

forma canónica por el uso de cinco transformaciones elementales.

1.- La Minimización de una función, fx, es matemáticamente equivalente a la Maximización de la expresión negativa de esta función, -fx. ejemplo:

$$\text{MINIMIZAR } X0 = C1X1 + C2X2 + \ldots\ldots + CnXn$$

es equivalente a

$$\text{MAXIMIZAR } g0 = -x0 = -c1x1 - c2x2 - \ldots\ldots - cnxn$$

2.- Una desigualdad en una dirección ($\geq$ o $\leq$) puede cambiarse a una desigualdad en la dirección opuesta ($\leq$ o $\geq$) multiplicando ambos lados de la desigualdad por -1.

ejemplo:

$$a1x1 + a2x2 \geq b$$

es equivalente a

$$-a1x1 - a2x2 \leq -b$$

3.- Una ecuación puede ser remplazada por 2 desigualdades en direcciones opuestas.

ejemplo:

$$a1x1 + a2x2 = b$$

es equivalente a

$$a1x1 + a2x2 \leq b \text{ y } a1x1 + a2x2 \geq b$$

4.- Una restricción de desigualdad con su lado izquierdo en forma de valor absoluto puede cambiarse a dos desigualdades regulares.

ejemplo para $b \geq 0$:

$$|a1x1 + a2x2| \leq b$$

es equivalente a

$$a1x1 + a2x2 \geq -b \text{ y } a1x1 + a2x2 \leq b$$

5.- Una variable que es irrestricta en signo es equivalente a la diferencia entre dos variables no-negativas. Por consiguiente si x es inrestricta en signo puede remplazarse por $(x^{+} - x^{-})$donde $x^{+} \geq 0$ y $x^{-} \geq 0$.

2.4.2. Forma Estándar.[6]

La forma Estándar se utiliza directamente para resolver el modelo . Las características de la forma Estándar son:

1.- Todas las restricciones son ecuaciones excepto para las restricciones de no-negatividad que permanecen como desigualdades (≥ 0).

2.- Los elementos del lado derecho de cada ecuación son no-negativos.

3.- Todas las variables son no-negativas.

4.- La Función Objetivo es de tipo Max o Min.

Las restricciones de desigualdad pueden cambiarse a ecuaciones introduciendo (sumando o restando) en el lado izquierdo cada una de tales restricciones una variable no negativa.

Estas nuevas variables se conocen como variables de holgura y se suman si la restricción es (≤) o se restan si la restrición es (≥). El lado derecho puede hacerse siempre

[6] IDEM PAG 33

positivo multiplicando ambos lados de la ecuación resultante por (-1) siempre que sea necesario. Las características restantes pueden realizarse usando las transformaciones elementales introducidas en la forma Canónica.

ejemplo:

$$a1x1 + a2x2 \geq b \ , \ b \geq 0$$

se cambia a la forma Estándar

$$a1x1 + a2x2 - s1 = b$$

donde $s1$ es ≥ 0.

tambien las restricción

$$p1x1 + p2x2 \leq q \ , \ q \geq 0$$

cambia a

$$p1x1 + p2x2 + s2 = q$$

donde $s2 \geq 0$.

Para ilustrar mejor la forma en que se construye el modelo de Programación Lineal, se mostrará el siguiente ejemplo:

Una fábrica de muebles tiene que producir 4 diferentes tipos de libreros, para lo cual necesita hacerlos con un mínimo de tiempo . Para el primer librero se dispone 3 horas del trabajador A y 5 del trabajador B; para el segundo librero se dispone de 7 Hrs. del A, y 6 Hrs. del B; del tercer librero se dispone de 4 Hrs. del A, y 1 Hrs. del B; Y para el último librero se dispone de 2 Hrs.para A, y 9 Hrs. para B.

Para los libreros producidos se esperan obtener los siguientes ingresos . Para el librero (M1) un ingreso de 14; para el librero (M2) un ingreso de 23; el tercer librero (M3) el ingreso es de 13; y para el último librero (M4) el ingreso será de 12.

¿Qué cantidad de tiempo se necesitará para producir los 4 tipos de libreros si el Trabajador A dispone de 50 Hrs. y el Trabajador B dispone de 55 Hrs.?

Planteamiento del problema:

MAXIMIZAR $Z = 14M1 + 23M2 + 13M3 + 12M4$

SUJETO A

$$3M1 + 7M2 + 4M3 + 2M4 \leq 50$$

$$5M1 + 6M2 + M3 + 9M4 \leq 55$$

$$M1, M2, M3, M4 \geq 0$$

Para la solución a problemas de Programación Lineal se estima conveniente tratar algunos conceptos básicos.

a) Se le llama SOLUCION BASICA[7] a una solución al modelo de Programación Lineal en forma canónica, que se obtiene de fijar en cero tantas variables como sea necesario para obtener un sistema en igual número de ecuaciones que de variables. Tal es el caso como el ejemplo que ponemos:

$$3\,x1 + 2\,x2 - x3 + x4 \text{ x } 0\,x5 = 8$$

[7] VICTOR FLORES Z.,"INGENIERIA DE SISTEMAS" PAG 71

$$6\ x1 - 4\ x2 + 2\ x3 + 0\ x4 + x5 = 80$$

Características :

1.- Numero de Ecuaciones igual a 2.
2.- Numero de Variables igual a 5.
3.- Por lo tanto $m <= n$ donde m = número de ecuaciones y n = número de variables.
4.- Se debe marcar en las ecuaciones originales, cuando menos un coeficiente igual a 1 en cada ecuación respectivamente.
5.- Los Coeficientes deben estar en distintas variables.
6.- Tanto en la parte de arriba, como abajo de los diferentes coeficientes deben tener un valor igual a Cero. Representando la Forma Canonica.
7.- Por lo tanto la Solución Basica en el ejemplo sera $x1=0$, $x2=0$, $x3=0$, $x4=8$ y $x5=80$.
8.- Cada Variable asociada a un 1 marcado es una variable basica y las otras variables son no basicas.

x1	x2	x3	x4	x5	b
0	0	0	1	0	8
0	0	0	0	1	80

b) Una solución básica que cumple además con las restricciones de no negatividad de las variables se llama SOLUCION BASICA FACTIBLE.

En el ejemplo anterior la Solución Básica Factible es la siguiente tabla:

x1	x2	x3	x4	x5	b
0	0	0	1	0	8
0	0	0	0	1	80

Donde se observa que a cada ecuación le corresponde una solución basica representada por el (1), que contiene ceros tanto arriba como abajo. A esta forma se le llama Forma

Canónica.

c) Al conjunto de variables para los que se resuelve el sistema se les denomina VARIABLES BASICAS ; a las que se fijan en cero se les llaman VARIABLES NO BASICAS.

NO BASICAS / BASICAS	X1	X2	X3		B
X4	0	0	0		8
X5	0	0	0		80

2.5. <u>Solución de Problemas Lineales</u>.

Para la solución a Problemas de Programación Lineal existen tres métodos diferentes los cuales son:

2.5.1. <u>Método Gráfico</u>.

El propósito del método gráfico no es proveer un método práctico para resolver programas lineales, ya que usualmente los problemas prácticos incluyen un gran número de variables. En lugar de ésto, el método demuestra los conceptos básicos para desarrollar la técnica algebraica para programas lineales con más de dos variables.

La idea del método es graficar el espacio de soluciones (factible), el cual se define como el espacio encerrado por las restricciones. La solución óptima es el punto (en el espacio de soluciones) que maximiza el valor de la función objetivo.

Por ejemplo: Una industria que produce bicicletas, cada una de las cuales se debe procesar a través de dos máquinas. La máquina 1 tiene un máximo de 120 horas disponibles y la máquina 2 tiene un máximo de 180 horas disponibles, la manufactura de una Bicicleta de tipo A requiere de 6 horas en la central de máquina 1 y 3 horas en la central de la máquina 2; la manufactura de una Bicicleta de tipo B, requiere de 4 horas en la máquina 1 y 10 horas en la central de la máquina 2. Si el Beneficio por Bicicleta del tipo A es de 45 y por la tipo B es de 55. Determinar el número de Bicicletas que deberían de manufacturarse para obtener un beneficio máximo.

Denotación:

X_1=Bicicleta Tipo A

X_2=Bicicleta Tipo B

$$\text{MAXIMIZAR } Z = 45X1 + 55X2$$

SUJETA A

$$6X1 + 4X2 \leq 120$$

$$3X1 + 10X2 \leq 180$$

$$X1, X2 \geq 0$$

SOLUCION

6X1+4X2=120 3X1+10X2=180

SI X1=0SI X2=0 SI X1=0SI

X2=0

6(0)+4X2=1206X1+4(0)=1203(0)+10X2=1803X1+10(0)=180

4X2 = 120	6X1 = 120	10X2 = 180	3X1=180
X2 = 120/4	X1 = 120/6	X2 = 180/10	X1=180/3
X2 = 30	X1 = 20	X2 = 18	X1=60

El Polígono de solución factible se obtuvo graficando las ecuaciones:

$$6x1 + 4x2 = 120$$

$$3x1 + 10x2 = 180$$

$$x1 = 0 , x2 = 0$$

cualquier punto que se encuentre en el Polígono sombreado, o sobre uno de sus lados corresponde a un par de valores x_1 y x_2 que satisfacen las restricciones del problema y por consiguiente es una solución factible.

INTERPRETACION

$$z = 45\ x1 + 55\ x2$$

$$(0,18) \quad 45(0) + 55(18) = 990$$

$$(10,15) \quad 45(10) + 55(15) = 1275$$

$$(20,0) \quad 45(20) + 55(0) = 900$$

Para la interpr e tación del anterior modelo podemos concluir que si se producen 10 Bicicletas del Tipo A y 15 Bicicletas del Tipo B, tendremos un Beneficio igual a $ 1275.

2.5.2. Método Simplex.

Consiste en vaciar los datos ya ordenados del problema planteado en una tabla, la cual se llama "Tabla o Cuadro Simplex"y partiendo de ésta, mediante una serie de interacciones, se logra obtener la solución.[9]

[9] HAMDY A. TAHA,"INVESTIGACION DE OPERACIONES", PAG 47-54

La base del método simplex que garantiza tal sucesión de soluciones básicas está formada por dos condiciones fundamentales:

1) La condicion de Optimidad asegura que nunca se encontrará una solución inferior (relativa al punto de solución actual).

2) La condición de factibilidad garantiza que partiendo de una solución básica factible únicamente se encontrarán durante el cálculo soluciones básicas factibles.

Para este método también se requiere de saber la cantidad de recursos no utilizados en el problema, cuya función principal desde el punto de vista matemático es romper las inecuaciones o desigualdades llamadas restricciones que se presentan en todo problema de programación lineal. A esta actividad se le conoce como Variable de Holgura.

Las variables de Holgura, desde el punto de vista económico representan la cantidad de recurso no utilizado en el problema es por ello que el coeficiente de estas variables siempre será de cero. Para resolver un problema empleando la Tabla Simplex, imaginaremos que estos recursos no utilizados se emplean en la producción únicamente para establecer las igualdades en las restricciones y al mismo tiempo para formar una matriz unitaria en el sistema de ecuaciones (restricciones) que es condición primordial para el empleo de la Tabla Simplex. Las variables de holgura son colocadas en orden sistemático; en la primera inecuación se introduce la primera variable de holgura, la cual ocupará el subíndice inmediato que precede a la última variable real; en la segunda ecuación introducimos la segunda variable, con el subíndice que precede a la primer variable de holgura; la tercer variable de holgura se colocará en la restricción que continua con el subíndice que precede a la segunda restricción, etc., y así sucesivamente hasta establecer todas las igualdades en todas las restricciones.

Para poder trabajar con la Tabla Simplex es fundamental y necesario que reuna las siguientes condiciones:

1.- debe existir una matriz idéntica o unitaria.

2.-Los elementos de la columna B, o sea los términos independientes deben ser positivos para que tenga sentido el problema, ya que desde un punto de vista económico no es lógico tener valores negativos.

3.-Debe aparecer el precio neto o costo (Cj)

4.-El cuadro también debe contener la fila de costo de oportunidad, que es lo que se deja de ganar por dedicarse a otra actividad (Zj).

5.- Debemos contar con el precio de sombra, que representa la variación del problema por adicionar una unidad más del producto (Z_j-C_j).

2.5.3. Método Dual-Simplex.

Uno de los más importantes descubrimientos en el desarrollado comtemporáneo de la Programación Lineal, fue el concepto de Dualidad, así como sus múltiples e importantes ramificaciones. Esta innovación descubrió que cualquier problema de Programación Lineal, está asociado con otro problema de Programación Lineal llamado Dual. Aunque al parecer, a primera vista, es una relación artificial la que existe entre el Problema Dual y el Problema original (llamado el Primal).[10]

Consideraremos el problema general de la Programación Lineal.

$$\text{MAXIMIZAR} \quad Z = c_1X_1 + c_2X_2 + \ldots + c_nX_n$$

sujeto a

$$a_{11}X_1 + a_{12}X_2 + \ldots + c_{1n}X_n \leq b_1$$

$$a_{21}X_1 + a_{22}X_2 + \ldots + c_{2n}X_n \leq b_2$$

$$a_{m1}X_1 + a_{m2}X_2 + \ldots + c_{mn}X_n \leq b_m$$

$$X_1 \geq 0 \,,\ X_2 \geq 0 \,, \ldots ,\ X_n \geq 0$$

El Problema del Dual correspondiente, es el que resulta de la transposición de filas y columnas de los coeficientes de la Función Objetivo y el lado derecho del sentido de las desigualdades, invirtiendo las inecuaciones y Minimizando en lugar de Maximizar.

[10] IDEM PAG 81-99

$$\text{MINIMIZAR } Z = b_1Y_1 + b_2Y_2 + \ldots + b_mY_n$$

sujeto a

$$a_{11}Y_1 + a_{12}Y_2 + \ldots + c_{1n}Y_n \geq c_1$$

$$a_{21}Y_1 + a_{22}Y_2 + \ldots + c_{2n}Y_n \geq c_2$$

$$\ldots\ldots\ldots\ldots\ldots\ldots$$

$$a_{m1}Y_1 + a_{m2}Y_2 + \ldots + c_{mn}Y_n \geq c_n$$

$$Y_1 \geq 0 \,,\ Y_2 \geq 0 \,, \ldots ,\ Y_n \geq 0$$

Entonces, los coeficientes de la i-esima restricción del problema dual son los coeficientes de X_i en las restricciones del problema Primal, y viceversa; ademas, el sentido de la i-esima de la restricción del problema dual es el coeficiente de X_i en la Función Objetivo del Problema Primal y viceversa.

De ahi que hay una variable Dual por cada restricción primal y una restricción dual por cada variable primal.

Un Problema Dual siempre existe para cualquier problema de Programación Lineal. En particular, el supuesto de que la restricción i-esima es la forma de una igualdad en lugar de una desigualdad; entonces, el Problema dual deberá construirse justamente como lo hicimos en el caso anterior, sólo que i-esima variable dual deberá ser una restricción con signo tal que no sea negativo. Similarmente, si las restricciones son no negativas son cambiadas por la i-esima variable primal, el resultado se modifica en el problema dual si la i-esima restrición es de la forma de una ecuación en lugar de una inecuación.

Es importante notar que los teoremas siguientes se desarrollan basados en la hipótesis de que el Problema Dual se ha obtenido de la forma estándar del Simplex. Es así porque los Cálculos del Método Simplex siempre parten de la Forma Estándar. Consecuentemente, será erróneo, en general, aplicar los resultados de estos cálculos a un Dual obtenido de cualquier otra forma del Primal.

Teorema I.- EL DUAL DEL DUAL ES EL PRIMAL.

El teorema implica una relación sistemática completa entre el Problema Dual y el Primal. En efecto, es inmaterial el Problema que es llamado Dual y el que es llamado Primal. Cualquier cosa que se desee decir de un Problema con respecto del otro se dirá de ambos.

Será necesario introducir una notación diferente antes de explicar el Teorema Fundamental de la Dualidad (conocido como el Teorema del dual).En general, cuando aparezca una variable con asterisco como exponente, denotará un valor óptimo, ejemplo ;

Definición: sea $Z^*_x, Z^*_y, X^*_j, Y^*_i$ denote el valor Optimo de Z_x, Z_y, X_j, Y_i, respectivamente.

$$Z^*_x = \sum_{j=1}^{n} U_j X^*_j \qquad y \qquad Z^*_y = \sum_{i=1}^{m} B_i Y^*_i$$

Teorema II (TEOREMA DUAL).

Asumiendo que existe una solución factible finita, tanto para el problema primal como para el dual, entonces existe una solución óptima finita para ambos problemas y tenemos:

$$Z^{*}_{x} = Z^{*}_{y}$$

Esta relación que se da en el Método Simplex tiene importantes implicaciones; una es que el Teorema I puede ser aplicado directamente a cualquier problema (al primal o al dual), la ventaja del Dual, es que a veces requiere de un menor esfuerzo en el cómputo; otra ventaja es que la alternativa del procedimiento de solución puede ser establecida para resolver el Problema Primal, operando directamente por el Problema Dual.

INTERPRETACION ECONOMICA

La solución óptima provee una interpretación económica muy importante al problema primal. Explicando esto, sea $Y^{*}i$ el valor óptimo de la i-esima variable dual y (i= 1,2,3......,m) y considera la i-esima restrición en el problema primal, $B_{i1} B_{1}+ B_{i2}B_{2}+B_{i3}+ \ldots +B\ x_{n} \leq B_{i}$(recuerdese que B_{i} es la cantidad de recurso i disponible que se puede aprovechar en la producción o en la elaboración del producto). El valor óptimo de la función objetivo $(Z_{x}=Z^{*}_{x})$, deberá ser interpretado como la utilidad óptima total obtenida. En este caso, Y^{*}_{i} indica el valor en que se deberá incrementar la utilidad (disminuir) si la cantidad de recurso i disponible fuera incrementada (disminuida) en una determinada proporción (esta proporción es el recurso b_{i}en el cual la base óptima original no fue alterada). Entonces Y^*_i puede ser interpretada como el valor marginal del recurso i.

En el Lenguaje Económico, la solución dual representa los precios sombras, o sea, los costos marginales; de ahí, que sea factible detectar qué producto es el más conveniente producir, al disponer de una cantidad adicional de recurso y a la vez, las limitantes que en realidad resultan ser restricciones.

En esta investigación se verá la solución de problemas de Programación Lineal por el Método Dual-Simplex cuyo método de solución, consiste en vaciar los datos ya ordenados del problema planteado en una tabla, la cual se llama Tableau de Tucker y partiendo de ésta, mediante una serie de interaciones se logra obtener la solución por medio del algorítmo de intercambio de Jordan que consiste en:

2.5.4. Tableau de Tucker para el Método Simplex.

El problema de programación lineal escrito en forma estándar es de la siguiente forma:[11]

$$\text{MAXIMIZAR } Z = cX$$
$$\text{SUJETA A } AX \leq b$$
$$X \geq 0$$

Este problema se puede reescribir como sigue :

$$\text{MINIMIZAR } Z = cX$$
$$\text{SUJETA A } y = A(-X) + b \geq 0$$
$$X, \; y \geq 0$$

Las restricciones son ahora un conjunto de relaciones

[11] GRACIELA BUENO DE A.,"INTRODUCCION A LA PROGRAMACION LINEAL Y ANALISIS DE SENSIBILIDAD",PAG 49-50

lineales, con la restricción adicional de que tanto las variables independientes (X) como las dependientes (y) son no-negativas.

Este problema se puede resumir en una tabla, que en la terminología clásica usada en el Método Dual-Simplex se llama tableau y que es de la siguiente forma:

		Variables Independientes			
		$-X_1$	....	$-X_n$	1
y_1	=	a_{11}		a_{1n}	b_1
y_2	=	a_{21}		a_{2n}	b_2
...			....		...
y_i	=	a_{i1}		a_{in}	b_i
...			....		...
y_n	=	a_{m1}		a_{mn}	b_m
z	=	$-c_1$	....	$-c_n$	0

El tableau contie n e las relaciones lineales que representan las restricciones y la función objetivo.

La hilera i de la tabla se leerá como $y_i = a_{i1}(-x_1) +...+ a_{in}(-x_n) + b_i$, que es igual a multiplicar la hilera i de la matriz A por el vector (-x) y sumarle b_i, es decir, $y_i = A_i(-x)+b_i$.

Contando con todos los elementos anteriores tenemos las condiciones necesarias para resolver cualquier tipo de problemas de Programación Lineal.

Una vez planteado y analizado el problema, lo que procede es su solución . Como se ha mencionado desde un inicio, en la actualidad la obtención de las soluciones es un problema poco importante, pues existen diversos paquetes que auxilian y logran rápidamente la solución a modelos de suma complejidad. Es decir, solución por computadora en nuestro ejemplo que hemos venido desarrollando, se mostrará la solución a través del Método Dual-Simplex y además la salida de resultados utilizando el Paquete "PROLIN", que esta diseñado exclusivamente para aplicarse en la solución de problemas de Programación Lineal.

INTERCAMBIO DE JORDAN

MAXIMIZAR Z= 14M1 + 23M2 + 13M3 + 12M4

SUJETO A

3M1 + 7M2 + 4M3 + 2M4 ≤ 50

5M1 + 6M2 + M3 + 9M4 ≤ 55

M1,M2,M3,M4 ≥ 0

Para llegar al modelo matemático con mayor facilidad, es conveniente elaborar una tabla presentando las interrelaciones que existen entre los elementos que se dan en el problema.

	M 1	M 2	M 3	M 4	DISPONIBLE
A	3	7	4	2	50
B	5	6	1	9	55
	14	23	13	12	INGRESO

Dicha tabla presenta la primera solución óptima factible, que es el objetivo del método dual simplex y después a través de una serie de interaciones nos permitirá llegar a la solución óptima de dicho problema, los cuales son realizados a través del Paquete PROLIN de la siguiente manera presentando sus respectivas interaciones y su tableu óptimo.

	M 1	A	M 3	M 4	B
M2	0.43	0.14	0.57	0.29	7.14
B	2.43	-0.86	-2.43	7.29	12.14
Z	-4.14	3.29	0.14	-5.43	164.29

	M 1	A	M 3	B	B
M2	0.33	0.18	0.67	-0.04	6.67
M4	0.33	-0.12	-0.33	0.14	1.67
Z	-2.33	2.65	-1.67	0.75	173.33

	M 4	A	M 3	B	B
M2	-1.00	0.29	1	-0.18	5
M1	3	-0.35	-1	0.41	5
Z	7	1.82	-4	1.71	185

TABLEAU OPTIMO

	M 4	A	M 2	B	B
M 3	-1	0.29	1	-0.18	5
M 1	2	-0.06	1	0.24	10
Z	3	3	4	1	205

Para poder interpretar la salida de resultados en problemas de Programación Lineal, se deben considerar las siguientes definiciones.[12] Considerando que no todos los problemas planteados tienen la misma interpretación.

[12] VICTOR RIOS G., "INVESTIGACION DE OPERACIONES", PAG 31-34

Función Objetivo.- Es la función lineal que indica la finalidad que se persigue en el problema, en el caso de la economía, maximizar ganancias o minimizar costos, es de la forma $Y = a + b x$.

Costos.- Se refiere a los coeficientes de las variables de la función objetivo e implica el total de recursos monetarios utilizados en la producción de un producto determinado.

Actividad.- Es un método específico que conduce a la realización de un fin determinado. Por ejemplo, producir muebles, automóviles, vasos, etc.; en una fábrica, de tal forma que los costos sean mínimos y sus utilidades sean atractivas. Así como cultivar ajonjolí, maíz, frijol, etc., en una granja, de tal forma que se obtenga la producción deseada.

Alternativas.- Implica las diferentes formas de combinación que se pueden hacer para lograr un objetivo. Por ejemplo, las formas de cultivo en que se deben combinar los tres productos anteriores, de tal manera que se obtenga una utilidad óptima.

Restricciones.- Es la condición en que normalmente se presentan los recursos escasos con que se cuenta para producir. Si no hay restricciones prácticamente no hay problema (pueden aparecer como ecuaciones o bien como inacueciones).

Linealidad.- O principio lineal, significa que la modificación en el tiempo para producir cierta cantidad de un producto deberá ser igual al cambio en la utilidad marginal esperada.

Costo de Oportunidad.- Es lo que se deja de ganar por dedicarse a otra actividad.

Precio Sombra.- Representa la medida en que varía un programa por adicionar una unidad más del producto y siempre será un indicador no real, pero de gran utilidad práctica para el análisis económico.

Programa Factible.- Representa la medida en que tome en consideración las restricciones, es decir, el que utiliza tan sólo los recursos disponibles y produce cantidades positivas óptimas.

Programa Optimo.- O solución óptima, es la solución básica factible que nos conduce a obtener una utilidad o beneficio mayores, esto es, a maximizar las ganancias o minimizar los costos de un programa.

Variable Entrante.- Es la variable o actividad que entra o se incorpora a la solución del problema, ella incrementa la función en cierta cantidad.

Variable Saliente.- Es la actividad que deja de intervenir en la producción por resultar ineficiente en relación con otras actividades.

Minimizar.- Es la reducción de los costos en la generación de un producto en su óptima expresión.

Maximizar.- Es elevar lo más posible las utilidades en la solución de un problema de Programación Lineal.

La solución óptima a nuestro problema a continuación damos la interpretación Matemática y Económica de la salida de resultados:

$$M3 = 5, \; M4 = 0, \; M1 = 10 \text{ Y } M2 = 0$$

$$Z = 205$$

En términos económicos esto quiere decir que si producimos 10 libreros del tipo M1 y 5 libreros del tipo M3 el máximo beneficio que obtendrá la fábrica es 205.

$$14M1 + 23M2 + 13M3 + 12M4$$

$$14\ (10) + 23\ (0) + 13\ (5) + 12\ (0) = 205$$

$$140 + 0 + 65 + 0 = 205$$

$$205 = 205$$

Los precios sombras o costos de oportunidad están representados por la última hilera del Tableau óptimo, los cuales son los siguientes:

1) Para variables no Seleccionadas:

$$M4 = 3 \text{ y } M2 = 4$$

Estos valores r e presentan para nuestro ejemplo, la cantidad en que disminuirá nuestro ingreso si tratamos de fabricar estos tipos de Libreros. Por ejemplo si se produce un librero de tipo M2 nuestro máximo beneficio desminuirá en 4 unidades y en forma analóga sucede lo mismo la variable M4.

2) Para los recursos observamos que éstos son consumidos en su totalidad y por consiguiente son escasos, lo cual se puede observar de la siguiente manera:

TRABAJADOR TIPO A

$$3M1 + 7\ M2 + 4M3 + 2M4 >= 50$$

$$3\ (10) + 7(0) + 4(5) + 2(0) >= 50$$

$$50 >= 50$$

TRABAJADOR TIPO B

$$5M1 + 6M2 + 1M3 + 9M4 >= 55$$

$$5(10) + 6\ (0) + 1(5) + 9(0) >= 55$$

$$55 > = 55$$

En lo que se refiere a sus costos de oportunidad sus resultados son los siguientes A = 3 y B = 1. Estos datos representan en nuestro ejemplo la cantidad que aumentaría nuestro máximo beneficio si utilizamos más horas de trabajo de cada recurso . Es decir si utilizamos una hora de trabajo del recurso A en producir más libreros la función objetivo se verá incrementada en 3 unidades y recíprocamente sucede con el

A continuación se mostrará la solución de este modelo utilizando el Paquete "PROLIN" y otros paquetes. Como ya se mencionó, dicho paquete es de muy fácil manejo pues se trata de una interacción, en la medida en que el mismo programa conduce al usuario. En el capítulo V, se podrá observar una gran gama de ejercicios desarrollados utilizando el "PROLIN", así como la solución a través de otros paquetes (QSB,LINDO).

El archivo "d:mueble.dat" contiene : 4 variables independientes
2 restricciones

	x1	x2	x3	x4	signo	B
y1	3.00	7.00	4.00	2.00	<	50.00
y2	5.00	6.00	1.00	9.00	<	55.00
z	14.00	23.00	13.00	12.00		0.00

---

	x1	y1	x3	x4	signo	B
x2	0.43	0.14	0.57	0.29	<	7.14
y2	2.43	-0.86	-2.43	7.29	<	12.14
z	-4.14	3.29	0.14	-5.43		164.29

---

	x1	y1	x3	y2	signo	B
x2	0.33	0.18	0.67	-0.04	<	6.67
x4	0.33	-0.12	-0.33	0.14	<	1.67
z	-2.33	2.65	-1.67	0.75		173.33

---

	x4	y1	x3	y2	signo	B
x2	-1.00	0.29	1.00	-0.18	<	5.00
x1	3.00	-0.35	-1.00	0.41	<	5.00
z	7.00	1.82	-4.00	1.71		185.00

---

	x4	y1	x2	y2	signo	B
x3	-1.00	0.29	1.00	-0.18	<	5.00
x1	2.00	-0.06	1.00	0.24	<	10.00
z	3.00	3.00	4.00	1.00		205.00

TABLEAU OPTIMO

Input Data Describing Your Problem mueble Page 1

Max +14.0000X1 +23.0000X2 +13.0000X3 +12.0000X4
Subject to
(1) +3.00000X1 +7.00000X2 +4.00000X3 +2.00000X4 < +50.0000
(2) +5.00000X1 +6.00000X2 +1.00000X3 +9.00000X4 < +55.0000

Summarized Results for mueble Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+10.000000	0	4	X4	0	+3.0000000
2	X2	0	+4.0000000	5	S1	0	+3.0000000
3	X3	+5.0000000	0	6	S2	0	+1.0000000

Maximum value of the OBJ = 205 Iters. = 4

```
MAX      14 X1 + 23 X2 + 13 X3 + 12 X4
SUBJECT TO
        2)    3 X1 + 9 X2 + 4 X3 <=    50
        3)    5 X1 + 6 X2 + X3 + 9 X4 <=    55
END

: solution

         OBJECTIVE FUNCTION VALUE

  1)          219.166700

 VARIABLE         VALUE          REDUCED COST
       X1          .000000           1.416667
       X2          .000000          11.250000
       X3        12.500000            .000000
       X4         4.722222            .000000

      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)          .000000           2.916667
       3)          .000000           1.333333

NO. ITERATIONS=         2
:
```

PROGRAMA FUENTE DE PROLIN

ESTE PROGRAMA FUE REALIZADO EN EL LENGUAJE PASCAL VERSION 6.0

```
PROGRAM PROLIN;
    uses crt,util,dos,turbo3,printer;
    type
     regist      = record
     g           : array [1..10] of real;
     f,k         : array [1..10] of string [8];
     numo,numv : integer;
     sig         : array [1..10] of char;
     end;
     const
     MAX_VAR=10;
     signo    : set of char=['<','>'];
     var
     gd          : array[1..max_var,1..max_var] of real;
     una,jj,h,n,ki,i,j,zc,obs,varbl,r,s,x,prim,cont  : integer;
     archivo                                    : file of regist;
     registro                                  : regist;
     vx                                         : string[15];
     cc,ww,vc                                   : char;
     v5,v6,sale,entra                          : string[5];
     sgn,opt,dr                                 : string[3];
     gr,bi,min,minh                             : real;
     cor,ch                                     : char;
     recu                                       :boolean;
     VALIDO                                     :CARACTERES;
     toma4,TOMA2,TOMA,TOMA3        :BYTE;
     posx,posy,DESTINO             :integer;

PROCEDURE WRITEXY(x,y:byte;s:string);
    begin
     gotoxy(x,Y);
     write(s);
    end;

PROCEDURE CERRAR_VENTANA;
    begin
      window(1,1,80,25);
      textbackground(negro);
      clrscr;
    end;

PROCEDURE INICIA;
      begin
        with registro do
          begin
            for i:=1 to max_var do
                BEGIN
                  for j:=1 to max_var do
                  g[j]:=0;
                END;
            for i:=1 to max_var do
                  BEGIN
                    for j:=1 to max_var do
                    gD[I,j]:=0;
                  END;
            end;
```

```
     end;

  PROCEDURE CAMBIO;
    begin
     vconm(10,6,65,8,cyan,negro,cyan,negro);
     gotoxy(10,2);write('Con Cual Drive Deseas Trabajar (A/B/C):   ');
     readln(dr);
     dr:= concat(dr,':');
     {$I-}
     chdir(dr);
     {$I+}
     if ioresult <> 0 then
        begin
          vconm(10,6,65,8,cyan,negro,cyan,negro);
          gotoxy(10,2);write('Drive  Invalido');
          writexy(1,1,readkey);
          cerrar_ventana;
         end;
      cerrar_ventana;
    end;

  PROCEDURE GRABRAR; {<1> creacion del archivo}
     begin
       clrscr;
       vconm(10,6,65,8,cyan,negro,cyan,negro);
       gotoxy(10,2);write('  Indique el Nombre del Archivo  :  ');
       gotoxy(50,2);readln(vx);
       vx:=concat(dr,vx,'.dat');
       if existe(vx) then
          begin
            gotoxy(50,2);clreol;
            write('EL ARCHIVO YA EXISTE, SE REGRABA S/N : ');
            if upcase(readkey)='N' then
              begin
               cerrar_ventana;
               exit;
              end;
          end;
       cerrar_ventana;
       clrscr;
       assign(archivo,vx);
       rewrite(archivo);
       with registro do
         begin
           vconm(2,2,74,20,cyan,negro,cyan,negro);
           gotoxy(16,8);write('         Cuantas Variables son : [      ]');
           repeat
           numv:=ROUND(VENTERO(49,8,3,1,MAX_VAR,VALIDO,CH));
           until CH=ENTER;
           gotoxy(16,10);write('   Cuantas Restricciones son : [      ]');
           repeat
           numo:=ROUND(VENTERO(49,10,3,1,MAX_VAR,VALIDO,CH));
           until CH=ENTER;
           cerrar_ventana;
           for i:= 1 to numv do g[i]:=0;
           clrscr;
           for i:=1 to numv do
             begin
               str(i,v5);
```

```
    f[i]:='x'+ v5;
  end;
f[i+1]:='B';
for j :=1 to numo do
  begin
   str(j,v6);
   k[j] := concat('y',v6);
  end;
k[j+1]:=' z';
seek(archivo,0);
write(archivo,registro);
h:=0;
posx:=4;
posy:=3;
textbackground(0);
clrscr;
vconm(2,2,76,23,cyan,negro,cyan,negro);
clrscr;
write('C A P T U R A  D E  D A T O S ');
for j:=1 to numv+1 do
  begin
    gotoxy(posx+(j*6),posy);writeln ('X',j);
    if j >= numv+1 then
      begin
        gotoxy(posx+(j*6),posy);write('<,=,>');
      end;
  end;
gotoxy(posx+((numv+2)*6),posy);write('B     ');
posy:=posy+1;
for j:= 1 to numo  do
  begin
    h:=h+1;
    gotoxy(posx,posy);write('Y',j);
    for i:=1 to numv do
      begin
       gotoxy(posx+(i*6),posy);
       CH:='A';
       repeat
       g[i]:=vreal(posx+(i*6),posy,5,1,MAX_VAR,VALIDO,CH);
       until CH=ENTER;
       gd[i,j]:=g[i];
      end;
    gotoxy(posx+((numv+1)*6),posy);
    repeat
    cor:=leechar(posx+((numv+1)*6),posy,1,1,max_var,VALIDO,CH);
    until (CH=ENTER ) and (cor in signo);
    sig[j] :=cor;
    CH:='A';
    repeat
    g[numv+1]:=vreal(posx+((numv+2)*6),posy,5,1,MAX_VAR,VALIDO,CH);
    until CH=ENTER;
    posy:=posy+1;
    seek(archivo,h);
    write(archivo,registro);
  end;
gotoxy(posx,posy);write('Z  ');
for i:=1 to numv do
  begin
    gotoxy(posx+(i*6),posy);
```

```
            CH:='A';
            repeat
            g[i]:=vreal(posx+(i*6),posy,5,1,MAX_VAR,VALIDO,CH);
            until CH=ENTER;
          end;
        g[numv+1]:=0;
        seek(archivo,h+1);
        write(archivo,registro);
      end;
    close(archivo);
    cerrar_ventana;
    recu:=true;
   end;

PROCEDURE LEER;
 var
  cad1,cad2 :string[80];
  convr : string;
  begin
    clrscr;
    with registro do
      begin
        clrscr;
        if una =0 then
          begin
            if destino=2 then
              begin
               writeln(lst,'Elarchivo"',vx,'"contiene:',rbl,'variables
               independientes');
               writeln(lst,'',obs,'    restricciones');
               writeln(lst);
               una:=1;
              end
            else
              begin
                writeln('El  archivo  "',vx,'"  contiene  :  ',varbl,'
                variables independientes');
                writeln(' ',obs,'restricciones');
                writeln;
              end;
          end;
        cad1:=' ';
        cad2:=' ';
        gotoxy(5,4);
        for i:=1 to varbl  do
          begin
           cad1:=cad1+'         '+f[i];
           write(f[i]:8);
          end;
        if destino=2 then
          begin
           cad1:=cad1+' signo    B      ';
           writeln(lst,cad1);
          end
        else
          begin
           write('  signo');
           write('  B   ');
          end;
```

```
        for j:=1 to obs + 1 do
          begin
            gotoxy(1,4+j); write(k[j]);
          end;
        for j:=1 to obs +1 do
          begin
            gotoxy(6,4+j);
            cad2:=cad2 +k[j];
            for i:=1 to varbl  do
               begin
                 write(gd[i,j]:8:2);
                 str(gd[i,j]:8:2,convr);
                 cad2:=cad2 + convr;
               end;
            { SIG[OBS+1]:=' ';
            SIG[OBS+2]:=' ';}
            write(sig[j]:4);
            cad2:= cad2 + '    '+sig[j];
            write(gd[varbl+1,j]:8:2);
            str(gd[varbl +1,j]:8:2,convr);
            if j=obs+1 then cad2 := cad2 +' '+ convr
            else cad2 := cad2 + convr;
            if destino =2 then writeln(lst,cad2);
            cad2 :=' ';
          end;
      end;
    if destino =2 then
      begin
        writeln(lst,'_________________________________');
        writeln(lst);
      end;
    writexy(13,15,'Oprime Cualquier Tecla Para Comenzar');
    writexy(1,1,readkey);
  end;

PROCEDURE RECUPERA;
  begin
    assign(archivo,vx );
    reset(archivo);
    with registro do
      begin
        while not eof(archivo) do
          begin
            h:=0;
            seek(archivo,h);
            read(archivo,registro);
            for j :=1 to numo +1 do
              begin
                h:=h+1;
                seek(archivo,h);
                read(archivo,registro);
                for i :=1 to numv +1 do
                  begin
                    if ww = '7' then
                    gd[j,i] := g[i]
                    else
                    gd[i,j] := g[i];
                  end;
                sgn[j] := sig[j];
```

```
            end;
          end;
        close(archivo);
        obs :=numo;
        varbl := numv;
      end;
    end;   { fin de recupera }

PROCEDURE RECUPERAR;
  begin
    clrscr;
    vconm(10,6,65,8,cyan,negro,cyan,negro);
    gotoxy(10,2);write('Nombre del Archivo a Recuperar : ');
    readln(vx);
    vx :=concat(dr,vx,'.dat');
    if not existe(vx) then
      begin
        clrscr;
        vconm(10,6,65,8,cyan,negro,cyan,negro);
        gotoxy(10,2);write('    El  Nombre  del  Archivo  No  Existe ');
        repeat  until keypressed;
        cerrar_ventana;
        clrscr;
        exit;
      end;
    cerrar_ventana;
    recupera;
    recu:=true;
    if destino = 1 then ;
  end;    { fin de recuperar; }

PROCEDURE MODIFICARAR; {<1> modificaci"n del archivo}
 var
  aux  : real;
  auxc : char;
  begin
    clrscr;
    if recu=false then
      begin
        clrscr;
        vconm(10,6,65,8,cyan,negro,cyan,negro);
        gotoxy(10,2);write('    No existen datos para modificar  ');
        repeat  until keypressed;
        cerrar_ventana;
        clrscr;
        exit;
      end;
    inicia;
    recupera;
    assign(archivo,vx);
    rewrite(archivo);
    with registro do
      begin
        seek(archivo,0);
        write(archivo,registro);
        h:=0;
        posx:=4;
        posy:=3;
        textbackground(0);
```

```
   clrscr;
   vconm(2,2,76,23,cyan,negro,cyan,negro);
   clrscr;
   write('M O D I F I C A C I O N    D E     D A T O S ');
   for j:=1 to numv+1 do
     begin
       gotoxy(posx+(j*6),posy);writeln (' X',j);
       if j >= numv+1 then
         begin
           gotoxy(posx+(j*6),posy);write('<,=,> ');
         end;
     end;
   gotoxy(posx+((numv+2)*6),posy);write(' B     ');
   posy:=posy+1;
   for j:= 1 to numo  do
     begin
       h:=h+1;
       gotoxy(posx,posy);write('Y',j);
       for i:=1 to numv do
         begin
           gotoxy(posx+(i*6),posy); { write(gd[i,j]:8:2); }
           CH:='A';
           repeat
           aux:=vreal(posx+(i*6),posy,5,1,MAX_VAR,VALIDO,CH);
           until CH=ENTER;
           if aux <> infinito then g[i]:=aux;
           ClrEol;
         end;
       gotoxy(posx+((numv+1)*6),posy); { write(sig[j]);  }
       auxc:=sig[j];
       repeat
       auxc:=leechar(posx+((numv+1)*6),posy,1,1,max_var,VALIDO,CH);
       until CH=ENTER;
       sig[j] :=auxc;
       gotoxy(posx+((numv+2)*6),posy); { write(gd[numv+1,j]:8:2); }
       CH:='A';
       repeat
       aux
       A=vreal(posx+((numv+2)*6),posy,5,1,MAX_VAR,VALIDO,CH);
       until CH=ENTER;
       if aux <> infinito then g[numv+1]:=aux;
       posy:=posy+1;
       seek(archivo,h);
       write(archivo,registro);
     end;
   gotoxy(posx,posy);write('Z  ');
   for i:=1 to numv do
     begin
       gotoxy(posx+(i*6),posy); {write(gd[i,j+1]:8:2);}
       CH:='A';
       repeat
      aux :=vreal(posx+(i*6),posy,5,1,MAX_VAR,VALIDO,CH);
      until CH=ENTER;
      if aux <> infinito then g[i]:=aux;
    end;
  g[numv+1]:=0;
  seek(archivo,h+1);
  write(archivo,registro);
end;
```

```
BEGIN
  VALIDO:=[ENTER,ESC];
  TOMA:=1;
  REPEAT
    repeat
    toma:=MENU(TOMA,1,1,3,'H',OPCIONES,PIE,ch);
    if ch=esc then halt;
    until ch=enter;
    case toma of
      1:begin
          TOMA2:=1;
          repeat
          toma2:=menu(toma2,2,2,5,'V',opcis,page,ch);
          until ch=enter;
          cerrar_ventana;
          case toma2 of
          1: cambio;
          2: grabar;
          3: recuperar;
          4: Modificar;
        end;
      end;
      2:begin
          TOMA3:=1;
            repeat
            toma3:=menu(toma3,22,2,3,'V',opcisa,pagina,ch);
            until ch=enter;
            cerrar_ventana;
            case toma3 of
              1: begin
                   destino:=1;
                   simplex;
                 end;
               2: begin
                    destino:=2;
                    una:=0;
                    simplex;
                  end;
                end;
             end;
                3:begin
                    cerrar_ventana;
                    halt;
                  end;
          end;
          until false;
END;
```

PROGRAMA PRINCIPAL

```
  begin
    dr:='c';
    s:=0;
    r:=0;
    una:=1;
    recu:=false;
    clrscr;
    VALIDO:=[ENTER,ESC];
    textbackground(7);
```

```
    close(archivo);
    cerrar_ventana;
    if destino = 1 then ;
    recu:=true;
  end;

PROCEDURE INFATIBLE;
  begin
    clrscr;
    if destino = 2 then writeln(lst,' SOLUCION INFACTIBLE ')
    else gotoxy(12,12); write(' SOLUCION INFACTIBLE ');
    writexy(13,15,'Oprime Cualquier Tecla Para Comenzar');
    writexy(1,1,readkey);
  end;

PROCEDURE COLUMNA;
  begin
    min:=0;
    for i:=1 to varbl do
      begin
        if gd[i, obs+1] < min then
          begin
            min:= gd[i,obs+1] ;
            r:=i;
          end;
        end;
  end;

PROCEDURE HILERA;
  begin
    minh:=1.0e+29;
    for j:=1 to obs do
      begin
        if gd[r,j] > 0 then
          begin
            if minh >   gd[varbl+1,j]/gd[r,j]  then
              begin
                minh:=  gd[varbl+1,j]/gd[r,j];
                s:=j;
              end;
            end;
          end;
      writeln;
      writeln(r:6,s:6);
  end;

PROCEDURE INTERCAMBIO;
  begin
    for j:=1 to obs +1 do
      begin
        if j <> s then
          begin
            for i:=1 to varbl +1 do
              begin
                if i <> r then
                  begin
                    gd[i,j] := gd[i,j] - ((gd[r,j] * gd[i,s]) / gd[r,s]);
                  end;
                end;
```

```
          end;
        end;
      for i:=1 to varbl +1 do
        begin
          if i <> r then gd[i,s]:= gd[i,s]/gd[r,s];
        end;
      for j:=1 to obs +1 do
        begin
          if j <> s then gd[r,j]:=( gd[r,j]/gd[r,s])*(-1);
        end;
      gd[r,s] := 1/gd[r,s];
      with registro do
        begin
          entra:= f[r];
          sale := k[s];
          k[s] := entra;
          f[r] := sale;
        end;
  end;

PROCEDURE FASE1;
  begin
    clrscr;
    gotoxy(33,11);WRITELN('FASE 1');
    for j:=1 to obs +1 do
      begin
        gd[varbl +2,j] := gd[varbl +1,j];
        if gd[varbl +2,j ] < 0 then gd[varbl +1,j] := -1
        else gd[varbl +1,j]:= 0;
      end;
    for i:=1 to varbl +2 do
      begin
        gd[i,obs +2] :=0;
        if i= varbl +1 then  gd[i,obs+2] :=1;
      end;
    with registro do
      begin
        k[numv+2] := 'mu';
        r:= varbl +1;
        minh:=-1.0e-29;
        for j:=1 to obs do
          begin
            if gd[varbl+2,j] < 0 then
              begin
                if minh > gd[varbl+2,j] then
                  begin
                    minh := gd[varbl+2,j];
                    s:=j;
                  end;
               end;
            end;
          f[numv+1] := 'ro';
          varbl := varbl +1;
          obs := obs +1;
          intercambio;
          writeln;
          gotoxy(31,13);writeln(' ENTRA "ro"  ');
          writexy(13,15,'Oprime Cualquier Tecla Para Comenzar');
          writexy(1,1,readkey);
```

```
            leer;
            repeat
            columna;
            obs :=obs-1;
            hilera;
            obs:=obs +1;
            if gd[r,s] =0 then
              begin
                infactible;
                jj:=varbl;
                end
                else
                  begin
                    intercambio;
                    writeln;
                    writeln(' "ro" CONTINUA EN LA BASE ');
                    leer;
                    jj:=varbl;
                    for i:=1 to varbl +1 do
                      begin
                        if gd[i,obs +1] < 0 then jj:=0;
                      end;
                    end;
                  until jj = varbl ;
                  varbl := varbl -1;
                  obs := obs -1;
                  for i:=1 to varbl +2 do
                    begin
                      if gd[i,obs +2] = 1 then x:= i;
                    end;
                  for i:=x to obs+1  do f[i]:= f[i+1];
                end;
              for j:=1 to obs+1 do
                begin
                  for i:=x to varbl +2 do
                    begin
                      gd[i,j] :=gd[i+1,j];
                      gd[i,obs+2] := 0;
                    end;
                end;
              writeln;
              writeln('SE ELIMINA "ro" ');
              leer;
  end;

PROCEDURE SIMPLEX;
 VAR
  PRESICION:REAL;
  begin
    PRESICION:= -0.9;
    if recu=false then
      begin
        clrscr;
        vconm(10,6,65,8,cyan,negro,cyan,negro);
        gotoxy(10,2);write('     No existen datos para procesar   ');
        repeat  until keypressed;
        cerrar_ventana;
        clrscr;
        exit;
```

```
  end;
cont :=0;
clrscr;
inicia;
recupera;
vconm(2,2,76,23,cyan,negro,cyan,negro);
leer;
opt:='0';
clrscr;
while ((opt <> 'max') and (opt <> 'MAX')) and ((opt <> 'min') and
(opt <> 'MIN')) do
  begin
    gotoxy(12,12); write('TECLEE : [MAX]IMIZAR / [MIN]IMIZAR ==> ');
    readln(opt);
  end;
if (opt = 'max') or (opt = 'MAX') then
  begin
    for i:=1 to varbl do  gd[i,obs +1]:= (gd[i,obs+1])*-1 ;
  end;
with registro do
   begin
     for j:=1 to obs do
       begin
        if sig[j] = '>' then
          begin
            sig[j]:= '<';
            for i:=1 to varbl +1 do gd[i,j]:=gd[i,j]*-1;
          end;
        end;
       end;
     for j:= 1 to obs do
       begin
        if gd[varbl+1,j ] < 0  then
          begin
            fasel;
            j :=obs;
          end;
        end;
       repeat
       columna;
        hilera;
        if gd[r,s] = 0 then
          begin
            infactible;
            jj :=varbl;
            end
          else
            begin
              intercambio;
              leer;
              jj:=0;
              for i:=1 to varbl do
                begin
                  if gd[i,obs+1] >= PRESICION then   jj:=jj+1;
                end;
              if jj= varbl then
                begin
                  if destino =2 then
                    begin
```

```
                    writeln(lst,' TABLEAU OPTIMO');
                    w.iteln(lst);
                    end
                  else writexy(13,15,' TABLEAU OPTIMO');
                  writexy(1,1,readkey);
                end;
              end;
            until jj = varbl ;
            cerrar_ventana;
  end;

PROCEDURE HOLA;
  begin
    writexy(17,1,'P   A   Q   U   E   T   E ');
    writexy(27,3,'D   E  ');
    writexy(10,5,'P R O G R A M A C I O N        L I N E A L');
    writexy(25,11,'Elaborado por :');
    writexy(20,12,'ALEJANDRO REYES GUERRERO');
    writexy(20,13,'JUAN MANUEL MUZOZ ARAUJO');
    writexy(13,14,'Oprime Cualquier Tecla Para Comenzar');
    writexy(1,1,readkey);
    cerrar_ventana;
    clrscr;
  end;

PROCEDURE PRESENTA;
  begin
    clrscr;
    textbackground(7);
    vconm(4,2,74,22,cyan,negro,cyan,negro);
    writexy(16,1,'    C O N S I D E R A C I O N E S');
    writexy(10,4,'El programa utiliza el metodo simplex para obtener');
    writexy(18,5,'   la  soluci"n del problema                    ');
    writexy(12,7,' El problema se resuelve en su forma estandar ');
    writexy(22,9,' Maximizar  z = cx ');
    writexy(22,11,' Sujeta a  Ax <= b ');
    writexy(22,13,'               x >= 0 ');
    writexy(4,15,'El programa  No  acepta mas de 10 variables y 10
    restricciones ');
    writexy(17,18,'Oprime Cualquier Tecla Para Comenzar');
    writexy(1,1,readkey);
    cerrar_ventana;
  end;

PROCEDURE ORQUESTA;
const
  OPCIONES:TIPOOPCION=('ARCHIVO','PROCESAMIENTO','
  TERMINAR','','','','','','','');
  PIE:TIPOOPCION=('Manejo de archivos','Encontrar las soluciones','Indicar
  el dispositivo de salida','Salir del programa y regresar a DOS','',
  '','','','','');
  OPCIS:TIPOOPCION=('DRIVE','CREAR','RECUPERAR','MODIFICAR',
  'MENU','','','','','');
  PAGE:TIPOOPCION=('Le permite cambiar de Unidad de Disco','Crea
  un archivo de trabajo','Recupera un archivo del disco','Modifica un
  Archivo ','Regresa al Menu Principal','','','','','');
  opcisa:TIPOOPCION=('PANTALLA','IMPRESORA','MENU','','','','','','','');
  PAGINA:TIPOOPCION=('Le permite procesar los datos del problema','Le
  permite imprimir los datos','Regresa al Menu Principal','','','','');
```

```
    vconm(8,4,70,20,cyan,negro,cyan,negro);
    hola;
    presenta;
    clrscr;
    cerrar_ventana;
    inicia;
    orquesta;
  end.
```

En el programa fuente se utilizo algunas utilerias del lenguaje Pascal, para realizar las ventanas del Menu Principal y de la captura de datos.

Los Procedimientos más importantes para capturtar la información son :

-) INICIA
-) GRABAR
-) LEER
-) COLUMNA
-) HILERA

Para recuperar y modificar los datos en los archivos creados fue necesario crear los siguientes Procedimientos :

-) RECUPERA
-) RECUPERAR
-) MODIFICAR

Una vez que se tenian los datos capturados los siguientes procedimientos que se realizaron fueron los de procesamiento de datos, es decir utilizando el Método de Intercambio de Jordan para obtener los valores de Z, así que los procedimientos que contienen esta información son:

-) COLUMNA
-) HILERA
-) INTERCAMBIO
-) FASE1
-) SIMPLEX

Los procedimientos HOLA y PRESENTA es donde vienen las dos portadas de entrada que presenta el paquete "PROLIN".

El procedimiento ORQUESTA es donde se le dice al programa donde debe buscar toda la información del Paquete, así como tambien mostrar el Menu Principal del mismo.

Y las ultimas instrucciones donde dice PROGRAMA PRINCIPAL, es donde se le indica al programa que ejecute todas las indicaciones que se presentan en dicho paquete.

En el presente trabajo se proporcionan los siguientes diskettes ;

1) Diskette del Paquete "PROLIN"

2) Diskette de Trabajo, donde se encuentran los ejercicios de la Tesis.

III MANEJO DEL PAQUETE DE PROGRAMACION LINEAL PROLIN

3.1 Consideraciones del programa.

PROLIN es un paquete que le permite resolver problemas de Programación Lineal, en computadoras PC con memoria RAM de 512 Kb o más, pudiendo realizar las siguientes acciones:

a) Editar rápidamente la formulación de un problema de programación lineal.

b) Resolver en segundos la formulación de un problema, presentando un reporte de resultados.

c) Aceptar modificaciones en la formulación y presentar de inmediato los nuevos resultados.

d) Podemos visualizar los resultados tanto en la pantalla como en papel.

El menú que presenta PROLIN está diseñado para monitores RGB en color y monocromáticos CGA, presentando ligeras variaciones en monitores de tipo Hárcules o TTL.

Usted puede utilizar PROLIN desde máquinas con una sola unidad de disco flexible hasta equipos provistos de disco fijo.

3.2. Instalación y Trabajar con PROLIN.

1) En una Computadora de Dos Floppys

Cuando se trabaja con una Computadora que carece de Disco Duro, el disco del Programa PROLIN tiene incorporado el arrancador del MS-DOS Versión 4.0, permitiendole trabajar con la computadora utilizando un solo disco.

3.2.1. PC'S de Dos Floppys.

Para entrar al sistema deben tomarse en consideración las siguientes instrucciones :

i) Inserte el disco de PROLIN en el drive "A" estando apagado el equipo.

ii) Encienda el equipo (CPU, Monitor).

Después de lo cual aparecerá la portada y leyenda de derechos reservados del PROLIN.

3.2.2. PC'S de Disco Duro.

Si contamos con una computadora de disco duro lo primero que se debe de hacer es instalar el programa en un subdirectorio del disco duro, para lo cual se deben seguir los siguientes pasos:

- Estando en el disco duro se crea un subdirectorio

C>MD PROLIN

- Colocamos el programa en la unidad de trabajo y se procede a copiar los siguientes archivos :

C>COPY A:PROLIN.EXE C:\PROLIN

Una vez copiado el archivo fuente el siguiente paso es entrar al subdirectorio antes mencionado:

C>CD PROLIN

y proceda a teclear lo siguiente:

C>PROLIN

Después el monitor desplegará la portada de PROLIN y los derechos reservados del mismo.

Una vez que se despliega la portada de los derechos del Programa, se deberá oprimir cualquier tecla para continuar, lo cual nos lleva a otra ventana que indica las consideraciones que deben tomarse en cuenta para trabajar dicho programa. Para continuar oprima cualquier tecla, donde aparecerá nuestro MENU PRINCIPAL.

3.3. Manejo del Menú Principal.

El Menú Principal ofrece ayuda permanente al usuario mediante una linea de texto ubicada en la parte baja de la pantalla, la cual proporciona información sobre el comando que se esta utilizando en ese momento.

PROLIN está diseñado para guiar al usuario, iluminando automáticamente los comandos en la secuencia correcta de trabajo.

El Menú Principal consta de los siguientes comandos, que nos van a permiter trabajar con el Programa:

ARCHIVO	PROCESAMIENTO	TERMINAR

Para invocar un comando del menú, los pasos a seguir son: 1) Posicionar el cursor en el comando deseado utilizando las flechas.

2) Una vez ubicado el cursor oprima <Enter>. Si la opción elegida es ARCHIVO aparecerá el siguiente Submenú:

DRIVE	CREAR	RECUPERAR	MODIFICAR	MENU

Pero si el comando elegido es PROCESAMIENTO entonces aparecerá el Submenú:

PANTALLA	IMPRESORA	MENU

Cada uno de los submenús contiene como último comando MENU, el cual al ser invocado lo lleva de regreso al menú principal.

Para salir del Programa PROLIN elija el comando TERMINAR, el cual nos llevará al Sistema Operativo.

Utilización del Comando ARCHIVO.

En el Menú Principal encontrará que el comando ARCHIVO está iluminado así que oprima <Enter> para entrar al submenú ARCHIVO.

Estando dentro del submenú ARCHIVO notará que esta compuesto por los siguientes comandos :

DRIVE	CREAR	RECUPERAR	MODIFICAR	MENU

Cuando estamos dentro del comando ARCHIVO notará que el primer comando DRIVE está iluminado . Con el Programa se puede almacenar la información ya sea en Disco Flexible o bien en Disco Duro, para lo cual donde se desea trabajar se oprime el Comando DRIVE, con lo cual aparece la siguiente pregunta :

Con Cual drive Deseas Trabajar (A/B/C):

Si se desea trabajar en cualquiera de las unidades de disco nada más basta indicarle el Drive de la siguiente forma: Con Cual drive Deseas Trabajar(A/B/C):B [ENTER]

Nota : Apesar de que en el disco del paquete existe espacio, sugerimos que el usuario almacene sus datos en otro disco (disco de trabajo o bien en disco duro).

Una vez definida la unidad donde reside el disco de trabajo, utilice el comando CREAR para lo cual con la flecha abajo ilumine el comando y oprima <Enter> ante lo cual aparece la pregunta:

Indique el Nombre del archivo :

Teclee el nombre del archivo donde se almacenarán los datos considerando las siguientes reglas :

1) El tamaño máximo del archivo son 8 caracteres.
2) No debe contener espacios, ni signos de puntuación.

Cuando haya terminado de teclear el nombre presione <Enter>. Después de definir el nombre del archivo nos pregunta :

Cuantas Variables son : [ ]

se teclea el número de variables que contiene el Problema y se da [ENTER] para continuar, apareciendo luego

Cuantas Restricciones son : [ ]

tecleamos el número de restricciones y damos <Enter> y en seguida aparece la pantalla de Captura de Datos.

CAPTURA DE DATOS

	X1	X2	X3	X4	XN	<=>	B
Y1							
Y2							
YN							
Z							

La forma de introducir la información debe ser compatible con la forma en que el usuario previamente planteó su problema y como desea que éste quede en el Tableau de Tycker. Los primeros datos que debe introducir son la primera restricción, con su respectiva desigualdad, así hasta terminar de vaciar toda su información. Tomando en cuenta que cada vez que se teclea un número se debe de dar un <Enter> así hasta terminar de introducir todo el problema. Una vez capturados todos los datos, nuevamente aparece el MENU PRINCIPAL.

Cabe señalar que al momento de llegar al MENU PRINCIPAL los datos introducidos al sistema han sido grabados al disco en el archivo creado y en la unidad de trabajo especificada.

En el caso que se desee recuperar un archivo ya existente entonces elegimos dentro del Submenú ARCHIVO el comando RECUPERAR el cual nos hace la siguiente pregunta:

Cual es el nombre del archivo :

Por lo cual se proporciona el nombre del archivo y se da <ENTER>. Y el programa lo va a recuperar en la memoria RAM de la computadora. Pero en el caso que se dé un nombre de archivo erróneo se exhibirá en la pantalla el siguiente mensaje :

El Nombre del Archivo No Existe

Oprimimos <ENTER> y nos regresa al Submenú ARCHIVO, para verificar si existe el nombre del archivo deberá hacer lo siguiente:

1.- Que la unidad de disco de trabajo esté correctamente definida.

2.- Desde sistema operativo consultar el directorio de su disco de trabajo, y anotar el nombre correcto.

Si desea modificar la información contenida de un problema, coloque el cursor en el Submenú ARCHIVO y posicione el cursor en el Comando MODIFICAR y a continuación aparecerá el Tableau con los datos para poderlos modificar.

Para modificar cualquiera de los coeficientes o desigualdad posicione el cursor con la tecla ENTER en el dato a corregir y teclee el coeficiente o signo correcto. Una vez modificada la tabla en la pantalla aparece el MENU PRINCIPAL.

Utilización del Comando PROCESAMIENTO

Una vez planteado el problema, el siguiente paso a seguir es el procesamiento de los datos, para lo cual utilizamos el comando PROCESAMIENTO, que al ser invocado nos aparece el siguiente Submenú:

PANTALLA	IMPRESORA	MENU

este Submenú nos permite mandar la salida de resultados ya sea en la pantalla o bien a la impresora.

Entonces, cuando queremos mandar a procesar algún problema ya sea en pantalla o en impresora, tecleamos la alternativa que se desea y a continuación aparece el tableau de datos y que teclemos <Enter> para continuar. Enseguida nos pide que si deseamos :

[MAX]imizar [MIN]imizar

Realizando lo que se desea optimizar en la función objetivo. Considerando que la salida de resultados se muestra Tableau por Tableau hasta llegar a la última tabla, donde nos arroja los siguientes resultados:

1) Los valores de las variables seleccionadas.
2) Los valores de los costos de oportunidad de las variables seleccionadas y no seleccionadas.
3) Los valores de los recursos utilizados.
4) Los valores de los costos de oportunidad de los recursos.
5) El valor de la función objetivo.

La interpretación económica de los resultados que proporciona el programa varía en función del tipo de problema.

3.4. Limitaciones del Paquete.

Este programa tiene algunas limitantes como son las siguientes :

1) Sólo se puede trabajar este paquete en una computadora PC, que tenga una Memoria RAM de 512 Kb ó más. Y tenga un Monitor CGA para que no presente problemas de configuración.

2) Sólo puede trabajar con 10 variables y 10 Restricciones.

3) No presenta la salida de Análisis de Sensibilidad como en otros paquetes.

4) Solamente está diseñado como una herramienta de autoaprendizaje.

5) El Programa utiliza el Método DUAL-SIMPLEX en su Forma Estándar, y por lo tanto no se considera el Método Gráfico.

6) No permite realizar gráficas.

7) En monitores de tipo Hércules o TTL, presenta problemas de visualización.

8) No muestra ninguna interpretación económica en la salida de resultados; dado que cada planteamiento del problema posee una interpretación de acuerdo a su objetivo.

9) Para fines lucrativos, de este paquete todos los derechos son cedidos a la Facultad de Economía de la Universidad Nacional Autónoma de México.

10) Sólo puede ser utilizado por usuarios que tengan conocimiento de Programación Lineal e Investigación de Operaciones.

IV MANUALES DEL MANEJO DE OTROS PAQUETES DE PROGRAMACION LINEAL

El presente capítulo tiene la finalidad de proporcionar al estudiante una idea de conjunto de lo se requiere para poner en operación los diferentes sistemas en su computadora. El usuario tendrá una guía completa de instalación rápida y además de proporcionar de manera muy superficial la descripción de los comandos más importantes de cada paquete. Cabe señalar que dichos paquetes están orientados a resolver problemas de Programación Lineal, los cuales son los siguientes: LINDO Y QSB.

Es importante hacer hincapié que dichos paquetes no son los únicos que permiten dar solución a problemas de donde existen problemas de asignación de recursos limitados, pero por sus características nos permiten establecer una descripción de ellos de una manera fácil y sencilla de entender.

4.1. Manual del Paquete LINDO.

LINDO (Linear Interactive Discrete Optimizer) es un paquete de cómputo diseñado para resolver problemas de programación matemática: Lineal, Entera y Cuadrática. En este trabajo sólo se considera lo correspondiente a Programación Lineal.

dentro del diskette de su programa se pueden guardar archivos, es decir, el diskette del paquete LINDO tiene espacio suficiente para guardar archivos.

En el presente capítulo se hace referencia a la versión de LINDO que permite incluir 119 Variables y 59 Restricciones para un Modelo.

Pasos a seguir para entrar al programa LINDO.

Para realizar una sesión de trabajo en LINDO se requieren los siguientes Diskettes:

1) Diskette del Sistema Operativo.

2) Diskette del programa LINDO.

Teniendo estos diskettes se realizan los siguientes pasos para entrar a LINDO:

1).- Se introduce el diskette del MSDOS en el drive A y se prende la Computadora.

2).- Una vez que se ha cargado el MS-DOS a la PC y aparezca en el monitor A>, sacamos el diskette del MS-DOS y colocamos el diskette del programa LINDO.

3).- Una vez dentro el diskette del programa LINDO ponemos:

A >LINDO $\boxed{\leftarrow}$

Esperamos un momento mientras que el programa se cargue a la memoria RAM. Cuando aparezca en el extremo izquierdo de la pantalla la indicación [:] significa que ya se está dentro del programa LINDO.

Como editar en LINDO.

Este programa permite editar la formulación de un problema de programación lineal de la misma forma a como uno lo haría en una hoja de papel.

En la edición de un problema de programación lineal en LINDO se tiene la siguiente secuencia:

1.- Se teclea "MAX" o "MIN" (segun sea el caso) para indicar que se inicia la edición de un nuevo problema.

2.- Se continua poniendo los coefiecientes y las variables de la Función Objetivo. Se indicará que ya se ha términado de editar la función objetivo hasta que se teclee la frase "Subject to ⟵┘. Después de ésto aparecerá en la pantalla un signo de "?", indicando que se proceda a escribir la primera o siguiente restricción.

NOTAS:

i) En caso de tener una función objetivo grande se puede utilizar más de un renglón.

ii) Al estar editando se puede o no dejar un espacio entre el signo (+ o -) y el coeficiente.

iii) El nombre de las variables puede contener hasta ocho caracteres y debe empezar con una letra.

3.- Se pretende a editar las restricciones. Una restricción esta compuesta por tres partes:

a) El lado izquierdo de la desigualdad.

b) La dirección (DIR) de la desigualdad.

c) El lado derecho de la desigualdad (RHS).

Al igual que en la función objetivo se pueden editar restricciones en uno o más renglones.

4.- Para indicar que ya se ha términado de editar el problema, hay que teclear la frase "END" . Entonces aparecerá en la pantalla de nuevo la indicación ":", señalando que se está listo para el siguiente comando.

Ejemplo de edición:

Supongase que se desea editar el siguiente problema:

MAX $Z = 14M1 + 23M2 + 13M3 + 12M4$

ST $3M1 + 7M2 + 4M3 + 2M4 \leq 50$

$5M1 + 6M2 + M3 + 9M4 \leq 55$

Entonces en el programa LINDO se teclearia así:

```
: MAX  14M1 + 23M2 + 13M3 + 12M4    [←┘]
? ST   3M1 + 7M2 + 4M3 + 2M4 ≤ 50   [←┘]
?      5M1 + 6M2 + M3 + 9M4 ≤ 55    [←┘]
? END  [←┘]
```

Comandos de mayor uso.

Una vez que se ha editado la formulación de un problema, se pueden presentar diversas necesidades como:

a) Correcciones en la Formulación.

b) Solicitud de la Solución.

c) Sensibilidad de la Solución.

d) Planteamiento del Problema en Forma Matricial.

e) Impresión de los Resultados.

Para atender estas necesidades se requieren del manejo de algunos comandos.

El programa está compuesto por 38 comandos específicos, los cuales se agrupan en 11 categorías[1]. Para observar esta estructura es necesario teclear:

: COM [⟵]

Para la formulación, solución y simulación de un Problema de Programación, en realidad sólo se requiere del manejo de 14 comandos, cuya función general es:

1) MAX o MIN Para iniciar la edición de un problema.

2) LOOK Para desplegar la formulación en forma algebráica

3) PIC Para desplegar la formulación en forma matricial.

4) ALTER Para realizar modificaciones a la formulación de un problema.

5) EXT Para aumentar restricciones.

6) DEL Para borrar restricciones.

7) GO Para ejecutar la solución de un problema.

8) SOLUTION Para despejar la Solución Básica.

9) RANGE Para desplegar el Análisis de Sensibilidad

10) APPC Para introducir variables por columnas.

11) SHOC Para desplegar la formulación por columna.

12) SAVE Para guardar archivos.

[1]Ver anexo A.

13) RETR Para mandar a recuperar un archivo.

14) QUIT Para salirse del programa LINDO.

En el tratamiento de problemas de programación lineal con LINDO se recomienda tener siempre presente lo siguiente:

i) En la estructura de la formulación de un problema a la función objetivo es considerada como la primera restricción, por lo tanto la primera restricción del problema real, será la segunda restricción en el formato del programa y así sucesivamente.

ii) Las partes o componentes de una restricción son:

a) ROW NUMBER (número de restricción)

b) COEFICIENT (coeficientes)

c) VAR (variables)

d) DIR (dirección de la desigualdad)

e) RHS (lado derecho de la desigualdad).

Ejemplo:

ROW NUMBER	COEFICIENT		VAR	DIR	RHS
↓	↓		↓	↓	↓
3)	30 SORGO	+ 25	MAIZ	≤	500

iii) En el uso del comando ALTER se debe especificar:

a) Número de la restricción.

b) Tipo de cambio (VAR,DIR,RHS)

c) Nuevo coeficiente o dirección.

Estas tres partes se pueden indicar juntas (al mismo tiempo) o por partes (una por una).

iv) Cada vez que se realice una modificación se pierde la solución del problema anterior, a menos que se haya guardado antes de realizar la modificación.

Comandos complementarios, pertenencientes a MS-DOS, que se utilizan en el uso del programa LINDO:

1) DIR Sirve para enlistar los archivos guardados en el diskette de LINDO. Se pide desde sistema operativo.

2) [Crtrl] [S] Para detener el despliegue de la pantalla. Este es muy usual en la formulación de problemas grandes, en donde una pantalla no alcanza para desplegar todo el problema y/o la solución del mismo.

3) [Crtl] [Print] Para activar la impresora y poder imprimir la ejecución de los comandos que se vayan solicitando.

Para mejores referencias del paquete LINDO,consultar con el manual del paquete.

4.2. Manual del Paquete QSB.

El paquete QSB (Quantitative Systems for Business) es un nuevo paquete que se ha especializado en solucionar los problemas relacionados con la optimización restringida, especialmente la Programación Lineal por el Método Simplex.

Las soluciones que QSB ofrece al usuario se muestran en pantalla y en papel en forma tabular, además de generar los gráficos correspondientes.

El QSB es un software que puede funcionar en cualquier PC que tenga al menos una unidad de disco. Consta de dos diskettes flexibles (floppys) y para funcionar sólo requiere de una memoria RAM de 512 K como mínimo, un monitor monocromático, de preferencia que tenga adaptador para modo gráfico ó en color a fin de que las soluciones gráficas en pantalla y en papel sean generadas.

Pasos a seguir para entrar al programa QSB.

Para realizar una sesión de trabajo en QSB se requieren los siguientes diskettes:

1) Diskette del MSDOS.

2) Dos diskettes del paquete QSB.

3) Diskette de trabajo,para guardar los archivos.

Teniendo estos diskettes se realizan los siguientes pasos para entrar a QSB:

1) Se introduce el diskette del MSDOS en el drive A, se cierra la compuerta y se enciende el equipo.

2) Una vez que haya sido cargada la PC con el MSDOS esperamos un momento hasta que aparezca el PROMPT A>,y sacamos el diskette del MSDOS y colocamos el disco [1] del Paquete QSB y ponemos:

A>QSB [↵]

Posteriormente oprimir barra espaciadora para continuar y ENTER, para que aparezca en la Pantalla el Menú Principal[2], donde se puede observar que este paquete no solamente nos permite trabajar con Programación Lineal, sino que además podemos utilizarlo para cuestiones estadísticas, caso contreto la obtención de series de tiempo.

Pero estas notas se refieren a la Programación Lineal dentro del Menu Principal elegimos la opción:

1) LINEAR PROGRAMMING

La cual le permitirá al usuario resolver problemas de Programación Lineal con un máximo de 500 variables y 500 restricciones, pero el tamaño de los problemas depende de la Memoria de su PC.

Las opciones de LINEAR PROGRAMMING son:

1.- Mostrar en la pantalla las capacidades y limitaciones

[2] Anexo B

del programa.

2.- Crear un nuevo Problema.

3.- Recuperar un Problema que está almacenado en disco.

4.- Visualizar en pantalla o en Impresora el Planteamiento del Problema.

5.- Solución al Problema Planteado.

6.- Almacenar el Problema en el disco.

7.- Hacer Modificaciones al Problema.

8.- Visualizar e Imprimir la Salidas de Resultados.

9.- Regresar al Menú Principal.

0.- Salir del QSB.

Este paquete trabaja en base a varios Menús que son fáciles de utilizar, dado que nada más se sigue la secuencia de lo que se desea obtener. A continuación se verán los pasos para construir un Modelo de Programación Lineal:

1) Primero se crea un nuevo Problema (ENTER NEW PROBLEM). A continuación se proporciona un Nombre al Problema no mayor de 6 caracteres.

2) Despues se procede a dar las caracteristicas del Problema.

3) Se especifican los coeficientes de las variables y el signo de las restricciones.

4) Una vez que se tiene el planteamiento del problema, lo que se debe de hacer es pedir la solucion del problema, para ello utilizamos la opcion (SOLVE PROBLEM). Donde se puede obtener la solucion a traves de dos Metodos los cuales son:

a) Metodo Dual-Simplex.

b) Metodo Grafico.

Se debe de considerar que la salida de resultados se nos muestra muy similar al Programa PROLIN. Dado que los dos Programas lo muestran en forma de tableaus.

5) Para mandar almacenar el problema al disco se utiliza la opción (SAVE PROBLEM ON DISKETTE).

6) Si se desea mandar a imprimir el Planteamiento del Problema se utiliza (DISPLAY AND/OR PRINT INPUT DATA). Pero si se desea imprimir la salida de resultados entonces, elegimos la opción (DISPLAY AND/OR PRINT FINAL SOLUTION). Donde se debe considerar que para utilizar cada una de estas opciones la impresora debe de estar en línea.

Dichos paquetes antes descritos sólo son analizados en forma muy general, para que el usuario tenga conocimiento de otros paquetes que permiten dar solución a problemas de Programación Lineal. La intención de tal descripción es de manera muy general, puesto que en dicho trabajo no es el objetivo describirlos en su totalidad; pero si que el usuario tenga conocimiento de ellos y se percate de su funcionamiento, puesto que la descripción antes desarrollada le permite utilizarlos sin que para ello sea un conocedor de dicha herramienta.

La descripción de los paquetes le permitirá al usuario elegir entre una gama de ellos, el que mejor se adecue al problema específico y su solución.

CAPITULO V

A N E X O S

En este capítulo damos a la tarea de realizar una serie de ejercicios que comprobarán la efectividad del paquete "PROLIN" y la comparación de él con otros paquetes, así como la metodología que se siguió y una pequeña interpretación económica en los ejemplos que se requieran. Considerando sobre todo a aquellas variables que son representativas económicamente en el modelo, determinando además los valores marginales o precios sombra. Donde se debe considerar que cada ejercicio tiene su propia lógica de planteamiento e interpretación económica.

Los siguientes ejercicios son meramente de carácter matemático y tienen como finalidad mostrar al estudiante cual fue la metodología que se utilizó en la construcción del paquete "PROLIN".

Ejemplo No. 1

TITULO: APUNTES DE PROGRAMACION LINEAL

AUTOR : M.C. MAURICIO VARELA HERNANDEZ

MAX Z= 2X1 + X2 - X3

SUJETA A

X1 - 2X3 <= 5

2X1 - 3X2 + 2X3 >= 3

2X1 - 4X2 + 6X3 <= 5

X1,X2,X3 >= 0

Para poder establecer una lógica apropiada del problema debemos en primera instancia llegar a una estructura metodológica donde los signos deben ser iguales, tal es el caso de la restricción número dos que en comparación con las otras restricciones es de signo contrario por lo cual se debe aplicar la teoría de las desigualdades y aplicar el teorema que establece que una restricción se puede multiplicar por menos uno y nos daría la misma restricción con diferente signo.

$2x1-3x2+2x3>=3$ multiplicado por -1 es igual

$-2x1+3x2-2x3<=-3$

El nuevo planteamiento sería:

MAX Z= 2X1 + X2 - X3

SUJETA A

X1 - 2X3 <= 5

- 2X1 + 3X2 - 2X3 <= -3

2X1 - 4X2 + 6X3 <= 5

X1,X2,X3 >= 0

TABLEAU No. 1

	-X1	-X2	-X3	B
Y1	1	0	-2	5
Y2	-2	3	-2	-3
Y3	2	-4	6	5
MAX Z	-2	-1	1	0

Para resolver este problema es necesario utilizar el Método de Dual-Simplex, porque en lado derecho de la segunda restricción es negativo por lo cual se requiere añadir una variable de holgura llamada RO.

TABLEAU No. 2

	-x1	-x2	-x3	@	B
Y1	1	0	-2	0	5
Y2	-2	3	-2	-1	-3
Y3	2	-4	6	0	5
Z	-2	-1	1	0	0
M	0	0	0	1	0

Para elegir el primer pivote, en este ejemplo en especial es tomar la columna que contiene a @ y el renglón que contiene el signo negativo por lo tanto se tiene el siguiente tableau :

TABLEAU No. 3

	-x1	-x2	-x3	-Y2	B
Y1	1	0	-2	-(0/1)	5
@	-2/-1	3/-1	-2/-1	-(-1/-1	-3/-1
Y3	2	-4	6	-(0/-1)	5
Z	-2	-1	1	-(0/-1)	0
M	-2	3	-2	-(1/-1)	-3

para obtener los valores del tableau se utiliza el procedimiento de Intercambio de Jordan, los cuales pondremos algunos de ellos para una mejor comprensión: cabe aclarar que todos los pivotazos se realizan en tableau No. 2.

PUNTO (Y1,X3)=((-2)*(0))/-1=0 (-2)-(0)= -2

PUNTO (Y3,X2)= ((0)(0))/-1=0 (-4)-(0)= -4

PUNTO (M,X2)= ((3)(1))/-1=-3 (0)-(-3)= 3

PUNTO (M,X1)= ((-2)(1))/-1= 2 (0)-(2)= -2

PUNTO (M,B)= ((-3)(1))/-1= 3 (0)-(3)= -3

De esta manera se obtienen los demás componentes del tableau.

TABLEAU No. 4

	@	-x2	-x3	-Y2	B
Y1	-1/2	3/2	-3	1/2	7/2
Y2	1/2	-3/2	1	-1/2	3/2
Y3	-1	-1	4	1	2
Z	1	-4	3	-1	3
M	1	0	0	0	0

En este tableau se puede eliminar ya la variable de holgura (@,M), dado que ya todas restricciones son de signo positivo por lo tanto nos encontramos en un punto extremo factible. El siguiente tableau nos quedará así:

TABLEAU No. 5

	-X2	-X3	-y2	B
Y1	3/2	-3	1/2	7/2
x1	-3/2	1	-1/2	3/2
Y3	-1	4	1	2
MAX Z	-4	3	-1	3

TABLEAU No. 6

	-Y1	-X3	-Y2	B
X2	2/3	-2	1/3	7/3
X1	1	-2	-1/4	5
Y3	2/3	2	4/3	13/3
MAX Z	8/3	-5	1/3	37/13

TABLEAU No. 7

	-Y1	-Y3	-Y2	B
X2	1.33	1.00	1.67	6.67
X1	1.67	1.00	1.33	9.33
X3	0.33	0.50	0.67	2.17
MAX Z	4.33	2.50	3.67	23.17

En este último tablaeu se obtiene la solución óptima del problema, donde el valor máximo de Z es igual a 23.17. Esta salida de resultados se puede comparar con los resultados que se obtienen utilizando la computadora, con los diversos paquetes.

El archivo "d:uno.dat" contiene : 3 variables independientes
3 restricciones

	x1	x2	x3	signo	B
y1	1.00	0.00	-2.00	<	5.00
y2	-2.00	3.00	-2.00	<	-3.00
y3	2.00	-4.00	6.00	<	5.00
z	2.00	1.00	-1.00		0.00

---

	x1	x2	x3	y2	signo	B
y1	1.00	0.00	-2.00	0.00	<	5.00
ro	2.00	-3.00	2.00	-1.00	<	3.00
y3	2.00	-4.00	6.00	0.00	<	5.00
z	-2.00	-1.00	1.00	0.00		0.00
mu	-2.00	3.00	-2.00	1.00		-3.00

---

	ro	x2	x3	y2	signo	B
y1	-0.50	1.50	-3.00	0.50	<	3.50
x1	0.50	-1.50	1.00	-0.50	<	1.50
y3	-1.00	-1.00	4.00	1.00	<	2.00
z	1.00	-4.00	3.00	-1.00		3.00
mu	1.00	0.00	0.00	0.00		0.00

---

	x2	x3	y2	signo	B
y1	1.50	-3.00	0.50	<	3.50
x1	-1.50	1.00	-0.50	<	1.50
y3	-1.00	4.00	1.00	<	2.00
z	-4.00	3.00	-1.00		3.00

---

	y1	x3	y2	signo	B
x2	0.67	-2.00	0.33	<	2.33
x1	1.00	-2.00	0.00	<	5.00
y3	0.67	2.00	1.33	<	4.33
z	2.67	-5.00	0.33		12.33

---

	y1	y3	y2	signo	B
x2	1.33	1.00	1.67	<	6.67
x1	1.67	1.00	1.33	<	9.33
x3	0.33	0.50	0.67	<	2.17
z	4.33	2.50	3.67		23.17

TABLEAU OPTIMO

Input Data Describing Your Problem uno   Page 1

```
Max  +2.00000X1  +1.00000X2  -1.00000X3
Subject to
(1)  +1.00000X1  ________X2  -2.00000X3  < +5.00000
(2)  -2.00000X1  +3.00000X2  -2.00000X3  < -3.00000
(3)  +2.00000X1  -4.00000X2  +6.00000X3  < +5.00000
```

Summarized Results for uno   Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+9.3333330	0	5	S2	0	+3.6666667
2	X2	+6.6666665	0	6	A2	0	-3.6666667
3	X3	+2.1666667	0	7	S3	0	+2.5000000
4	S1	0	+4.3333335				

Maximum value of the OBJ = 23.16667   Iters. = 3

```
SUBJECT TO
        2)   X1 - 2 X3 <=   5
        3)   2 X1 - 3 X2 + 2 X3 >=   3
        4)   2 X1 - 4 X2 + 6 X3 <=   5
END
: solution

          OBJECTIVE FUNCTION VALUE

  1)        23.1666700

 VARIABLE        VALUE          REDUCED COST
       X1        9.333333           .000000
       X2        6.666667           .000000
       X3        2.166667           .000000

      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)         .000000          4.333334
       3)         .000000         -3.666667
       4)         .000000          2.500000

NO. ITERATIONS=       3
:
```

EJERCICIO No. 2

TITULO : MATEMATICAS PARA ECONOMISTAS.

AUTOR: EDWARD T. DOWLING

PAG 297 Y 298 EJERCICIO 14.2

$$\text{MAXIMICESE } Z = 30X1 + 24X2 + 60X3$$

SUJETO A

$$6X1 + 3X2 + 5X3 <= 30$$

$$2X1 + 2X2 + 10X3 <= 50$$

$$X1, X2, X3 >= 0$$

La solución que se obtiene es la siguiente :

Z = 330 ; X2 = 2.5 ; X3 = 4.5 Y LA VARIABLE NO SELECCIONADA ES X1 QUE ES IGUAL A CERO. LOS VALORES MARGINALES SON, PARA EL PRIMER INSUMO ES 6 Y PARA EL SEGUNDO ES IGUAL A 3.

El archivo "d:dos.dat" contiene : 3 variables independientes
2 restricciones

	x1	x2	x3	signo	B
y1	6.00	3.00	5.00	<	30.00
y2	2.00	2.00	10.00	<	50.00
z	30.00	24.00	60.00		0.00

	x1	x2	y2	signo	B
y1	5.00	2.00	-0.50	<	5.00
x3	0.20	0.20	0.10	<	5.00
z	-18.00	-12.00	6.00		300.00

	y1	x2	y2	signo	B
x1	0.20	0.40	-0.10	<	1.00
x3	-0.04	0.12	0.12	<	4.80
z	3.60	-4.80	4.20		318.00

	y1	x1	y2	signo	B
x2	0.50	2.50	-0.25	<	2.50
x3	-0.10	-0.30	0.15	<	4.50
z	6.00	12.00	3.00		330.00

Input Data Describing Your Problem dos   Page 1

```
Max  +30.0000X1  +24.0000X2  +60.0000X3
Subject to
(1)  +6.00000X1  +3.00000X2  +5.00000X3  < +30.0000
(2)  +2.00000X1  +2.00000X2  +10.0000X3  < +50.0000
```

Summarized Results for dos   Page : 1

Variables No.	Variables Names	Solution	Opportunity Cost	Variables No.	Variables Names	Solution	Opportunity Cost
1	X1	0	+12.000000	4	S1	0	+6.0000000
2	X2	+2.5000000	0	5	S2	0	+3.0000000
3	X3	+4.5000000	0				

Maximum value of the OBJ = 330 Iters. = 3

```
MAX      30 X1 + 24 X2 + 60 X3
SUBJECT TO
        2)   6 X1 + 3 X2 + 5 X3 <=    30
        3)   2 X1 + 2 X2 + 10 X3 <=    50
END

: solution

        OBJECTIVE FUNCTION VALUE

 1)         330.000000

 VARIABLE        VALUE          REDUCED COST
       X1           .000000         12.000000
       X2          2.500000           .000000
       X3          4.500000           .000000


      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)           .000000         6.000000
       3)           .000000         3.000000

NO. ITERATIONS=        2
:
```

EJEMPLO No. 3

TITULO : INVESTIGACION DE OPERACIONES

AUTOR : HAMDY A. TAHA

PAG 277 EJERCICIO 8-26

$$MAX \quad X0 = X1 + 2X2 - 3X3$$

SUJETO A

$$20X1 + 15\,X2 - X3 <= 10$$

$$12X1 - 3X2 + 4X3 <= 20$$

$$X1,X2 >= 0$$

LA SOLUCION ES Z= 1.33, X2=0.67 Y X1,X3 SON IGUAL A CERO.

EJEMPLO No. 4

TITULO : INVESTIGACION DE OPERACIONES

AUTOR : HAMDY A. TAHA

PAG 110 EJERCICIO 4-5

$$MIN \; X0 = X1 + X2 + X3 + X4 + X5 + X6$$

SUJETO A

$$X1 + X6 >= 4$$

$$X1 + X2 >= 8$$

$$X2 + X3 >= 10$$

$$X3 + X4 >= 7$$

$$X4 + X5 >= 12$$

$$X5 + X6 >= 4$$

$$X1,X2,X3,X4,X5,X6 >= 0$$

LOS RESULTADOS SON : Z= -26 , X2=10, X4=12, X6=4 Y DONDE X1,X3,X5 SON IGUAL A CERO.

El archivo "d:tres.dat" contiene : 3 variables independientes
2 restricciones

	x1	x2	x3	signo	B
y1	20.00	15.00	-1.00	<	10.00
y2	12.00	-3.00	4.00	<	20.00
z	1.00	2.00	-3.00	<	0.00

	x1	y1	x3	signo	B
x2	1.33	0.07	-0.07	<	0.67
y2	16.00	0.20	3.80	<	22.00
z	1.67	0.13	2.87	<	1.33

TABLEAU OPTIMO

Input Data Describing Your Problem tres Page 1

```
Max   +1.00000X1  +2.00000X2  -3.00000X3
Subject to
(1)   +20.0000X1  +15.0000X2  -1.00000X3  < +10.0000
(2)   +12.0000X1  -3.00000X2  +4.00000X3  < +20.0000
```

Summarized Results for tres Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	0	+1.6666666	4	S1	0	+.13333334
2	X2	+.66666669	0	5	S2	+22.000000	0
3	X3	0	+2.8666666				

Maximum value of the OBJ = 1.333333 Iters. = 1

```
MAX      X1 + 2 X2 - 3 X3
SUBJECT TO
        2)   20 X1 + 15 X2 - X3 <=    10
        3)   12 X1 - 3 X2 + 4 X3 <=    20
END

: solution

             OBJECTIVE FUNCTION VALUE

 1)          1.33333300

 VARIABLE          VALUE            REDUCED COST
       X1          .000000              1.666667
       X2          .666667               .000000
       X3          .000000              2.866667

       ROW   SLACK OR SURPLUS      DUAL PRICES
        2)         .000000           .133333
        3)       22.000000           .000000

NO. ITERATIONS=        1
:
```

El archivo "d:cuatro.dat" contiene : 6 variables independientes
6 restricciones

	x1	x2	x3	x4	x5	x6	signo	B
y1	1.00	0.00	0.00	0.00	0.00	1.00	>	4.00
y2	1.00	1.00	0.00	0.00	0.00	0.00	>	8.00
y3	0.00	1.00	1.00	0.00	0.00	0.00	>	10.00
y4	0.00	0.00	1.00	1.00	0.00	0.00	>	7.00
y5	0.00	0.00	0.00	1.00	1.00	0.00	>	12.00
y6	0.00	0.00	0.00	0.00	1.00	1.00	>	4.00
z	1.00	1.00	1.00	1.00	1.00	1.00		0.00

	x1	x2	x3	x4	x5	x6	y5	signo	B
y1	-1.00	0.00	0.00	1.00	1.00	-1.00	-1.00	<	8.00
y2	-1.00	-1.00	0.00	1.00	1.00	0.00	-1.00	<	4.00
y3	0.00	-1.00	-1.00	1.00	1.00	0.00	-1.00	<	2.00
y4	0.00	0.00	-1.00	0.00	1.00	0.00	-1.00	<	5.00
ro	0.00	0.00	0.00	1.00	1.00	0.00	-1.00	<	12.00
y6	0.00	0.00	0.00	1.00	0.00	-1.00	-1.00	<	8.00
z	1.00	1.00	1.00	1.00	1.00	1.00	0.00		0.00
mu	0.00	0.00	0.00	-1.00	-1.00	0.00	1.00		-12.00

	x1	x2	x3	y3	x5	x6	y5	signo	B
y1	-1.00	1.00	1.00	-1.00	0.00	-1.00	0.00	<	6.00
y2	-1.00	0.00	1.00	-1.00	0.00	0.00	0.00	<	2.00
x4	0.00	-1.00	-1.00	1.00	1.00	0.00	-1.00	<	2.00
y4	0.00	0.00	-1.00	0.00	1.00	0.00	-1.00	<	5.00
ro	0.00	1.00	1.00	-1.00	0.00	0.00	0.00	<	10.00
y6	0.00	1.00	1.00	-1.00	-1.00	-1.00	0.00	<	6.00
z	1.00	2.00	2.00	-1.00	0.00	1.00	1.00		-2.00
mu	0.00	-1.00	-1.00	1.00	0.00	0.00	0.00		-10.00

	x1	y1	x3	y3	x5	x6	y5	signo	B
x2	-1.00	1.00	1.00	-1.00	0.00	-1.00	0.00	<	6.00
y2	-1.00	0.00	1.00	-1.00	0.00	0.00	0.00	<	2.00
x4	-1.00	1.00	0.00	0.00	1.00	-1.00	-1.00	<	8.00
y4	0.00	0.00	-1.00	0.00	1.00	0.00	-1.00	<	5.00
ro	1.00	-1.00	0.00	0.00	0.00	1.00	0.00	<	4.00
y6	1.00	-1.00	0.00	0.00	-1.00	0.00	0.00	<	0.00
z	3.00	-2.00	0.00	1.00	0.00	3.00	1.00		-14.00
mu	-1.00	1.00	0.00	0.00	0.00	-1.00	0.00		-4.00

	y6	y1	x3	y3	x5	x6	y5	signo	B
x2	1.00	0.00	1.00	-1.00	-1.00	-1.00	0.00	<	6.00
y2	1.00	-1.00	1.00	-1.00	-1.00	0.00	0.00	<	2.00
x4	1.00	0.00	0.00	0.00	0.00	-1.00	-1.00	<	8.00
y4	0.00	0.00	-1.00	0.00	1.00	0.00	-1.00	<	5.00
ro	-1.00	0.00	0.00	0.00	1.00	1.00	0.00	<	4.00
x1	1.00	-1.00	0.00	0.00	-1.00	0.00	0.00	<	0.00
z	-3.00	1.00	0.00	1.00	3.00	3.00	1.00		-14.00
mu	1.00	0.00	0.00	0.00	-1.00	-1.00	0.00		-4.00

	y6	y1	x3	y3	ro	x6	y5	signo	B
x2	0.00	0.00	1.00	-1.00	1.00	0.00	0.00	<	10.00
y2	0.00	-1.00	1.00	-1.00	1.00	1.00	0.00	<	6.00
x4	1.00	0.00	0.00	0.00	0.00	-1.00	-1.00	<	8.00
y4	1.00	0.00	-1.00	0.00	-1.00	-1.00	-1.00	<	1.00
x5	-1.00	0.00	0.00	0.00	1.00	1.00	0.00	<	4.00
x1	0.00	-1.00	0.00	0.00	1.00	1.00	0.00	<	4.00
z	0.00	1.00	0.00	1.00	-3.00	0.00	1.00		-26.00
mu	0.00	0.00	0.00	0.00	1.00	0.00	0.00		0.00

	y6	y1	x3	y3	x6	y5	signo	B
x2	0.00	0.00	1.00	-1.00	0.00	0.00	<	10.00
y2	0.00	-1.00	1.00	-1.00	1.00	0.00	<	6.00
x4	1.00	0.00	0.00	0.00	-1.00	-1.00	<	8.00
y4	1.00	0.00	-1.00	0.00	-1.00	-1.00	<	1.00
x5	-1.00	0.00	0.00	0.00	1.00	0.00	<	4.00
x1	0.00	-1.00	0.00	0.00	1.00	0.00	<	4.00
z	0.00	1.00	0.00	1.00	0.00	1.00		-26.00

	y6	y1	x3	y3	x5	y5	signo	B
x2	0.00	0.00	1.00	-1.00	0.00	0.00	<	10.00
y2	1.00	-1.00	1.00	-1.00	-1.00	0.00	<	2.00
x4	0.00	0.00	0.00	0.00	1.00	-1.00	<	12.00
y4	0.00	0.00	-1.00	0.00	1.00	-1.00	<	5.00
x6	-1.00	0.00	0.00	0.00	1.00	0.00	<	4.00
x1	1.00	-1.00	0.00	0.00	-1.00	0.00	<	0.00
z	0.00	1.00	0.00	1.00	0.00	1.00		-26.00

TABLEAU OPTIMO

```
                 Input Data Describing Your Problem cuatro    Page  1

 Min  +1.00000X1  +1.00000X2  +1.00000X3  +1.00000X4  +1.00000X5
      +1.00000X6
 Subject to
 (1)  +1.00000X1  _________X2  _________X3  _________X4  _________X5
      +1.00000X6  > +4.00000
 (2)  +1.00000X1  +1.00000X2  _________X3  _________X4  _________X5
      _________X6  > +8.00000
 (3)  _________X1  +1.00000X2  +1.00000X3  _________X4  _________X5
      _________X6  > +10.0000
 (4)  _________X1  _________X2  +1.00000X3  +1.00000X4  _________X5
      _________X6  > +7.00000
 (5)  _________X1  _________X2  _________X3  +1.00000X4  +1.00000X5
      _________X6  > +12.0000
 (6)  _________X1  _________X2  _________X3  _________X4  +1.00000X5
      +1.00000X6  > +4.00000
```

Summarized Results for cuatro   Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+4.0000000	0	10	A2	0	0
2	X2	+10.000000	0	11	S3	0	+1.0000000
3	X3	0	0	12	A3	0	-1.0000000
4	X4	+8.0000000	0	13	S4	+1.0000000	0
5	X5	+4.0000000	0	14	A4	0	0
6	X6	0	0	15	S5	0	+1.0000000
7	S1	0	+1.0000000	16	A5	0	-1.0000000
8	A1	0	-1.0000000	17	S6	0	0
9	S2	+6.0000000	0	18	A6	0	0

Minimum value of the OBJ = 26 (multiple sols.)   Iters. = 7

```
MIN      X1 + X2 + X3 + X4 + X5 + X6
SUBJECT TO
        2)   X1 + X6 >=    4
        3)   X1 + X2 >=    8
        4)   X2 + X3 >=   10
        5)   X3 + X4 >=    7
        6)   X4 + X5 >=   12
        7)   X5 + X6 >=    4
END

:
:
```

```
          OBJECTIVE FUNCTION VALUE

 1)         26.0000000

 VARIABLE         VALUE          REDUCED COST
       X1        4.000000           .000000
       X2       10.000000           .000000
       X3         .000000           .000000
       X4        7.000000           .000000
       X5        5.000000           .000000
       X6         .000000           .000000

      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)         .000000         -1.000000
       3)        6.000000           .000000
       4)         .000000         -1.000000
       5)         .000000           .000000
       6)         .000000         -1.000000
       7)        1.000000           .000000

NO. ITERATIONS=        6
:
:
```

EJEMPLO No. 5

TITULO: MATEMATICAS PARA ECONOMISTAS.

AUTOR: EDWARD T. DOWLING

PAG : 288 EJERCICIO 13.20

MAX Z= 24 X1 + 8X2

SUJETA A

2X1 + 5X2 <= 40

10X1 + 5X2 <= 60

4X1 + X2 <= 20

X1,X2 >= 0

DONDE Z=128 , X1=4 Y X2 = 4

Los siguientes ejemplos son de caracter económico, donde se dará además de la salida de resultados una breve interpretación de los mismos. Primeramente se pondrán todos aquellos ejercicios que se refieren a obtener una maximización.

El archivo "d:cinco.dat" contiene : 2 variables independientes
3 restricciones

	x1	x2	signo	B
y1	2.00	5.00	<	40.00
y2	4.00	1.00	<	20.00
y3	10.00	5.00	<	60.00
z	24.00	8.00		0.00

	y2	x2	signo	B
y1	-0.50	4.50	<	30.00
x1	0.25	0.25	<	5.00
y3	-2.50	2.50	<	10.00
z	6.00	-2.00		120.00

	y2	y3	signo	B
y1	4.00	-1.80	<	12.00
x1	0.50	-0.10	<	4.00
x2	-1.00	0.40	<	4.00
z	4.00	0.80		128.00

TABLEAU OPTIMO

Input Data Describing Your Problem cinco Page 1

```
Max  +24.0000X1  +8.00000X2
Subject to
(1)  +2.00000X1  +5.00000X2  < +40.0000
(2)  +10.0000X1  +5.00000X2  < +60.0000
(3)  +4.00000X1  +1.00000X2  < +20.0000
```

Summarized Results for cinco Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+4.0000000	0	4	S2	0	+.80000001
2	X2	+4.0000000	0	5	S3	0	+4.0000000
3	S1	+12.000000	0				

Maximum value of the OBJ = 128 Iters. = 2

```
MAX      24 X1 + 8 X2
SUBJECT TO
        2)   2 X1 + 5 X2 <=    40
        3)   10 X1 + 5 X2 <=    60
        4)   4 X1 + X2 <=    20
END

: solution

          OBJECTIVE FUNCTION VALUE

 1)          128.000000

 VARIABLE        VALUE          REDUCED COST
       X1        4.000000           .000000
       X2        4.000000           .000000


      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)       12.000000           .000000
       3)         .000000           .800000
       4)         .000000          4.000000

NO. ITERATIONS=        2
:
```

EJEMPLO No. 6

TITULO: MATEMATICAS PARA ECONOMISTAS.

AUTOR: EDWARD T. DOWLING

PAG : 280 EJERCICIO 13.2

Un fabricante de ladrillos para patios produce dos tipos distintos: gruesos (x1) y finos (x2). Los ladrillos gruesos necesitan 2 horas de trituración, 5 horas de amalgamación y 8 horas de secado.

Los ladrillos finos necesitan 6 horas de trituración, 3 horas de amalgamación y 2 horas de secado. El margen de beneficio para los ladrillos gruesos es de 40; para los finos, es de 50. El fabricante dispone de 36 horas de trituración, 30 horas de amalgamación y 40 horas de secado.

Determínese la mezcla de producción de maximización de beneficios, reduciendo estos datos a ecuaciones y desigualdades.

PLANTEAMIENTO DEL PROBLEMA

MAXIMICESE Z= 40 X1 + 50 X2

SUJETA A

2 X1 + 6 X2 <= 36

5 X1 + 3 X2 <= 30

8 X1 + 2 X2 <= 40

X1, X2 >= 0

La salida de resultados que se obtuvieron fueron los siguientes:

Las variables seleccionadas fueron : $x1=3$, y $x2=5$ y el valor de la función objetivo fue de $z=370$. Esto representado económicamente quiere decir que si se producen 3 ladrillos gruesos y 5 tipo fino el valor máximo sera de 370. Podemos observar que tanto el recurso trituración y amalgamación fueron consumidos en toda la producción, no así como el tercer recurso dado que se observa que si utilizamos 6 horas de secado para producir los distintos tipos de ladrillos no se verá incrementada en nada la función objetivo. En cambio si nosotros utilizamos 5 horas más de trituración en todo el proceso la función objetivo se verá incrementada en 5.42 unidades.

El archivo "d:seis.dat" contiene : 2 variables independientes
3 restricciones

	x1	x2	signo	B
y1	2.00	6.00	<	36.00
y2	5.00	3.00	<	30.00
y3	8.00	2.00	<	40.00
z	40.00	50.00		0.00

---

	x1	y1	signo	B
x2	0.33	0.17	<	6.00
y2	4.00	-0.50	<	12.00
y3	7.33	-0.33	<	28.00
z	-23.33	8.33		300.00

---

	y2	y1	signo	B
x2	-0.08	0.21	<	5.00
x1	0.25	-0.13	<	3.00
y3	-1.83	0.58	<	6.00
z	5.83	5.42		370.00

TABLEAU OPTIMO

Input Data Describing Your Problem seis Page 1

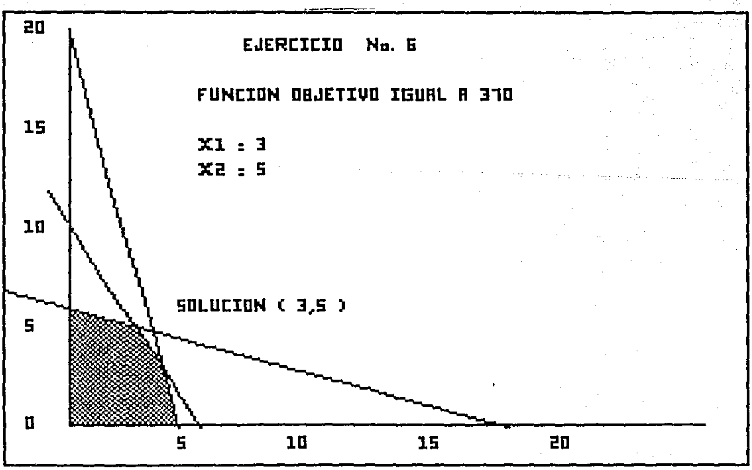
```
Max   +40.0000X1  +50.0000X2
Subject to
(1)   +2.00000X1  +6.00000X2  < +36.0000
(2)   +5.00000X1  +3.00000X2  < +30.0000
(3)   +8.00000X1  +2.00000X2  < +40.0000
```

Summarized Results for seis Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+3.0000000	0	4	S2	0	+5.8333335
2	X2	+5.0000000	0	5	S3	+6.0000000	0
3	S1	0	+5.4166665				

Maximum value of the OBJ = 370 Iters. = 2

```
MAX      40 X1 + 50 X2
SUBJECT TO
        2)   2 X1 + 6 X2 <=    36
        3)   5 X1 + 3 X2 <=    30
        4)   8 X1 + 2 X2 <=    40
END

: solution

         OBJECTIVE FUNCTION VALUE

 1)          370.000000

 VARIABLE         VALUE          REDUCED COST
       X1          3.000000           .000000
       X2          5.000000           .000000


      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)          .000000          5.416667
       3)          .000000          5.833333
       4)         5.999999           .000000

NO. ITERATIONS=       2
:
```



EJEMPLO No. 7

TITULO : INGENIERIA DE SISTEMAS.

AUTOR : VICTOR FLORES ZAVALA.

FACULTAD DE INGENIERIA, PAG 49.

Una compañía denominada " Alimentos para Pollos, S. A." fabrica 2 mezclas de comida para pollo : " Chicken-pollo " y " Coqui-pollo" . Tiene disponible para la solución 2 materias primas, harina de pescado y una base nutriente.

Si los datos de la siguiente tabla se aplican al problema, la compañía quiere saber cuánto producir mensualmente de cada una de las marcas, con el objeto de maximizar sus ganancias:

	CHICKEN	COQUI
CONTENIDO DEL PAQUETE TERMINADO	2.5KG	3.0KG
PRECIO DE VENTA POR PAQUETE	$ 90	$ 75
MATERIAS PRIMAS USADAS POR PAQUETE		
HARINA DE PESCADO	1 KG	2 KG
NUTRIENTE	1.5 KG	1 KG
COSTOS DE LAS MATERIAS PRIMAS		
HARINA DE PESCADO	$ 10/KG	$ 10/KG
NUTRIENTE	$ 20/KG	$ 20/KG
COSTO DE MEZCLADO,EMPAQUE Y DEMAS		
COSTOS VARIABLES POR PAQUETES	$ 14	$ 18
RECURSOS DISPONIBLES PARA LA		
PRODUCCION MENSUAL :		
MATERIA PRIMA :		
HARINA DE PESCADO	240,000 KG	
NUTRIENTE	180,000 KG	

CAPACIDAD DE PROCESAMIENTO: UN MAXIMO DE 110,000 PAQUETES POR MES.

SOLUCION DEL PROBLEMA :

Analizando el problema, queda claro que el objetivo es determinar cuantos paquetes mensuales hay que producir de CHICKEN y COQUI con objeto de obtener la máxima utilidad posible, sujetándose a las restricciones de disponibilidad de harina de pescado, nutrientes, y capacidad de empaque; con ésto se establece la estructura del modelo:

a) Variables : el número de paquetes de cada producto que se deben producir al mes.

$x1$ = Numero de paquetes a producir por mes de Chicken.

$x2$ = Numero de paquetes a producir al mes de Coqui.

b) Parámetros de la función objetivo : Serán los beneficios brutos por paquetes, que se obtienen de la siguiente forma:

beneficio bruto = precio de venta - costo de producción los cuales son para Chicken = 76 y para Coqui = 57 pesos. c) Parámetros del lado derecho de las restricciones: son 240000 kgs de harina de pescado, 180000 kgs de nutrientes y 110000 unidades de capacidad de empaque.

d) Parámetros o Coeficientes de las restricciones : son las cantidades que consumen cada paquete de Chicken y Coqui de harina de pescado, nutriente y capacidad de empaque.

Para el planteamiento del problema es necesario introducir variables de holgura, que permitirán obtener el beneficio máximo.

PLANTEAMIENTO DEL PROBLEMA

MAXIMIZAR $Z= 76X1 + 57X2 + 0X3 + 0X4 + 0X5$

SUJETA A

$$X1 + 2X2 + X3 + 0X4 + 0X5 = 240000$$

$$1.5X1 + X2 + 0X3 + X4 + 0X5 = 180000$$

$$X1 + 0X2 + 0X3 + 0X4 + X5 = 110000$$

$$X1,X2,X3,X4,X5 >= 0$$

Analizando la función objetivo de este modelo, se ve que la solución es igual a 9690 pesos, ya que si a x3 se le diera un valor mayor que cero, z disminuirá en 4.75 pesos y si a x4 se le diera un valor positivo, z disminuirá en 47.5 pesos por cada unidad de aumento. Es decir, ya no se puede obtener un aumento de la función objetivo, lo máximo que se puede conseguir es z = 9690 pesos , produciendo 60000 paquetes mensuales de Chicken y 90000 paquetes mensuales de Coqui, utilizando toda la harina de pescado y nutriente, dejando sin utilizar 50000 unidades de capacidad de empaque.

El archivo "d:siete.dat" contiene : 5 variables independientes
3 restricciones

	x1	x2	x3	x4	x5	signo	B
y1	1.00	2.00	1.00	0.00	0.00	<	240.00
y2	1.50	1.00	0.00	1.00	0.00	<	180.00
y3	1.00	0.00	0.00	0.00	1.00	<	110.00
z	76.00	57.00	0.00	0.00	0.00		0.00

	y3	x2	x3	x4	x5	signo	B
y1	-1.00	2.00	1.00	0.00	-1.00	<	130.00
y2	-1.50	1.00	0.00	1.00	-1.50	<	15.00
x1	1.00	0.00	0.00	0.00	1.00	<	110.00
z	76.00	-57.00	0.00	0.00	76.00		8360.00

	y3	y2	x3	x4	x5	signo	B
y1	2.00	-2.00	1.00	-2.00	2.00	<	100.00
x2	-1.50	1.00	0.00	1.00	-1.50	<	15.00
x1	1.00	0.00	0.00	0.00	1.00	<	110.00
z	-9.50	57.00	0.00	57.00	-9.50		9215.00

	y1	y2	x3	x4	x5	signo	B
y3	0.50	-1.00	0.50	-1.00	1.00	<	50.00
x2	0.75	-0.50	0.75	-0.50	0.00	<	90.00
x1	-0.50	1.00	-0.50	1.00	0.00	<	60.00
z	4.75	47.50	4.75	47.50	0.00		9690.00

TABLEAU OPTIMO

Input Data Describing Your Problem siete Page 1

```
Max   +76.0000X1  +57.0000X2  _________X3  _________X4  _________X5
Subject to
(1)   +1.00000X1  +2.00000X2  +1.00000X3   _________X4  _________X5  < +240.000
(2)   +1.50000X1  +1.00000X2  _________X3  +1.00000X4   _________X5  < +180.000
(3)   +1.00000X1  _________X2 _________X3  _________X4  +1.00000X5   < +110.000
```

Summarized Results for siete Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+59.999996	0	5	X5	0	0
2	X2	+90.000000	0	6	S1	0	+4.7500010
3	X3	0	+4.7500010	7	S2	0	+47.500000
4	X4	0	+47.500000	8	S3	+50.000000	0

Maximum value of the OBJ = 9690 (multiple sols.) Iters. = 3

```
MAX      76 X1 + 57 X2
SUBJECT TO
        2)   X1 + 2 X2 + X3 <=    240
        3)   1.5 X1 + X2 + X4 <=    180
        4)   X1 + X5 <=    110
END

:
:
:

: solution

          OBJECTIVE FUNCTION VALUE

 1)          9690.00000

 VARIABLE        VALUE          REDUCED COST
       X1        60.000000           .000000
       X2        90.000000           .000000
       X3          .000000          4.750000
       X4          .000000         47.500000
       X5          .000000           .000000

      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)          .000000          4.750000
       3)          .000000         47.500000
       4)        50.000000           .000000

NO. ITERATIONS=        2
:
:
:
:
:
```

EJEMPLO No. 8

TITULO : MATEMATICAS PARA ECONOMISTAS.

AUTOR : EDWARD T. DOWLING

PAG 280 EJERCICIO 13.1

Un fabricante acerero especializado produce dos tipos de acero (x1,x2). El tipo 1 requiere 2 horas de fusión, 4 horas de laminado y 10 horas de corte . El tipo 2 necesita 5 horas de fusión, 1 hora de laminado y 5 horas de corte. Se dispone de 40 horas para la fusión, 20 para el laminado y 60 para el corte. El margen de beneficio para el tipo 1 es 24 y para el tipo 2 es 8. Redúzcanse los datos a las ecuaciones y las desigualdades que se necesiten para determinar la dualidad de producción que maximizará los beneficios.

MAXIMICESE Z = 24X1 + 8 X2

SUJETA A:

2X1 + 5X2 <= 40 RESTRICCION DE FUSION

4X1 + X2 <= 20 RESTRICCION DE LAMINACION

10X1 + 5X2 <= 60 RESTRICCION DE CORTE

X1,X2 >= 0

Se observa en la salida de resultados que el fabricante de acero necesita producir 4 unidades de acero de tipo I y 4 unidades de II, para obtener un máximo beneficio de 128 unidades. También se puede ver que si emplea una hora adicional de laminado el fabricante podra obtener un incremento 4 unidades en la función objetivo. En cambio si utiliza en el proceso una hora adicional de fusión, no tendrá ningún incremento su beneficio y estaría desperdiciando un recurso.

El archivo "d:ocho.dat" contiene : 2 variables independientes
3 restricciones

	x1	x2	signo	B
y1	2.00	5.00	<	40.00
y2	10.00	5.00	<	60.00
y3	4.00	1.00	<	20.00
z	24.00	8.00		0.00

	y3	x2	signo	B
y1	-0.50	4.50	<	30.00
y2	-2.50	2.50	<	10.00
x1	0.25	0.25	<	5.00
z	6.00	-2.00		120.00

	y3	y2	signo	B
y1	4.00	-1.80	<	12.00
x2	-1.00	0.40	<	4.00
x1	0.50	-0.10	<	4.00
z	4.00	0.80		128.00

TABLEAU OPTIMO

```
Max   +24.0000X1   +8.00000X2
Subject to
(1)   +2.00000X1   +5.00000X2   < +40.0000
(2)   +4.00000X1   +1.00000X2   < +20.0000
(3)   +10.0000X1   +5.00000X2   < +60.0000
```

Summarized Results for ocho   Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+4.0000000	0	4	S2	0	+4.0000000
2	X2	+4.0000000	0	5	S3	0	+.80000001
3	S1	+12.000000	0				

Maximum value of the OBJ = 128   Iters. = 2

```
MAX      24 X1 + 8 X2
SUBJECT TO
       2)    2 X1 + 5 X2 <=    40
       3)    4 X1 + X2 <=    20
       4)    10 X1 + 5 X2 <=    60
END

: solution

         OBJECTIVE FUNCTION VALUE

 1)          128.000000

 VARIABLE        VALUE          REDUCED COST
       X1        4.000000           .000000
       X2        4.000000           .000000

      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)       12.000000           .000000
       3)         .000000          4.000000
       4)         .000000           .800000

NO. ITERATIONS=       2
:
```

EJEMPLO No. 9

TITULO : MATEMATICAS PARA ECONOMISTAS.

AUTOR : EDWARD T. DOWLING

PAG 280 EJERCICIO 13.3

Un fabricante de juguetes produce dos juegos: BONG (x1) y ZONG (x2), el margen de beneficios sobre BONG es de 30 y el de ZONG es de 20. BONG requiere de 6 horas de elaboración, 4 horas de montaje y 5 horas de embalaje, por su parte ZONG necesita 3 horas de elaboración, 6 horas de montaje y 5 horas de embalaje. Se dispone de 54 horas para la elaboración, 48 horas para el ensamblaje y 50 horas para el embalaje. ¿Cuál es la mezcla de producción para maximizar los beneficios en ecuaciones y desigualdades ?

MAXIMICESE Z = 30 X1 + 20 X2

SUJETA A

6 X1 + 3 X2 <= 54 RESTRICCION DE ELABORACION

4 X1 + 6 X2 <= 48 RESTRICCION DE ENSAMBLAJE

5 X1 + 5 X2 <= 50 RESTRICCION DE EMBALAJE

X1,X2 >= 0

Para este ejemplo se obtuvo que si se producen 8 juguetes de tipo BONG y 2 juguetes de tipo ZONG, tendremos un beneficio máximo de 280.

Si el fabricante utiliza una hora adicional de elaboración, la función objetivo tendra un incremento de 3 unidades. En el caso de que utilice una hora adicional de embalaje, el beneficio se vera incrementado en 2 unidades.

En cambio si utiliza hora adicional de montaje, la función objetivo no se verá afectada en nada. Ya que este recurso en el proceso de producción no es consumido en su totalidad.

El archivo "d:nueve.dat" contiene : 2 variables independientes
3 restricciones

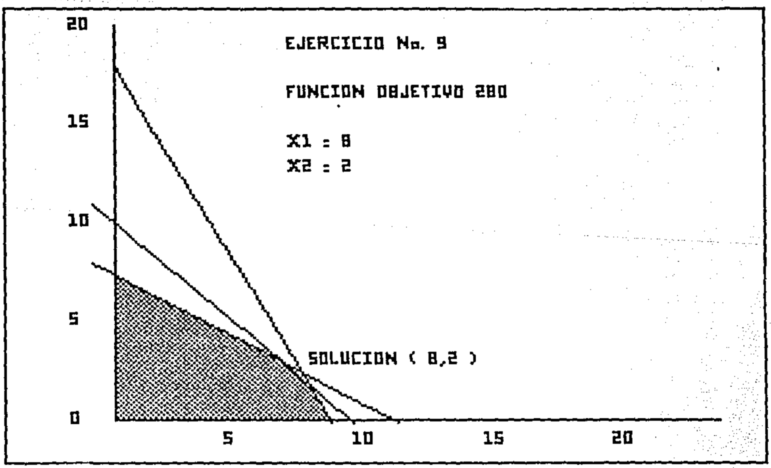
	x1	x2	signo	B
y1	6.00	3.00	<	54.00
y2	4.00	6.00	<	48.00
y3	5.00	5.00	<	50.00
z	30.00	20.00		0.00

	y1	x2	signo	B
x1	0.17	0.50	<	9.00
y2	-0.67	4.00	<	12.00
y3	-0.83	2.50	<	5.00
z	5.00	-5.00		270.00

	y1	y3	signo	B
x1	0.33	-0.20	<	8.00
y2	0.67	-1.60	<	4.00
x2	-0.33	0.40	<	2.00
z	3.33	2.00		280.00

TABLEAU OPTIMO

```
MAX      30 X1 + 20 X2
SUBJECT TO
        2)    6 X1 + 3 X2 <=    54
        3)    4 X1 + 6 X2 <=    48
        4)    5 X1 + 5 X2 <=    50
END

: solution

         OBJECTIVE FUNCTION VALUE

 1)          280.000000

 VARIABLE        VALUE          REDUCED COST
       X1        8.000000           .000000
       X2        2.000000           .000000


      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)         .000000          3.333333
       3)        4.000000           .000000
       4)         .000000          2.000000

NO. ITERATIONS=       2
:
```



EJEMPLO No. 10

TITULO : MATEMATICAS PARA ECONOMISTAS.

AUTOR : EDWARD T. DOWLING

PAG 280 EJERCICIO 13.4

Un fabricante de tocadiscos produce 3 tipos de aparatos: standar(x1), de calidad (x2) y de lujo (x3). Sus márgenes de beneficio son de 15, 20 y 24 respectivamente. El modelo estándar requiere de 3 horas de alambrado y 1 hora de sujeción a la caja. El de calidad necesita 1 hora de alambrado y 5 horas de sujeción. El de lujo requiere 3 horas de alambrado y 2 horas para la sujeción a la caja. Si se dispone de 120 horas para alambrado y 60 horas para la sujeción a la caja, exprésese la mezcla de productos que maximizarán los beneficios en la forma de ecuaciones y desigualdades.

MAXIMICESE Z = 15 X1 + 20 X2 + 24 X3

SUJETA A

3X1 + X2 + 3X3 <= 120 RESTRICCION DE ALAMBRADO

X1 + 5X2 + 2X3 <= 60 SUJECIÓN A LA CAJA

X1,X2,X3 >= 0

Input Data Describing Your Problem nueve   Page 1

Max +30.0000X1 +20.0000X2
Subject to
(1) +6.00000X1 +3.00000X2 < +54.0000
(2) +4.00000X1 +6.00000X2 < +48.0000
(3) +5.00000X1 +5.00000X2 < +50.0000

Summarized Results for nueve   Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+8.0000000	0	4	S2	+4.0000000	0
2	X2	+2.0000000	0	5	S3	0	+2.0000000
3	S1	0	+3.3333333				

Maximum value of the OBJ = 280 Iters. = 2

El fabricante de tocadiscos para obtener una ganancia de \$780, tiene que producir 20 tocadicos de tipo standar y 20 tocadiscos de lujo, ya que se si produce un tocadisco de calidad su beneficio máximo se vera desminuido en \$27.

Teniendose que todas las horas que dispone de alambrado y sujeción son consumidas en su totalidad en el proceso de producción . En caso que si el fabricante deseara incrementar el valor máximo de su beneficio, se recomienda que por cada hora adicional se sujeción que se utilice la función objetivo se incrementara en 9 unidades.

El archivo "d:dies.dat" contiene : 3 variables independientes
2 restricciones

	x1	x2	x3	signo	B
y1	3.00	1.00	3.00	<	120.00
y2	1.00	5.00	2.00	<	60.00
z	15.00	20.00	24.00	<	0.00

	x1	x2	y2	signo	B
y1	1.50	-6.50	-1.50	<	30.00
x3	0.50	2.50	0.50	<	30.00
z	-3.00	40.00	12.00	<	720.00

	y1	x2	y2	signo	B
x1	0.67	-4.33	-1.00	<	20.00
x3	-0.33	4.67	1.00	<	20.00
z	2.00	27.00	9.00	<	780.00

TABLEAU OPTIMO

Input Data Describing Your Problem dies Page 1

```
Max   +15.0000X1  +20.0000X2  +24.0000X3
Subject to
(1)   +3.00000X1  +1.00000X2  +3.00000X3  < +120.000
(2)   +1.00000X1  +5.00000X2  +2.00000X3  < +60.0000
```

Summarized Results for dies Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+19.999996	0	4	S1	0	+2.0000000
2	X2	0	+27.000000	5	S2	0	+9.0000000
3	X3	+20.000002	0				

Maximum value of the OBJ = 780 Iters. = 2

```
MAX      15 X1 + 20 X2 + 24 X3
SUBJECT TO
        2)   3 X1 + X2 + 3 X3 <=    120
        3)     X1 + 5 X2 + 2 X3 <=    60
END

: solution

         OBJECTIVE FUNCTION VALUE

 1)          780.000000

 VARIABLE        VALUE          REDUCED COST
       X1        20.000000           .000000
       X2          .000000         27.000000
       X3        20.000000           .000000


      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)          .000000          2.000000
       3)          .000000          9.000000

NO. ITERATIONS=        2
:
```

EJEMPLO No. 11

TITULO : INVESTIGACION DE OPERACIONES

AUTOR ; VICTOR RIOS GARCIA

PAG :38,39, 40 Y 41 EJERCICIO No. 1

Una ama de casa interesada por satisfacer las necesidades alimenticias de su esposo, empleando para ello el menor gasto posible, decide comprar solamente leche, carne y huevos.

Al llegar al mercado encuentra los siguientes costos: Un litro de leche cuesta $1200 , un Kilo de carne cuesta $13000 y una docena de huevos $ 4200. Consultando su guía de dietas aprende que el contenido de vitamina B en las anteriores cantidades de alimento es igual a 20 grs, 10 grs y 10 grs, respectivamente. El contenido de vitamina C es en forma análoga: 10 grs, 20 grs, y 10 grs.

¿Qué cantidades de cada uno de los alimentos debe comprar el ama de casa si los requerimientos mínimos diarios para un adulto son de 60 grs de vitamina B y 50 grs de vitamina C.?

PLANTEAMIENTO DEL PROBLEMA

	LECHE	CARNE	HUEVOS	REQUERIMIENTOS
VIT B	20	10	10	60
VIT C	10	20	10	50
COSTO	1200	13000	4200	

$$\text{MIN} \quad Z = 1200\ X1 + 13000\ X2 + 4200\ X3$$

SUJETA A

$$20\ X1 + 10\ X2 + 10\ X3 >= 60$$

$$10\ X1 + 20\ X2 + 10\ X3 >= 50$$

$$X1, X2, X3 >= 0$$

Para satisfacer las necesidades alimenticias es necesario comprar solamente 5 litros de leche, cubriendo así las necesidades vitamínicas, que requiere un adulto con un costo mínimo de $ 6000, y no es necesario comprar ninguna unidad de los otros productos.

Dado que si se compra un kilo de carne el costo se incrementará en $ 10600, con esto la función objetivo tendría un valor de $ 16600. Por otro lado se observa que no se utiliza toda la vitamina B, teniendose un sobrante de 40 gramos, a diferencia de la vitamina C que si cubre las necesidades alimenticias con 50 gramos. Por otra parte si se utiliza en el proceso de alimentación una unidad adicional de vitamina C el costo se desminuirá en $ 120, no ocurre lo mismo con la vitamina B, ya que esta solamente con 20 gramos satisface los requerimientos mínimos.

```
El archivo "d:once.dat" contiene : 3  variables independientes
                                   2   restricciones

        x1        x2        x3 signo   B
 y1   20.00    10.00    10.00    >   60.00
 y2   10.00    20.00    10.00    >   50.00
  z 1200.0013000.00 4200.00    <      0.00
___________________________________________________________

        x1        x2        x3        y1 signo   B
 ro   20.00    10.00    10.00    -1.00    <    60.00
 y2   10.00   -10.00     0.00    -1.00    <    10.00
  z 1200.0013000.00 4200.00     0.00    <     0.00
  z  -20.00   -10.00   -10.00     1.00        -60.00
___________________________________________________________

        y2        x2        x3        y1 signo   B
 ro   -2.00    30.00    10.00     1.00    <    40.00
 x1    0.10    -1.00     0.00    -0.10    <     1.00
  z -120.0014200.00 4200.00   120.00    <-1200.00
  z    2.00   -30.00   -10.00    -1.00        -40.00
___________________________________________________________

        y2        ro        x3        y1 signo   B
 x2   -0.07     0.03     0.33     0.03    <     1.33
 x1    0.03     0.03     0.33    -0.07    <     2.33
  z  826.67  -473.33  -533.33  -353.33    <-20133.33
  z    0.00     1.00     0.00     0.00          0.00
___________________________________________________________

        y2        x3        y1 signo   B
 x2   -0.07     0.33     0.03    <     1.33
 x1    0.03     0.33    -0.07    <     2.33
  z  826.67  -533.33  -353.33    < -20133.33
___________________________________________________________

        y2        x2        y1 signo   B
 x3   -0.20     3.00     0.10    <     4.00
 x1    0.10    -1.00    -0.10    <     1.00
  z  720.00  1600.00  -300.00    < -18000.00
___________________________________________________________

        y2        x2        x3 signo   B
 y1   -2.00    30.00    10.00    <    40.00
 x1   -0.10     2.00     1.00    <     5.00
  z  120.0010600.00 3000.00    < -6000.00
___________________________________________________________
```

TABLEAU OPTIMO

Input Data Describing Your Problem once   Page 1

```
Min   +1200.00X1   +13000.0X2   +4200.00X3
Subject to
(1)   +20.0000X1   +10.0000X2   +10.0000X3   >  +60.0000
(2)   +10.0000X1   +20.0000X2   +10.0000X3   >  +50.0000
```

Summarized Results for once   Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+5.0000000	0	5	A1	0	0
2	X2	0	+10600.000	6	S2	0	+120.00001
3	X3	0	+3000.0000	7	A2	0	-120.00001
4	S1	+40.000000	0				

Minimum value of the OBJ = 6000.001   Iters. = 2

```
MIN      1200 X1 + 13000 X2 + 4200 X3
SUBJECT TO
        2)   20 X1 + 10 X2 + 10 X3 >=    60
        3)   10 X1 + 20 X2 + 10 X3 >=    50
END

: solution

           OBJECTIVE FUNCTION VALUE

 1)          6000.00000

 VARIABLE        VALUE          REDUCED COST
       X1         5.000000           .000000
       X2          .000000       10600.000000
       X3          .000000        3000.000000

      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)        40.000000           .000000
       3)          .000000       -120.000000

NO. ITERATIONS=        3
:
```

EJEMPLO No. 12

TITULO : MATEMATICAS PARA ECONOMISTAS

AUTOR : EDWARD T. DOWLING

PAG 281 EJERCICIO 13.6

Un Horticultor desea mezclar fertilizantes que proporcionen un mínimo de 15 unidades de potasa, 20 unidades de nitratos, y 24 unidades de fosfatos. La marca 1 proporciona tres unidades de potasa, 1 nitratos y 3 de fosfatos; su costo es de 120 dólares. La marca 2 da 1 unida de de potasa, 5 de nitatros y 2 de fosfatos; su costo es de 60 dólares.

Exprésese la combinación de fertilizantes de menor costo que satisfará las especificaciones deseadas, como ecuaciones y desigualdades.

MIN 120 X1 + 60 X2

SUJETA A

3 X1 + X2 >= 15

X1 + 5 X2 >= 20

3 X1 + 2 X2 >= 24

X1,X2,X3 >= 0

En la salida se resultados se observó que si se mezcla 3 unidades de potasa, una unidad de nitrato y 3 unidades de fosfato se obtienen 2 marcas de tipo 1. Y al mezclar uno de potasa, 5 de nitrato y 2 de fosfato se obtuvieron 9 marcas de tipo 2, teniéndose un costo mínimo de 780 dólares.

El archivo "d:doce.dat" contiene : 2 variables independientes
3 restricciones

	x1	x2	signo	B
y1	3.00	1.00	>	15.00
y2	1.00	5.00	>	20.00
y3	3.00	2.00	>	24.00
z	120.00	60.00		0.00

---

	x1	x2	y3	signo	B
y1	0.00	1.00	-1.00	<	9.00
y2	2.00	-3.00	-1.00	<	4.00
ro	3.00	2.00	-1.00	<	24.00
mu	120.00	60.00	0.00		0.00
	-3.00	-2.00	1.00		-24.00

---

	y2	x2	y3	signo	B
y1	0.00	1.00	-1.00	<	9.00
x1	0.50	-1.50	-0.50	<	2.00
ro	-1.50	6.50	0.50	<	18.00
mu	-60.00	240.00	60.00		-240.00
	1.50	-6.50	-0.50		-18.00

---

	y2	ro	y3	signo	B
y1	0.23	-0.15	-1.08	<	6.23
x1	0.15	0.23	-0.38	<	6.15
x2	-0.23	0.15	0.08	<	2.77
mu	-4.62	-36.92	41.54		-904.62
	0.00	1.00	0.00		0.00

---

	y2	y3	signo	B
y1	0.23	-1.08	<	6.23
x1	0.15	-0.38	<	6.15
x2	-0.23	0.08	<	2.77
mu	-4.62	41.54		-904.62

---

	y1	y3	signo	B
y2	4.33	-4.67	<	27.00
x1	-0.67	0.33	<	2.00
x2	1.00	-1.00	<	9.00
mu	20.00	20.00		-780.00

TABLEAU OPTIMO

```
Min  +120.000X1  +60.0000X2
Subject to
(1)  +3.00000X1  +1.00000X2  > +15.0000
(2)  +1.00000X1  +5.00000X2  > +20.0000
(3)  +3.00000X1  +2.00000X2  > +24.0000
```

Summarized Results for doce Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+2.0000000	0	5	S2	+27.000000	0
2	X2	+9.0000000	0	6	A2	0	0
3	S1	0	+19.999996	7	S3	0	+20.000002
4	A1	0	-19.999996	8	A3	0	-20.000002

Minimum value of the OBJ = 780 Iters. = 4

```
MIN      120 X1 + 60 X2
SUBJECT TO
        2)    3 X1 + X2 >=    15
        3)    X1 + 5 X2 >=    20
        4)    3 X1 + 2 X2 >=    24
END

: solution

         OBJECTIVE FUNCTION VALUE

  1)         780.000000

  VARIABLE        VALUE          REDUCED COST
        X1        2.000000           .000000
        X2        9.000000           .000000


       ROW   SLACK OR SURPLUS     DUAL PRICES
        2)         .000000        -20.000000
        3)       27.000000           .000000
        4)         .000000        -20.000000

 NO. ITERATIONS=        3
 :
```

```
MIN      120 X1 + 60 X2
SUBJECT TO
        2)    3 X1 + X2 >=    15
        3)    X1 + 5 X2 >=    20
        4)    3 X1 + 2 X2 >=    24
END

: solution

          OBJECTIVE FUNCTION VALUE

 1)          780.000000

 VARIABLE          VALUE           REDUCED COST
       X1          2.000000            .000000
       X2          9.000000            .000000

      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)           .000000       -20.000000
       3)         27.000000          .000000
       4)           .000000       -20.000000

NO. ITERATIONS=       3
:
```

EJEMPLO No. 13

TITULO :MATEMATICAS PARA ECONOMISTAS

AUTOR : EDWARD T. DOWLING

PAG 281 EJERCICIO 13.7

Un entusiasta de la salud desea tener un mínimo de 36 unidades de vitamina A al dia, 28 unidades de vitamina C y 32 unidades de vitamina D. La marca uno cuesta 3 dólares y proporciona 2 unidades de vit A, 2 Vit B y 8 Vit D. La marca 2 cuesta 4 dólares y da 3 unidades de Vit A, 2 de C y 2 de D. En ecuaciones y desigualdades, ¿Cual es la combinación de costo más bajo que garantiza las necesidades diarias ?.

MIN Z = 3 X1 + 4 X2

SUJETO A

2 X1 + 3 X2 >= 36

2 X1 + 2 X2 >= 28

8 X1 + 2 X2 >= 32

X1 , X2 >= 0

El archivo "d:trece.dat" contiene : 2 variables independientes
3 restricciones

	x1	x2	signo	B
y1	2.00	3.00	>	36.00
y2	2.00	2.00	>	28.00
y3	8.00	2.00	>	32.00
z	3.00	4.00		0.00

---

	x1	x2	y1	signo	B
ro	2.00	3.00	-1.00	<	36.00
y2	0.00	1.00	-1.00	<	8.00
y3	-6.00	1.00	-1.00	<	4.00
mu	3.00	4.00	0.00		0.00
	-2.00	-3.00	1.00		-36.00

---

	x1	y3	y1	signo	B
ro	20.00	-3.00	2.00	<	24.00
y2	6.00	-1.00	0.00	<	4.00
x2	-6.00	1.00	-1.00	<	4.00
mu	27.00	-4.00	4.00		-16.00
	-20.00	3.00	-2.00		-24.00

---

	y2	y3	y1	signo	B
ro	-3.33	0.33	2.00	<	10.67
x1	0.17	-0.17	0.00	<	0.67
x2	1.00	0.00	-1.00	<	8.00
mu	-4.50	0.50	4.00		-34.00
	3.33	-0.33	-2.00		-10.67

---

	y2	y3	ro	signo	B
y1	-1.67	0.17	0.50	<	5.33
x1	0.17	-0.17	0.00	<	0.67
x2	-0.67	0.17	0.50	<	13.33
mu	2.17	-0.17	-2.00		-55.33
	0.00	0.00	1.00		0.00

---

	y2	y3	signo	B
y1	-1.67	0.17	<	5.33
x1	0.17	-0.17	<	0.67
x2	-0.67	0.17	<	13.33
mu	2.17	-0.17		-55.33

---

	y2	y1	signo	B
y3	-10.00	6.00	<	32.00
x1	-1.50	1.00	<	6.00
x2	1.00	-1.00	<	8.00
mu	0.50	1.00		-50.00

TABLEAU OPTIMO

Input Data Describing Your Problem trece Page 1

```
Min    +3.00000X1   +4.00000X2
Subject to
(1)   +2.00000X1   +3.00000X2   > +36.0000
(2)   +2.00000X1   +2.00000X2   > +28.0000
(3)   +8.00000X1   +2.00000X2   > +32.0000
```

Summarized Results for trece Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+6.0000000	0	5	S2	0	+.50000000
2	X2	+8.0000000	0	6	A2	0	-.50000000
3	S1	0	+1.0000000	7	S3	+32.000000	0
4	A1	0	-1.0000000	8	A3	0	0

Minimum value of the OBJ = 50 Iters. = 4

```
MIN      3 X1 + 4 X2
SUBJECT TO
       2)   2 X1 + 3 X2 >=   36
       3)   2 X1 + 2 X2 >=   28
       4)   8 X1 + 2 X2 >=   32
END

: solution

         OBJECTIVE FUNCTION VALUE

 1)         50.0000000

 VARIABLE          VALUE          REDUCED COST
       X1          6.000000           .000000
       X2          8.000000           .000000


      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)          .000000         -1.000000
       3)          .000000          -.500000
       4)        32.000000           .000000

NO. ITERATIONS=       3
:
```

EJEMPLO No. 14

TITULO : MATEMATICAS PARA ECONOMISTAS

AUTOR : EDWARD T. DOWLING

PAG 281 EJERCICIO 13,8

Hank Burdue se asegura de que sus pollos reciban cada día, al menos, 24 unidades de hierro y 8 de vitamina. El maíz (x1) proporciona 2 unidades de hierro y 5 unidades de vitamina. El alimento de harina de huesos (x2) da 4 unidades de hierro y 1 de vitaminas. La mezcla (x3) proporciona 2 unidades de hierro y 1 de vitamina. ¿Como se deberán mezclar los alimentos para proporcionar la satisfación de las necesidades diarias al menor costo, si los costos de los alimentos citados son de 40,20 y 60 dólares, respectivamente?

MIN Z = 40 X1 + 20 X2 + 60 X3

SUJETA A

2 X1 + 4 X2 + 2 X3 >= 24

5 X1 + X2 + X3 >= 8

X1 , X2 X3 >= 0

El propietario de los pollos deberá gastar 133.333 dolares, que representar el mínimo costo para satisfacer los requerimientos alimencios. Esto se obtiene a traves de mezclar los siguientes alimentos: de maíz 0.44 unidades y 5.77 unidades de harina de hueso. Por otra parte por cada unidad de gallinaza que se utilice en el modelo, la función objetivo se incrementara en 46.67 unidades.

En cuanto a los requerimientos mínimos de alimentación se observa que éstos cumplen satisfactoriamente las necesidades que necesitan los pollos. Ahora bien, si el avicultor en los alimentos seleccionados emplea un procedimiento de mejora, tendremos que por cada unidad adicional de hierro y vitaminas el costo desminuirá en 10 unidades.

El archivo "d:catorce.dat" contiene : 3 variables independientes
2 restricciones

	x1	x2	x3	signo	B
y1	2.00	4.00	2.00	>	24.00
y2	5.00	1.00	1.00	>	8.00
z	40.00	20.00	60.00	<	0.00

---

	x1	x2	x3	y1	signo	B
ro	2.00	4.00	2.00	-1.00	<	24.00
y2	-3.00	3.00	1.00	-1.00	<	16.00
z	40.00	20.00	60.00	0.00	<	0.00
mu	-2.00	-4.00	-2.00	1.00		-24.00

---

	x1	y2	x3	y1	signo	B
ro	6.00	-1.33	0.67	0.33	<	2.67
x2	-1.00	0.33	0.33	-0.33	<	5.33
z	60.00	-6.67	53.33	6.67	<	-106.67
mu	-6.00	1.33	-0.67	-0.33		-2.67

---

	ro	y2	x3	y1	signo	B
x1	0.17	-0.22	0.11	0.06	<	0.44
x2	0.17	0.11	0.44	-0.28	<	5.78
z	-10.00	6.67	46.67	3.33	<	-133.33
mu	1.00	0.00	0.00	0.00		0.00

---

	y2	x3	y1	signo	B
x1	-0.22	0.11	0.06	<	0.44
x2	0.11	0.44	-0.28	<	5.78
z	6.67	46.67	3.33	<	-133.33

---

	x2	x3	y1	signo	B
x1	2.00	1.00	-0.50	<	12.00
y2	9.00	4.00	-2.50	<	52.00
z	-60.00	20.00	20.00	<	-480.00

---

	y2	x3	y1	signo	B
x1	-0.22	0.11	0.06	<	0.44
x2	0.11	0.44	-0.28	<	5.78
z	6.67	46.67	3.33	<	-133.33

TABLEAU OPTIMO

```
Min  +40.0000X1  +20.0000X2  +60.0000X3
Subject to
(1)  +2.00000X1  +4.00000X2  +2.00000X3  >  +24.0000
(2)  +5.00000X1  +1.00000X2  +1.00000X3  >  +8.00000
```

Summarized Results for catorc Page : 1

Variables No.	Names	Solution	Opportunity Cost	Variables No.	Names	Solution	Opportunity Cost
1	X1	+.44444445	0	5	A1	0	-3.3333333
2	X2	+5.7777777	0	6	S2	0	+6.6666665
3	X3	0	+46.666668	7	A2	0	-6.6666665
4	S1	0	+3.3333333				

Minimum value of the OBJ = 133.3333 Iters. = 2

```
MIN      40 X1 + 20 X2 + 60 X3
SUBJECT TO
       2)    2 X1 + 4 X2 + 2 X3 >=    24
       3)    5 X1 + X2 + X3 >=    8
END

: solution

         OBJECTIVE FUNCTION VALUE

  1)           133.333300

 VARIABLE          VALUE          REDUCED COST
       X1           .444444           .000000
       X2          5.777778           .000000
       X3           .000000         46.666660

      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)           .000000       -3.333333
       3)           .000000       -6.666667

NO. ITERATIONS=        2
:
```

EJEMPLO No. 15

TITULO : MATEMATICAS PARA ECONOMISTAS.

AUTOR : EDWARD T. DOWLING

PAG 277 EJEMPLO No. 2

Un ganadero desea asegurarse que sus animales obtengan los requerimientos diarios básicos de 3 nutrientes esenciales: A, B y C. Los requerimientos diarios son de 14 para A, 12 para B y 18 para C. El producto x1 tiene 2 unidades de A y 1 de B y C ; el producto x2 tiene 1 de A, 1 de B y 3 de C. El costo de x1 es 2 dólares y de x2 de 4 dolares. Determinar la combinación de menor costo de x1 y x2 que satisfaga todas las necesidades mínimas.

MINIMICESE Z = 2X1 + 4X2

SUJETA A

2 X1 + X2 > = 14 RESTRICCION DE A

X1 + X2 >= 12 RESTRICCION DE B

X1 + 3 X2 >= 18 RESTRICCION DE C

X1,X2 >= 0

El ganadero para asegurar que sus animales obtengan sus requerimientos diarios de nutrientes deberá invertir 30 dolares, que representan el costo mínimo por utilizar 9 unidades del producto X1 y 3 unidades de X2. Por lo tanto, nos encontramos que los nutrientes B y C satisfacen en su totalidad las necesidades mínimas del ganado. En cambio el nutriente A se observa que se esta desaprovechando en el proceso en 7 unidades.

El archivo "d:quince.dat" contiene : 2 variables independientes
3 restricciones

	x1	x2	signo	B
y1	2.00	1.00	>	14.00
y2	1.00	1.00	>	12.00
y3	1.00	3.00	>	18.00
z	2.00	4.00		0.00

---

	x1	x2	y3	signo	B
y1	-1.00	2.00	-1.00	<	4.00
y2	0.00	2.00	-1.00	<	6.00
ro	1.00	3.00	-1.00	<	18.00
mu	2.00	4.00	0.00		0.00
mu	-1.00	-3.00	1.00		-18.00

---

	x1	y1	y3	signo	B
x2	-0.50	0.50	-0.50	<	2.00
y2	1.00	-1.00	0.00	<	2.00
ro	2.50	-1.50	0.50	<	12.00
mu	4.00	-2.00	2.00		-8.00
mu	-2.50	1.50	-0.50		-12.00

---

	y2	y1	y3	signo	B
x2	0.50	0.00	-0.50	<	3.00
x1	1.00	-1.00	0.00	<	2.00
ro	-2.50	1.00	0.50	<	7.00
mu	-4.00	2.00	2.00		-16.00
mu	2.50	-1.00	-0.50		-7.00

---

	y2	ro	y3	signo	B
x2	0.50	0.00	-0.50	<	3.00
x1	-1.50	1.00	0.50	<	9.00
y1	-2.50	1.00	0.50	<	7.00
mu	1.00	-2.00	1.00		-30.00
mu	0.00	1.00	0.00		0.00

---

	y2	y3	signo	B
x2	0.50	-0.50	<	3.00
x1	-1.50	0.50	<	9.00
y1	-2.50	0.50	<	7.00
mu	1.00	1.00		-30.00

---

	y2	y1	signo	B
x2	-2.00	1.00	<	10.00
x1	1.00	-1.00	<	2.00
y3	-5.00	2.00	<	14.00
mu	6.00	-2.00		-44.00

	y2	y3	signo	B
x2	0.50	-0.50	<	3.00
x1	-1.50	0.50	<	9.00
y1	-2.50	0.50	<	7.00
mu	1.00	1.00		-30.00

TABLEAU OPTIMO

Input Data Describing Your Problem quince Page 1

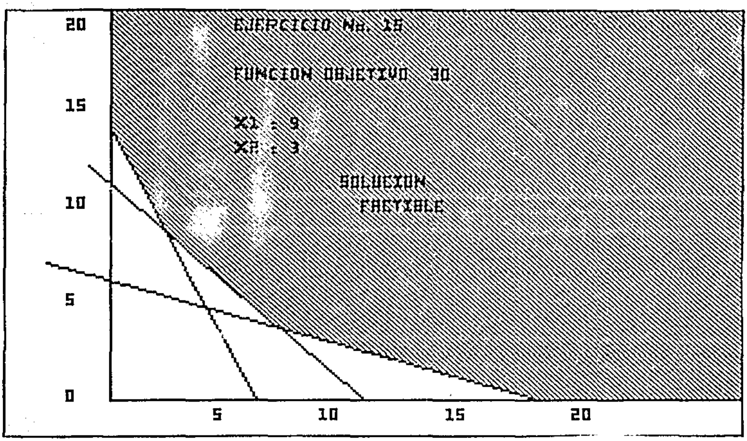
```
Min  +2.00000X1  +4.00000X2
Subject to
(1)  +2.00000X1  +1.00000X2  > +14.0000
(2)  +1.00000X1  +1.00000X2  > +12.0000
(3)  +1.00000X1  +3.00000X2  > +18.0000
```

Summarized Results for quince Page : 1

Variables No. Names	Solution	Opportunity Cost	Variables No. Names	Solution	Opportunity Cost
1 X1	+9.0000000	0	5 S2	0	+1.0000000
2 X2	+3.0000000	0	6 A2	0	-1.0000000
3 S1	+7.0000000	0	7 S3	0	+1.0000000
4 A1	0	0	8 A3	0	-1.0000000

Minimum value of the OBJ = 30 Iters. = 3

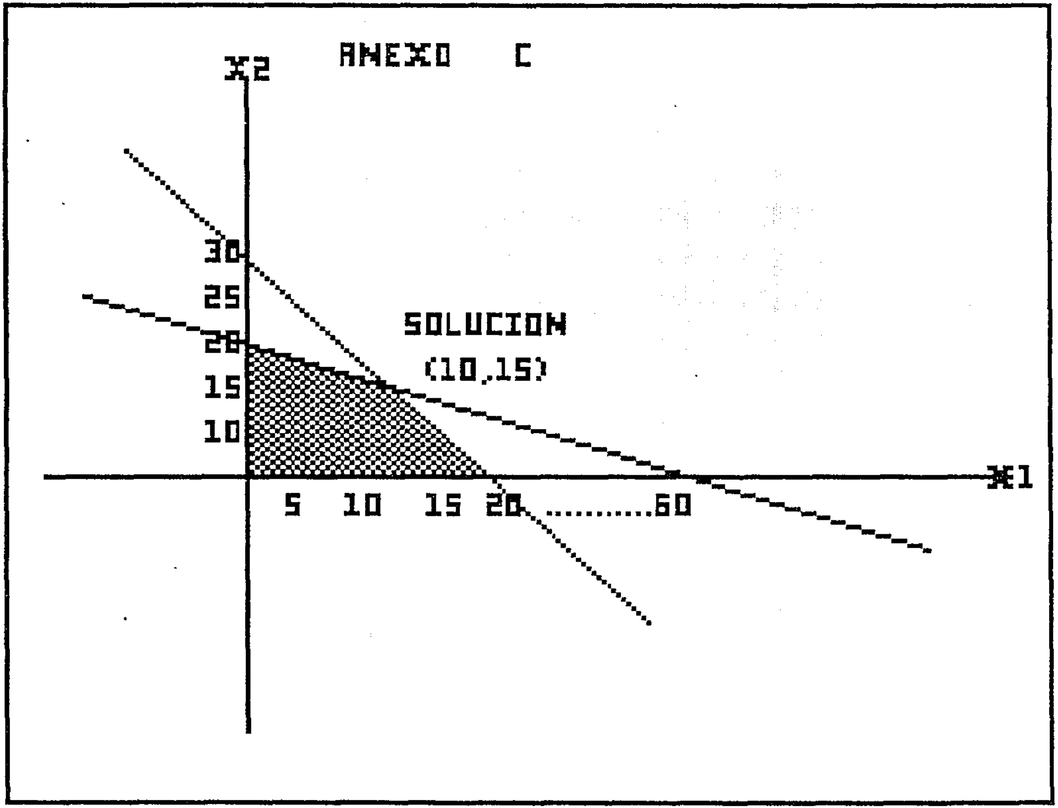
```
MIN      2 X1 + 4 X2
SUBJECT TO
       2)    2 X1 + X2 >=    14
       3)    X1 + X2 >=    12
       4)    X1 + 3 X2 >=    18
END

: solution

         OBJECTIVE FUNCTION VALUE

 1)          30.0000000

 VARIABLE        VALUE          REDUCED COST
      X1        9.000000           .000000
      X2        3.000000           .000000


      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)        6.999999           .000000
       3)         .000000         -1.000000
       4)         .000000         -1.000000

NO. ITERATIONS=        3
:
```



WELCOME TO QSB (QUANTITATIVE SYSTEMS FOR BUSINESS) YOU MAY CHOOSE FROM FOLLOWING MANAGEMENT SCIENCE DECISION SUPPORT	
CODE No. PROGRAM	
1.- LINEAR PROGRAMMING 2.-INTEGER LINEAR PROGRAMMING 3.- TRANSSHIPMENT PROBLEM 4.- ASSIGNMENT PROBLEM 5.- NETWORK MODELING 6.- PROJECT SCHEDULING--CPM 7.- PROJECT SCHEDULING--PERT 8.- DYNAMIC PROGRAMMING	9.- INVENTORY THEORY A.- QUEUING THEORY B.- QUEUING SYSTEM SIMULATION C.- DECISION/PROBABILITY THEORY D.- MARKOV PROCESS E.- TIME SERIES FORECASTING F.- SPECIFY THE TYPE OF PRINTER G.- EXIT FROM QSB.

QSB (I): PROGRAM 1 TO 5 , QSB (II): PROGRAM 6 TO E

```
LINDO COMMANDS BY CATEGORY, FOR INFORMATION ON A SPECIFIC
COMMAND, TYPE: HELP FOLLOWED BY THE COMMAND NAME.
1) INFORMATION
   HELP   COM   CAT
2) INPUT
   MAX  MIN  RETR   TAKE  LEAV
3) DISPLAY
    PIC  TABL  LOOK  NONZ   SHOC  SOLU   RANGE   BPIC
4) FILE OUTPUT
    SAVE   DIVE  RVRT SDBC
5) SOLUTION
    GO  PIV
6) PROBLEM EDITTING
    ALT   EXT  DEL  SUB  APPC
7) QUIT
8) INTEGER QUADRATIC, AND PARAMETRIC PROGRAMS
9) CONVERSATIONAL PARAMETERS
10) USER SUPPLIED ROUTINES
11) MISCELLANEOUS
    TNU   STAT  BUG
```


ANEXO C
X2
30
25
20
15
10
SOLUCION
(10,15)
5 10 15 20 60
X1

B I B L I O G R A F I A

1.- PROGRAMACION LINEAL. Base teorica y aplicaciones administrativas. Luis Peñafiel Millán. Editorial TRILLAS. México 1985.

2.- INVESTIGACION DE OPERACIONES. Victor Ríos García. Editorial Publicaciones del Instituto Politécnico Nacional, México DF. 1983.

3.- PROGRAMACION LINEAL. Hector M. Espinosa Berriel. Editorial Pax-México. México 1977.

4.- METODOS Y MODELOS DE INVESTIGACION DE OPERACIONES. volumen I. Juan Prawda. Editorial Limusa, México 1976.

5.- FUNDAMENTOS DE INVESTIGACION DE OPERACIONES. Russel L. Ackoff y Maurice W. Sasieni. Editorial Limusa, México 1986.

6.- OPTIMIZACION SISTEMAS ESTATICOS. DR. Meinardo A. Boizán. Editorial Oriente. Santiago de Cuba. 1984.a

7.- METODOS Y MODELOS DE LA INVESTIGACION DE OPERACIONES. Kaufmann. Compañia Editorial Continental S.A. México 1989.

8.- TEORIA ECONOMICA Y ANALISIS DE OPERACIONES. Baumol W. Editorial Herrero Hermanos. S.A. México 1969.

9.- ECONOMIA INTERINDUSTRIAL: Insumo, producto y programación lineal. Chennery Hollis, B. y Clark Paul. Editorial Fondo de Cultura Económica. México 1988.

10.- MANUAL DE TURBO PASCAL VERSION 5.5. Steve Wood. Editorial Osborne/McGraw-Hill 1989.

11.- INTRODUCCION A LA PROGRAMACION LINEAL APLICADA A LA AGRICULTURA. Lara Ixmukane Calderón, Marcos Portillo Vázquez. Editorial Universidad Autónoma Chapingo, Departamento de Economía Agrícola 1987.

12.- INTRODUCCION A LA PROGRAMACION LINEAL Y AL ANALISIS DE SENSIBILIDAD. Graciela Bueno de Arjona. Editorial Trillas. México 1987.

13.- PROGRAMACION LINEAL (METODOS Y APLICACIONES). Saul I. Gass. Edición Revolucionaria Instituto Cubano Del Libro, Cuba 1971.

14.- INVESTIGACION DE OPERACIONES (Una Introducción). Hamdy A. Taha. Editorial Representaciones y Servicios de Ingenieria. S.A. México 1982.

15.- OPTIMIZACION (Sistemas Estáticos). Dr. Meinardo A. Boizán. Editorial Oriente. Santiago De Cuba. 1984.

16.- MANUAL DEL PASANTE (PARA OBTENER SU TITULO), C.P. César Calvo Langarica. Editorial Pax. México 1985. 2a. Edición.

17.- ALGEBRA LINEAL Y SUS APLICACIONES. Gilbert Strang. Editorial Fondo Educativo Interamericano. México 1982.

18.- INGENIERIA DE SISTEMAS. Victor Flores Zavala. Editorial Facultad de Ingeniería de la Universidad Nacional Autónoma de México. 1982.

19.- MATEMATICAS PARA ECONOMISTAS. Edward T. Dowling. Editorial McGRAW-HILL. México 1982.

20.- APLICACION E INTERPRETACION ECONOMICA DEL DUAL DE LA PROGRAMACION LINEAL. LOS PROBLEMAS DE TRANSPORTES, DIETA Y PRODUCCION. Florinda Carlos Garcia. Editorial Colegio de Postgraduados. México 1987.

21.- PROGRAMACION LINEAL. Hector M. Espinosa Berriel. Editorial Pax-México. México. D.F.

22.- APUNTES DE PROGRAMACION LINEAL. Luis E. Chalita Tovar, Daniel Barrera Islas. Editorial Chapingo, México.