

PROGRAMACIÓN ENTERA Y BINARIA PARA MAXIMIZACIÓN DE COMBOS DE ARTÍCULOS DE HOGAR Y CONSTRUCCIÓN EN TODO DECO

Matías González Virgili - matiagonzalez@itba.edu.ar

Paula González - paulgonzalez@itba.edu.ar

Instituto Tecnológico de Buenos Aires

Resumen

En este estudio, se aborda la optimización de la disponibilidad de combos promocionales en una sucursal de la empresa TODO DECO que brinda artículo del hogar y construcción ubicada en el interior del país. Se examinan diferentes combinaciones de productos teniendo en cuenta las limitaciones de espacio de almacenamiento en la sucursal y la frecuencia de reposición de los productos. El objetivo principal es asegurar que los clientes tengan la oportunidad de adquirir los combos en promoción que deseen al visitar la sucursal, maximizando así la satisfacción del cliente y la eficiencia en la gestión de inventario.

Palabras clave

Programación Lineal, Stock, Modelo Matemático, Optimización, Maximización, Combos, Variables Enteras, Variables Binarias, LINGO.

Abstract

In this study, we address the optimization of the availability of promotional bundles at a branch of the TODO DECO company, which provides home and construction products and is located in the interior of the country. We examine various combinations of products, taking into account the limitations of storage space at the branch and the replenishment frequency of products. The main objective is to ensure that customers have the opportunity to acquire the promotional bundles they desire when visiting the branch, thereby maximizing customer satisfaction and inventory management efficiency.

Key words

Linear Programming, Inventory, Mathematical Model, Optimization, Maximization, Combos, Integer Variables, Binary Variables, LINGO.

1. Introducción

La empresa TODO DECO tiene como objetivo la implementación de una nueva línea de productos y electrodomésticos para la cocina. Estos serán comercializados en forma de combos, los cuales estarán disponibles para la venta únicamente si todos sus productos están en stock.

Baldosas B1 - Blanco. B2 - Marfil. B3 - Blanco y azul a cuadros. B4 - Blanco y amarillo a cuadros.	Empapelado Vinílico E1 – Símil marfil blanco. E2 – Símil marfil a rayas celestes. E3 –Símil mármol azul. E4 - Símil mármol amarillo claro.	Aplicques de Luz L1 - Plafón único rectangular. L2 - Plafones led ovalados. L3 - Bombillas de filamentos. L4 - Globos de luz fría.	Alacenas A1 - Madera clara. A2 - Madera oscura. A3 - Madera clara con puertas traslúcidas. A4 - Madera oscura con puertas traslúcidas.
Mesadas M1 - Madera laqueada M2 - Cemento alisado. M3 - Mármol sintético oscuro. M4 - Granito.	Bacha y grifería G1 - Bacha dividida con grifo mono-comando. G2 - Bacha dividida con grifos separados. G3 - Bacha única con grifo mono-comando. G4 - Bacha única con grifos separados.	Lavavajillas W1 - Blanco. W2 - Gris.	Cocinas C1 - Cocina eléctrica blanca. C2 - Cocina eléctrica negra. C3 - Cocina a gas blanca. C4 - Cocina a gas negra.

Figura 1.1: Listado de productos de cocina

La Figura 1.1 exhibe los elementos de cocina disponibles para su comercialización, los cuales se pueden categorizar en ocho tipos distintos: baldosas, empapelado vinílico, apliques de luz, alacenas, mesadas, bachas y grifería, lavavajillas y cocinas. En su conjunto, se presenta una diversidad de treinta productos.

Combo 1	B2	E2	L4	A2	M4	G2	W2	C2
Combo 2	B1	E1	L1	A4	M4	G4	W1	C2
Combo 3	B1	E2	L2	A1	M1	G4	W1	C3
Combo 4	B3	E3	L3	A3	M3	G1	W1	C1
Combo 5	B4	E4	L1	A2	M2	G2	W1	C1
Combo 6	B2	E2	L2	A4	M4	G3	W2	C4
Combo 7	B1	E3	L4	A3	M2	G1	W1	C1
Combo 8	B2	E1	L3	A1	M1	G3	W2	C4
Combo 9	B4	E1	L2	A3	M2	G2	W2	C2
Combo 10	B1	E4	L1	A1	M3	G4	W1	C3
Combo 11	B3	E4	L3	A3	M1	G1	W1	C3
Combo 12	B2	L1	A2	M2	G4	W2	C2	
Combo 13	B4	L3	A3	M1	G2	W1	C3	
Combo 14	B4	E1	L4	A1	M3	G1	C1	
Combo 15	B3	E2	L1	A1	M1	G3	C3	
Combo 16	B3	E3	L4	A1	M3	G2	C1	
Combo 17	B1	E3	L2	A3	M3	G4	C3	
Combo 18	B2	E3	L3	A2	M4	G1	C2	
Combo 19	B2	E2	L4	A4	M4	G2	C4	
Combo 20	B2	E2	L1	A1	M2	G3	C4	

Figura 1.2: Listado de combos de cocina.

Por otro lado, la Figura 1.2 dispone las combinaciones posibles de los productos previamente mencionados. Hay 11 combos de 8 productos y 9 de 7 productos. En su totalidad, se generan veinte combinaciones distintas.

En este paper, se persigue la meta de proporcionar la máxima cantidad de combinaciones posibles de conjuntos de productos, tomando en consideración la capacidad de almacenamiento de la tienda. Para lograrlo, se plantea un modelo de programación que involucra tanto variables enteras (representando la cantidad de productos) como variables binarias (indicando la disponibilidad de los conjuntos) utilizando LINGO, un software especializado en optimización y programación.

En la Figura 1.3 subsecuente, se muestra el diagrama de proceso que representa la ensambladura de los conjuntos de productos. El propósito subyacente de este diagrama es proporcionar una comprensión detallada de cómo los conjuntos de productos se constituyen a partir de los diversos artículos disponibles en la venta.

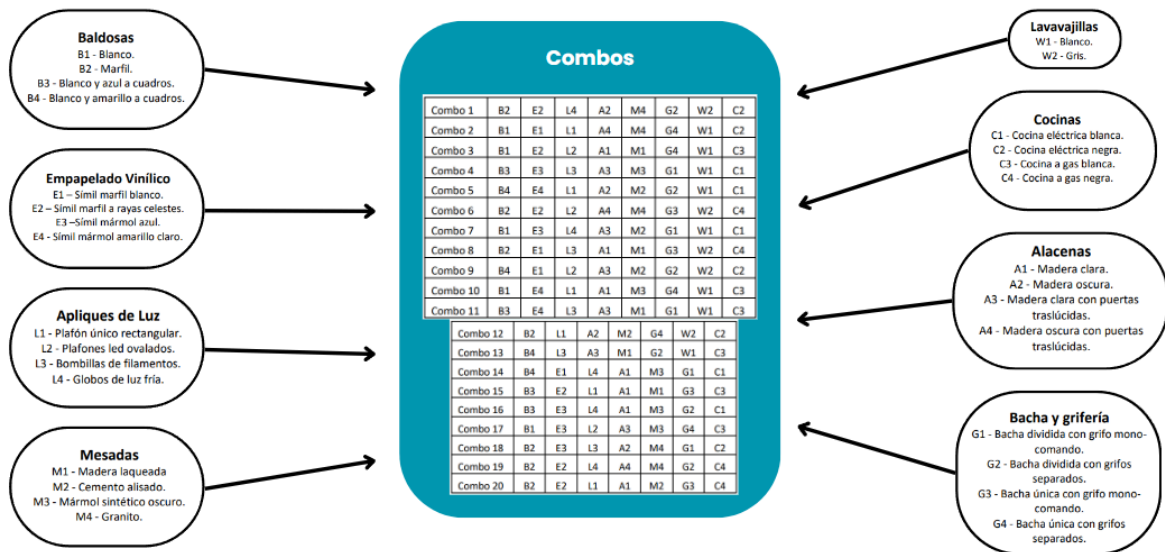


Figura 1.3: Diagrama de proceso de los combos

2. Desarrollo del caso de estudio

En esta sección se desarrollará el modelo matemático para poder encontrar la forma de maximizar la cantidad de combos en venta.

2.1. Variables de decisión y datos

Las variables del modelo abarcan los artículos que se venden en la tienda, los combos y una tercera variable que representa el producto utilizado en un combo específico.

- $X_j, \forall j \leq 30$: Cantidad de artículos de cada tipo en el almacén (variable entera).
donde j = número del producto
- $C_i, \forall i \leq 20$: Combos disponibles/no disponibles (variable binaria):
 $1 = disponible$; $0 = no disponible$
donde i = número de combo
- $M_{ij}, \forall i \leq 20 \wedge \forall j \leq 30$: Artículo j utilizando en el combo i .

2.2. Restricciones

Se definieron seis restricciones principales que se detallan a continuación.

- El local tiene un límite de 5 baldosas:

$$\sum_{i=1}^4 X_i \leq 5$$

- Se pueden almacenar hasta 8 juegos de empapelado vinílico:

$$\sum_{i=5}^8 Xi \leq 8$$

- c. Se pueden almacenar hasta 4 apliques de luz:

$$\sum_{i=9}^{12} Xi \leq 4$$

- d. Se puede almacenar hasta 4 juegos de alacenas:

$$\sum_{i=13}^{16} Xi \leq 4$$

- e. El local puede almacenar hasta 3 mesadas:

$$\sum_{i=17}^{20} Xi \leq 3$$

- f. El local puede almacenar hasta 4 juegos de bacha y grifería:

$$\sum_{i=21}^{24} Xi \leq 4$$

- g. Los lavavajillas y las cocinas tienen el mismo tamaño, así que pueden almacenarse juntos, con un máximo de 5 unidades en total:

$$\sum_{i=25}^{30} Xi \leq 5$$

- h. La próxima restricción se hizo para que, en la matriz de combos, un producto no figure en más veces de los que hay en stock:

$$\forall j \leq 30, Xj \geq \sum_{i=1}^{20} Ci * Mij$$

- i. La cantidad de artículos es un número positivo:

$$Xj \geq 0$$

2.3. Función objetivo

El objetivo consiste en maximizar la variedad de combos disponibles para la venta de TODO DECO, denotada como "Z". Esta magnitud se calcula mediante el siguiente procedimiento.

$$Z = \sum_{i=1}^{20} Ci$$

3. Resultados

En esta sección, se presentarán los resultados derivados de los problemas de maximización de la variedad de combos, considerando las condiciones específicas de cada caso en particular. Para abordar esta cuestión, se procedió a resolver el modelo matemático previamente formulado en la sección 2 mediante el uso del software LINGO.

3.1. Cantidad de combos bajo las condiciones iniciales

En este primer escenario, se consideró un sistema de reposición mensual de los artículos, lo que motivó la búsqueda de una estrategia que permitiera determinar la selección óptima de artículos y su cantidad correspondiente con el fin de maximizar la diversidad de los combos, manteniendo en consideración las restricciones establecidas.

Al ejecutar el modelo, se ha llegado a la conclusión de que, considerando las restricciones y limitaciones inherentes al negocio, la máxima cantidad de combos que la empresa puede ofrecer asciende a cuatro, específicamente los identificados como "combo11", "combo13", "combo14" y "combo20". Es importante destacar que se ha determinado la cantidad óptima de productos de cada tipo que debe mantenerse en inventario para maximizar la diversidad de los combos, como se detalla en la *Tabla 3.3.1* que se presenta a continuación.

Producto	Cantidad
B1	2
B2	1
B3	1
B4	1
E1	1
E2	1
E3	2
E4	1
L1	2
L2	0
L3	0
L4	2
A1	0
A2	1

A3	1
A4	2
M1	0
M2	1
M3	1
M4	1
G1	1
G2	1
G3	1
G4	1
W1	0
W2	0
C1	1
C2	1
C3	0
C4	3

Tabla 3.3.1: Productos a tener en stock para el caso 3.1

3.2. Cantidad de combos con reposición de productos automática

En este contexto, se plantea que la empresa TODODECO realiza un nuevo contrato con proveedores de la zona para que puedan disminuir considerablemente los tiempos de los envíos. Bajo este nuevo escenario, se propone un enfoque diferente en el cual, en lugar de realizar reposiciones mensuales de productos, estos se reponen de manera automática una vez que son vendidos, siguiendo un funcionamiento similar al de una máquina expendedora. En consecuencia, ya no es necesario mantener múltiples unidades de un mismo producto en el almacenamiento, evitando así un uso ineficiente del espacio disponible, como sería el caso de una máquina expendedora que ocupará varios espacios con el mismo producto, limitando la variedad de productos ofrecidos.

Este cambio de enfoque implica ajustes menores en relación al modelo original. Dado que ahora el interés no radica en la cantidad específica de un producto, sino en su disponibilidad o

no, la variable X_j se redefine como una variable binaria para reflejar esta nueva consideración:

- $X_j, \forall j \leq 30$: Artículos disponibles/no disponibles en el almacén (variable binaria).
donde j = número del producto.

En adición, el enfoque se ha desplazado de la consideración de si la cantidad de productos en stock coincide con su frecuencia de aparición en la matriz de combos a una evaluación de la capacidad para ensamblar un combo utilizando los productos actualmente disponibles. Es relevante observar que los primeros once combos consisten en la combinación de ocho productos, mientras que los subsiguientes se componen de siete elementos. Como resultado de este cambio de perspectiva, se introducen restricciones adicionales que se detallan a continuación:

- a) Determinamos que si un combo de 8 artículos existe, entonces todos sus productos deben estar disponibles:

$$\forall i \leq 11, C_i * 8 = j = \sum_{i=1}^{30} X_j * M_i$$

- b) Determinamos que si un combo de 7 artículos existe, entonces todos sus productos deben estar disponibles:

$$\forall i \geq 12 \wedge \forall i \leq 20, C_i * 7 = j = \sum_{i=1}^{30} X_j * M_i$$

Al ejecutar el modelo, se ha llegado a la conclusión de que, considerando las restricciones y limitaciones inherentes al negocio, la máxima cantidad de combos que la empresa puede ofrecer asciende a 13, específicamente los identificados como "combo1", "combo2", "combo3", "combo6", "combo9", "combo12", "combo13", "combo15", "combo17" y "combo18". Y en cuanto a los artículos disponibles para el armado de los combos, la mayoría están disponibles exceptuando los artículos de mármol sintético oscuro (M3), lavavajillas gris (W2) y cocina a gas blanca (C3).

Se puede inferir que la formalización de acuerdos contractuales con los proveedores regionales, orientados a la significativa reducción de los tiempos de entrega, incide de manera positiva en la ampliación de la diversidad de combos que se encuentran disponibles para su oferta junto a los productos de la Sección Cocina de TODODECO.

3.3. Cantidad de combos con ampliación de almacenamiento.

En esta sección, se plantea la perspectiva de expandir la capacidad de almacenamiento destinada a alacenas y mesas, contemplando la posibilidad de mantener hasta doce artículos en conjunto para ambos tipos de productos. Con el propósito de evaluar de manera rigurosa el efecto de esta modificación, se llevará a cabo una modificación de la restricción

correspondiente, considerando las implicaciones de este cambio en el modelo que se puede observar a continuación.

$$\sum_{i=13}^{20} X_i \leq 12$$

Al ejecutar el programa, la cantidad de combos siguen siendo los mismos que en el inciso anterior: "combo1", "combo2", "combo3", "combo6", "combo9", "combo12", "combo13", "combo15", "combo17" y "combo18". En este caso, a diferencia del inciso anterior, los productos que no se ofrecen en este caso son el lavavajillas gris (W2) y la cocina a gas blanca (C3).

4. Conclusiones

Tras un detenido análisis de las secciones previamente presentadas, partiendo del modelo original, se advierte que la empresa TODODECO puede generar un número máximo de cuatro combos. No obstante, es plausible argumentar en favor de la inversión en un nuevo acuerdo contractual con proveedores locales, con el propósito de substancialmente reducir los tiempos de entrega, lo que resultaría en un aumento significativo en la diversidad de combos máximos ofrecidos, alcanzando un total de trece.

Por último, se considera que la reconfiguración física del almacenamiento del negocio, específicamente en relación a la reubicación de las mesadas y las alacenas, no conlleva a una alteración en la cantidad máxima de combos que pueden ser ofrecidos, manteniéndose esta cifra en trece como máximo. Es por eso que no recomendamos llevar a cabo esta última medida.

5. Bibliografía

Hieller, F. S. & Lieberman, G. J. (2010). Introducción a la programación lineal. Introducción a la Investigación de Operaciones. México: MCGRAW-HILL/INTERAMERICANA EDITORES, S.A. DE C.V

6. Anexo

a) Código de LINGO para los resultados de la seccion 3.1

SETS:

PRODUCTOS/1..30/:X;

COMBOS/1..20/:C;

PAR(PRODUCTOS,COMBOS):MATRIZ;

ENDSETS

DATA:


```

MATRIZ =
010001000000101000000101000010100
1000100010000000100010001100100
1000010001001000100000001100010
001000100010001000101000101000
0001000110000010001000100101000
0100010001000000100010010010001
1000001000001001001001000101000
01001000000101000100000010010001
00011000001000001001000100010100
10000001100010000001000001100010
0010000010010001010001000100010
010000000100000100010000001010100
00010000000100010100000100100010
00011000000011000000101000001000
00100100100010001000000100000010
00100010000011000000100100001000
10000010010000010001000001000010
01000010001001000000110000000100
01000100000010001000101000000001
01000100100010000010000010000001
;
ENDDATA

```

!DEFINICIÓN DE FUNCION OBJETIVO PARA MAXIMIZAR;
MAX = @SUM(COMBOS(i):C(i));

!RESTRICCIONES ENTERO/BINARIO;
@FOR(PRODUCTOS(j):@GIN(X(j)));
@FOR(COMBOS(i):@BIN(C(i)));

!RESTRICCION MATRIZ;
@FOR(PRODUCTOS(j):X(j)>=@SUM(COMBOS(i):C(i)*MATRIZ(j,i)));

!RESTRICCIONES ALMACENAMIENTO;
!BALDOSAS;
@SUM(PRODUCTOS(j)|j#LE#4: X(j)) <= 5;
!PAPEL VINILICO;
@SUM(PRODUCTOS(j)|j#GT#4 #AND# j#LE#8: X(j)) <= 8;
!APLIQUES DE LUZ;
@SUM(PRODUCTOS(j)|j#GT#8 #AND# j#LE#12: X(j)) <= 4;
!ALACENAS;
@SUM(PRODUCTOS(j)|j#GT#12 #AND# j#LE#16: X(j)) <= 4;
!MESADAS;

```
@SUM(PRODUCTOS(j)|j#GT#16 #AND# j#LE#20: X(j)) <= 3;
!BACHAS Y GRIFERIAS;
@SUM(PRODUCTOS(j)|j#GT#20 #AND# j#LE#24: X(j)) <= 4;
!LAVAVAJILLAS Y COCINAS;
@SUM(PRODUCTOS(j)|j#GT#24: X(j)) <= 5;

!POSITIVIDAD;
@FOR(PRODUCTOS(j):X(j)>=0);
```

b) Código de LINGO para los resultados de la seccion 3.2

```
SETS:
PRODUCTOS/1..30/:X;
COMBOS/1..20/:C;
PAR(PRODUCTOS,COMBOS):MATRIZ;
ENDSETS

DATA:
MATRIZ =
0 1 0 0 0 1 0 0 0 0 0 1 0 1 0 0 0 0 0 1 0 1 0 0 0 1 0 1 0 0
1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 1 0 0 0 1 0 0 0 1 1 0 0 1 0 0
1 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 1 0 0 0 0 0 0 1 1 0 0 0 1 0
0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 1 0 0 0 1 0 1 0 0 0
0 0 0 1 0 0 0 1 1 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 1 0 1 0 0 0
0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 0 1 0 0 1 0 0 0 1
1 0 0 0 0 0 1 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0 1 0 1 0 0 0
0 1 0 0 1 0 0 0 0 0 1 0 1 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 0 1
0 0 0 1 1 0 0 0 0 1 0 0 0 0 1 0 0 1 0 0 0 1 0 0 0 1 0 1 0 0
1 0 0 0 0 0 0 1 1 0 0 0 1 0 0 0 0 0 1 0 0 0 0 1 1 0 0 0 1 0
0 0 1 0 0 0 0 1 0 0 1 0 0 0 1 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0
0 1 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 0 0 1 0 1 0 1 0 0
0 0 0 1 0 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 0 1 0 0 1 0 0 0 1 0
0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 1 0 1 0 0 0 0 0 1 0 0 0
0 0 1 0 0 1 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 1 0
0 0 1 0 0 0 1 0 0 0 0 1 1 0 0 0 0 0 1 0 0 1 0 0 0 0 1 0 0 0
1 0 0 0 0 0 1 0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 0 0 1 0 0 0 0 1 0
0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 1 0 0
0 1 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 1 0 1 0 0 0 0 0 0 0 1
0 1 0 0 0 1 0 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 1
;
ENDDATA
```

```
!DEFINICIÓN DE FUNCION OBJETIVO PARA MAXIMIZAR;
MAX = @SUM(COMBOS(i):C(i));
```

!RESTRICCIONES ENTERO/BINARIO;

@FOR(PRODUCTOS(j):@BIN(X(j)));

@FOR(COMBOS(i):@BIN(C(i)));

!RESTRICCION MATRIZ;

@for(COMBOS(i)|i#LE#11:@sum(PRODUCTOS(j): X(j)*MATRIZ(j,i)) >= 8*C(i));

@for(COMBOS(i)|i#GE#12:@sum(PRODUCTOS(j): X(j)*MATRIZ(j,i)) >= 7*C(i));

!RESTRICCIONES ALMACENAMIENTO;

!BALDOSAS;

@SUM(PRODUCTOS(j)|j#LE#4: X(j)) <= 5;

!PAPEL VINILICO;

@SUM(PRODUCTOS(j)|j#GT#4 #AND# j#LE#8: X(j)) <= 8;

!APLIQUES DE LUZ;

@SUM(PRODUCTOS(j)|j#GT#8 #AND# j#LE#12: X(j)) <= 4;

!ALACENAS;

@SUM(PRODUCTOS(j)|j#GT#12 #AND# j#LE#16: X(j)) <= 4;

!MESADAS;

@SUM(PRODUCTOS(j)|j#GT#16 #AND# j#LE#20: X(j)) <= 3;

!BACHAS Y GRIFERIAS;

@SUM(PRODUCTOS(j)|j#GT#20 #AND# j#LE#24: X(j)) <= 4;

!LAVAVAJILLAS Y COCINAS;

@SUM(PRODUCTOS(j)|j#GT#24: X(j)) <= 5;

!POSITIVIDAD;

@FOR(PRODUCTOS(j):X(j)>=0);

c) Código de LINGO para los resultados de la seccion 3.3

SETS:

PRODUCTOS/1..30/:X;

COMBOS/1..20/:C;

PAR(PRODUCTOS,COMBOS):MATRIZ;

ENDSETS

DATA:

MATRIZ =

```
0 1 0 0 0 1 0 0 0 0 0 1 0 1 0 0 0 0 0 1 0 1 0 0 0 1 0 1 0 0
1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 1 0 0 0 1 0 0 0 1 1 0 0 1 0 0
1 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 1 0 0 0 0 0 0 1 1 0 0 0 1 0
0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 1 0 0 0 1 0 1 0 0 0
0 0 0 1 0 0 0 1 1 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 1 0 1 0 0 0
0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 0 1 0 0 1 0 0 0 1
```

```

1 0 0 0 0 0 1 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0 1 0 1 0 0 0
0 1 0 0 1 0 0 0 0 0 1 0 1 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 0 1
0 0 0 1 1 0 0 0 0 1 0 0 0 0 1 0 0 1 0 0 0 1 0 0 0 1 0 1 0 0
1 0 0 0 0 0 0 1 1 0 0 0 1 0 0 0 0 0 1 0 0 0 0 1 1 0 0 0 1 0
0 0 1 0 0 0 0 1 0 0 1 0 0 0 1 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0
0 1 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 0 0 1 0 1 0 1 0 0
0 0 0 1 0 0 0 0 0 0 1 0 0 0 1 0 1 0 0 0 0 1 0 0 1 0 0 0 1 0
0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 1 0 1 0 0 0 0 0 1 0 0 0
0 0 1 0 0 1 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 1 0
0 0 1 0 0 0 1 0 0 0 0 1 1 0 0 0 0 0 1 0 0 1 0 0 0 0 1 0 0 0
1 0 0 0 0 0 1 0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0
0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 1 0 0
0 1 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 1 0 1 0 0 0 0 0 0 0 1
0 1 0 0 0 1 0 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1

```

```

;
ENDDATA

```

```

!DEFINICIÓN DE FUNCION OBJETIVO PARA MAXIMIZAR;
MAX = @SUM(COMBOS(i):C(i));

```

```

!RESTRICCIONES ENTERO/BINARIO;
@FOR(PRODUCTOS(j):@BIN(X(j)));
@FOR(COMBOS(i):@BIN(C(i)));

```

```

!RESTRICCION MATRIZ;
@for(COMBOS(i)|i#LE#11:@sum(PRODUCTOS(j): X(j)*MATRIZ(j,i)) >= 8*C(i));
@for(COMBOS(i)|i#GE#12:@sum(PRODUCTOS(j): X(j)*MATRIZ(j,i)) >= 7*C(i));

```

```

!RESTRICCIONES ALMACENAMIENTO;
!BALDOSAS;
@SUM(PRODUCTOS(j)|j#LE#4: X(j)) <= 5;
!PAPEL VINILICO;
@SUM(PRODUCTOS(j)|j#GT#4 #AND# j#LE#8: X(j)) <= 8;
!APLIQUES DE LUZ;
@SUM(PRODUCTOS(j)|j#GT#8 #AND# j#LE#12: X(j)) <= 4;
!ALACENAS Y ALACENAS;
@SUM(PRODUCTOS(j)|j#GT#12 #AND# j#LE#20: X(j)) <= 12;

```

```

!BACHAS Y GRIFERIAS;
@SUM(PRODUCTOS(j)|j#GT#20 #AND# j#LE#24: X(j)) <= 4;
!LAVAVAJILLAS Y COCINAS;
@SUM(PRODUCTOS(j)|j#GT#24: X(j)) <= 5;

```

```

!POSITIVIDAD;

```

@FOR(PRODUCTOS(j):X(j)>=0);