

!En Lingo, para que indicar que la variable X es de naturaleza discreta (entera): @GIN(X)
En Lingo, para indicar cotas sobre la variable X: @BND(limite_inferior,X,limite_superior);

SETS:

!Definición de sets primitivos;

Periodo /Ene, Feb, Mar, Abr, May, Jun/ ;

Maquina /Cotadora, FresadoraV, FresadoraH, Taladro, Alisadora/ ;

Producto /P1, P2, P3, P4, P5, P6, P7/: Beneficio;

!Fabricación de sets derivados;

Fabricacion (Periodo, Producto): UProducidas, UVendidas, UStock, MaxStock, CosteStock;

CapMaquinas (Periodo, Maquina): HrsDispMaquina, MaquinasDisp, MaquinasMant;

HrsProduccion (Maquina, Producto): HrsNecesarias;

ENDSETS

DATA:

MaquinasDisp= 4 2 3 1 1
 4 2 3 1 1
 4 2 3 1 1
 4 2 3 1 1
 4 2 3 1 1
 4 2 3 1 1;

MaquinasMant= 1 0 0 0 0
 0 0 2 0 0
 0 0 0 1 0
 0 1 0 0 0
 1 1 0 0 0
 0 0 1 0 1;

Beneficio = 10 6 8 4 11 9 3;

HrsNecesarias= 0.5 0.7 0 0 0.3 0.2 0.5
 0.1 0.2 0 0.3 0 0.6 0

```

0.2  0    0.8  0    0    0    0.6
0.05 0.03 0    0.07 0.1  0    0.08
0    0    0.01 0    0.05 0    0.05;

MaxStock= 100  100  100  100  100  100  100
          100  100  100  100  100  100  100
          100  100  100  100  100  100  100
          100  100  100  100  100  100  100
          100  100  100  100  100  100  100
          100  100  100  100  100  100  100;

CosteStock= 0.5  0.5  0.5  0.5  0.5  0.5  0.5
            0.5  0.5  0.5  0.5  0.5  0.5  0.5
            0.5  0.5  0.5  0.5  0.5  0.5  0.5
            0.5  0.5  0.5  0.5  0.5  0.5  0.5
            0.5  0.5  0.5  0.5  0.5  0.5  0.5
            0.5  0.5  0.5  0.5  0.5  0.5  0.5;

HrsDispMaquina= 384  384  384  384  384
                 384  384  384  384  384
                 384  384  384  384  384
                 384  384  384  384  384
                 384  384  384  384  384
                 384  384  384  384  384;

ENDDATA

!Maximizar el beneficio total;
!Sumamos todos los periodos para todos los productos      Beneficios_Totales - Gastos_almacenamiento_totales;
[Beneficio_Total]MAX = @SUM(Fabricacion(per,prod): (Beneficio(prod) * UVendidas(per,prod) - CosteStock(per,prod) * UStock(per,p

!Restricciones de equilibrio;
@FOR(Fabricacion(per,prod):
! Uds. Vendidas mes t + Uds. Stock mes t = Uds. Producidas mes t + Uds. Stock mes (t-1);
! en el primer mes tenemos un stock de 0;
[Equilibrio]UVendidas(per,prod) + UStock(per,prod) = UProducidas(per,prod) + @IF(per#EQ#1,0,UStock(per-1,prod));

```

```

! OPCION 2 EQUILIBRIO
    @FOR(Fabricacion(per,prod) | per #EQ# @INDEX(Periodo,Ene) : UVendidas(per,prod) + UStock(per,prod) = UProducidas(per,pr
    @FOR(Fabricacion(per,prod) | per #NE# @INDEX(Periodo,Ene) : UVendidas(per,prod) + UStock(per,prod) = UProducidas(per,pr
!Capacidad de almacenamiento;
! En cada periodo se almacena menos que el maximo ;
[Almacenamiento]UStock(per,prod) <= MaxStock(per,prod);
);

!Para cada máquina,en cada periodo las horas utilizadas deben ser <= horas disponibles;
! Horas produccion < = num_maquinas_de_ese_tipo_en_ese_periodo * horas_disponibles;
@FOR(CapMaquinas(per,maq):
!Capacidad de producción y disponibilidad de Máquina;
[Produccion]@SUM(Producto(a): HrsNecesarias(maq,a) * UProducidas(per,a)) <= HrsDispMaquina(per,maq) * (MaquinasDisp(per,maq) -
);

@FOR(Producto(z):
!Existencias en el mes de Junio;
! ¿ poner >= para dar mayor flexibilidad ?;
[Existencias]UStock(6,z) = 50;
);

```