
```

%Leidy Vargas M
%15/05/2019
%Enunciadol
%Reddy Mikks produce pinturas para interiores y Exteriores,M1 y M2.Con
una
%demanda diaria de pintura para interiores no mayor a 1 tonelada más
que la
%pintura para exteriores.También que la demanda máxima diaria de
pintura
%para interiores es de 2 toneladas.
%      Pintur ext./Pintura Int/Disponibilidad
%M1      6      4      24
%M2      1      2      6
%Utilidad 5      4
%Determinar la mezcla óptima de productos para exteriores e interiores
que
%maximice la utilidad diaria total.
%x1:toneladas de pintura para exteriores X2:toneladas de pintura para
interiores

                                %Max z= 5 X1+4 X2
                                %s.a: 6X1+4X2<=24
                                %      X1+2X2<=6
                                %      -X1+X2<=1
                                %      x1, x2 #0

%A=[5 4]; %Se define A como funcion objetivo

%B=[ 6 4; %%Se define B como restricciones
% 1 2;
% -1 1];

%C=[ 24; %% vector de recursos de las desigualdades
% 6;
% 1];

function simplexAplicacion1()
clc;
clear all;
A=[5 4];

B=[ 6 4;
    1 2;
   -1 1];

C=[ 24;
    6;
    1];

% Ventana para definir la función objetivo
prompt = {'Función objetivo = ', 'max == 1 ', 'Numero de restricciones
= '};
lineno = 1;
title = 'Ingreso de Datos';

```

```

def      = {'A','1','3'};
options.Resize = 'on';
a        = inputdlg(prompt,title,lineno,def,options);
a        = char(a);
[m,n]    = size(a);
cout     = eval(a(1,:)); %Se transforman los valores (cadena de
    caracteres) ingresados en la FO a un vector de enteros
type     = eval(a(2,1)); %Se transforman los valores (cadena de
    caracteres) ingresados en el tipo de FO a un entero
nbr      = eval(a(3,1)); %Se transforman los valores (cadena de
    caracteres) ingresados en el numero de restricciones a un entero
str1     = struct('vari',{},'Type',{});
str2     = struct('var_base',{},'valeur',{});
%=====
% Ventana para definir restricciones
for i=1:nbr %Se definen los tipos de restricciones en orden
    prompt = {strcat('Ingrese el tipo de restricción para la
        condición ',num2str(i),' (<=,>=,=):')};
    title  = 'Ingreso, de Datos';
    def    = {'<='};
    options.Resize = 'on';
    p = inputdlg(prompt,title,lineno,def,options);
    p = char(p);
    opert = p;
    str1(1,i).Type = opert;
end
%=====
% Ventana para definir coeficientes de las restricciones
prompt = {'Ingrese la matriz de restricciones'};
lineno = 1;
title  = 'Ingreso de Datos';
def    = {'B'};
options.Resize='on';
t      = inputdlg(prompt,title,lineno,def,options);
t      = char(t);
sc     = eval(t); %Se transforman los valores (cadena de caracteres)
    ingresados en el campo, a un matriz de enteros
%=====
% Ventana para definir coeficientes de 'b'
prompt = {'Ingrese el vector de valores independientes b'};
lineno = 1;
title  = 'Ingreso de Datos';
def    = {'C'};
options.Resize = 'on';
u      = inputdlg(prompt,title,lineno,def,options);
u      = char(u);
second = eval(u);
%=====

sc1     = []; %Matriz de Variables de holgura
v_a     = zeros(1,length(cout));
v_e     = [];
v_b     = [];
j       = 1;

```

```

%Paso a forma estandar
for i=1:nbr
    n = str1(1,i).Type;
    if n(1)~= '<' && isempty(sc2)
        sc2=zeros(nbr,1);
    end
    switch str1(1,i).Type
        case '<='
            v_e=[v_e second(i)];
            sc1(j,length(v_e))=1;
            v_b=[v_b,second(i)];

    end
    j=j+1;
end
%=====
sc      =[sc,sc1]; %Nueva Matriz de restricciones con variables de
    holgura añadidas
vari     =[];
vari_a   =[];
vari_e   =[];

for i=1:size(sc,2)
    str1(1,i).vari=['x',num2str(i)];
    vari=[vari,str1(1,i).vari,' '];
    if i<length(v_a)
        vari_a=[vari_a,str1(1,i).vari,' '];
    elseif i<=length(v_a)+length(v_e)
        vari_e=[vari_e,str1(1,i).vari,' '];

    end
end
%Primera iteración
x=[v_a,v_e];
Cj=[cout,0.*v_e];
Vb=[];
Q=v_b;
Ci=[];
tabl=[];
for i=1:length(Q)
    tabl=[tabl; ' | '];
    str2(1,i).valeur=Q(i);
    ind=find(x==Q(i));
    str2(1,i).var_base=str1(1,ind).vari;
    Vb=[Vb,str2(1,i).var_base,' '];
    Ci=[Ci,Cj(ind)];
end
Z=sum(Ci.*Q);
for i=1:length(Cj)
    Zj(i)=sum(Ci'.*sc(:,i));
end
Cj_Zj=Cj-Zj;
l=[];

```

```

for i=1:nbr
    if length(str2(1,i).var_base)==2
        l=[1;str2(1,i).var_base, ' '];
    else
        l=[1;str2(1,i).var_base];
    end
end
end
fprintf('\n');
disp('===== Problema en forma estandar
=====');
disp(['Variables : ',vari]);
disp(['          -Variables No Básicas      : ',vari_a]);
disp(['          -Variables Básicas         : ',vari_e]);
disp('=====');
disp(' ');
disp('===== Tabla 0
=====');
disp(['Inicialización de variables : ',vari]);
disp(['          -Variables No Básicas      :
',num2str(v_a)]);
disp(['          -Variables Básicas         :
',num2str(v_e)]);
disp('=====');
disp(' ');
disp(['Cj          : ',num2str(Cj)]);
disp('_____');
disp([tabl,num2str(Ci'),tabl,1,tabl,num2str(Q'),tabl,num2str(sc),tabl]);
disp('_____');
disp(['Zj          : ',num2str(Zj)]);
disp(['Cj-Zj        : ',num2str(Cj-Zj)]);
disp(['Z          : ',num2str(Z)]);
disp('_____');
disp(' ');
%Iteraciones de Simplex
t      = 1;
arret  = 1;
while arret==1
    if type==1
        num=max(Cj_Zj);num=num(1);
        num1=find(Cj_Zj==num);num1=num1(1);
        V_ent=str1(1,num1).vari;
    else
        g=min(Cj_Zj);g=g(1);
        num1=find(Cj_Zj==g);num1=num1(1);
        V_ent=str1(1,num1).vari;
    end
    b=sc(:,num1);
    k=0;d=10000;
    for i=1:length(Q)
        if b(i)>0
            div=Q(i)/b(i);
            if d>div
                d=div;
                k=i;
            end
        end
    end
    [ 'x',num2str(num1)];
end

```

```

        end
    end
end
if k~=0
    num2=k;
else
    disp('No se puede encontrar solución : La solución es
infactible ');
    break;
end
V_sort=str2(1,num2).var_base;
str2(1,num2).var_base=str1(1,num1).vari;
pivot=sc(num2,num1);
Ci(num2)=Cj(num1);
sc(num2,:)=sc(num2,:)./pivot;
Q(num2)=Q(num2)/pivot;
h=size(sc,1);
for i=1:h
    if i~=num2
        Q(i)=Q(i)-sc(i,num1)*Q(num2);
        sc(i,:)=sc(i,:)-sc(i,num1).*sc(num2,:);

    end
end
Z=sum(Ci.*Q);
for i=1:size(sc,2)
    Zj(i)=sum(Ci'.*sc(:,i));
end
Cj_Zj=Cj-Zj;
l=[];V=[];
for i=1:nbr
    if length(str2(1,i).var_base)==2
        l=[l;str2(1,i).var_base, ' '];
        V=[V,l(i,:), ' '];
    else
        l=[l;str2(1,i).var_base];
        V=[V,l(i,:), ' '];
    end
end
Vb      = V;
disp(['===== Tabla ',num2str(t),'
=====']);
disp(['Variable de entrada : ',num2str(V_ent)]);
disp(['Variable de salida : ',num2str(V_sort)]);
disp(['Pivote : ',num2str(pivot)]);
disp(['Variables Básicas : ',num2str(Vb)]);

disp('=====');
disp(' ');
disp(['Cj : ',num2str(Cj)]);

disp('_____');

disp([tabl,num2str(Ci'),tabl,1,tabl,num2str(Q'),tabl,num2str(sc),tabl]);

```

```

disp('_____');
disp(['Zj          : ',num2str(Zj)]);
disp(['Cj-Zj       : ',num2str(Cj-Zj)]);
disp(['Z          : ',num2str(Z)]);

disp('_____');
disp(' ');
disp(' ');
t      = t+1;
if type==1
    a=max(Cj_Zj);a=a(1);
    if a<=0
        break;
    end
else
    a = min(Cj_Zj);a=a(1);
    if a>=0 break;
    end
end
end
p      = num2str(Z);
disp(['Resultado F.O. OPTIMO : ',num2str(Z)]);
disp('=====');
disp('SOLUCIÓN')
disp(['Se deben producir ',num2str(Q(1)),' toneladas diarias de
pintura para exteriores']);
disp(['Se deben producir ',num2str(Q(2)),' toneladas
diarias de pintura para interiores,es decir,aproximadamente:
',num2str(ceil(Q(2)))]);
disp(['Obteniendo una utilidad diaria de:',num2str(Z),' dólares']);
%k      = msgbox( p,'RESULTADO F.O. OPTIMO :')

```

===== Problema en forma estandar

=====

Variables : x1 x2 x3 x4 x5

-Variables No Básicas : x1

-Variables Básicas : x2 x3 x4 x5

=====

===== Tabla 0

=====

Inicialización de variables : x1 x2 x3 x4 x5

-Variables No Básicas : 0 0

-Variables Básicas : 24 6 1

=====

Cj : 5 4 0 0 0

/	0	/	x3	/	24	/	6	4	1	0	0	/
/	0	/	x4	/	6	/	1	2	0	1	0	/
/	0	/	x5	/	1	/	-1	1	0	0	1	/

```

Zj          : 0  0  0  0  0
Cj-Zj       : 5  4  0  0  0
Z           : 0

```

===== Tabla 1

```

=====
Variable de entrada : x1
Variable de salida  : x3
Pivote              : 6
Variables Básicas   : x1  x4  x5
=====

```

```

Cj          : 5  4  0  0  0

```

5 x1	4 1	0.66667	0.16667	0	0
0 x4	2 0	1.3333	-0.16667	1	0
0 x5	5 0	1.6667	0.16667	0	1

```

Zj          : 5      3.3333      0.83333      0
0
Cj-Zj       : 0      0.66667      -0.83333      0
0
Z           : 20

```

===== Tabla 2

```

=====
Variable de entrada : x2
Variable de salida  : x4
Pivote              : 1.3333
Variables Básicas   : x1  x2  x5
=====

```

```

Cj          : 5  4  0  0  0

```

5 x1	3 1	0	0.25	-0.5	0
4 x2	1.5 0	1	-0.125	0.75	0
0 x5	2.5 0	0	0.375	-1.25	1

```

Zj          : 5      4      0.75      0.5
0
Cj-Zj       : 0      0      -0.75      -0.5
0
Z           : 21

```

Resultado F.O. OPTIMO : 21

SOLUCIÓN

Se deben producir 3 toneladas diarias de pintura para exteriores

Se deben producir 1.5 toneladas diarias de pintura para interiores, es decir, aproximadamente: 2

Obteniendo una utilidad diaria de: 21 dólares

Published with MATLAB® R2019a