SHREEYA CHATTERJI (102103447)

Contents

- Convert the problem into standard form
- Convert the problem into standard form
- Convert the problem into standard form
- Convert the problem into standard form

Convert the problem into standard form

Max z = 2x1-3x2+6x3 s.t.x1-3x3>=4 2x1-8x2+3x3<=4 x1+x2>=-7 x1,x2,x3>=0

```
clc
clear all
close all
format short
 % phase 1 input parameteres
 c=[2 -3 6]; %cost of objectivve function
 A = [1 0 -3;2 -8 3;1 1 0];
 B=[4;4;-7];  %RHS of the constraint
 for i = 1:size(B,1)
     if B(i,:)<0
         A(i,:)=-A(i,:);
         B(i)=-B(i);
     end
 end


 % Phase 2 Identify <= or >= types constraint
 Ineqsign=[-1 1 1]; %-1 for greater than sign 1 for less than sign
 % Introduce slack and surplus variable
 s = eye(size(A,1));
 index = find(Ineqsign<0);
 s(index,:) = -s(index,:);

% Phase 4 To write standard form
objfun= array2table(c);
objfun.Properties.VariableNames(1:size(c,2))={'x1','x2','x3'};
Mat=[A s B];
const=array2table(Mat);
const.Properties.VariableNames(1:size(Mat,2))={'x1','x2','x3','s1','s2','s3','sol'};
disp("The cost function:")
objfun
disp("The function in Standard Form:")
const
```

```
The cost function:

objfun =

  1×3 table
```

```
    x1    x2    x3
    __    __    __

    2     -3    6

The function in Standard Form:

const =

  3×7 table

    x1    x2    x3    s1    s2    s3    sol
    __    __    __    __    __    __    ___

     1     0    -3    -1    0     0      4
     2    -8     3     0    1     0      4
    -1    -1     0     0    0     1      7
```

Convert the problem into standard form

Max z = 3x1+5x2 s.t. x1+2x2<=20 x1+x2<=15 x2>=6 x1,x2,x3>=0

```
clc
clear all
close all
format short
 % phase 1 input parameteres
 c=[3 5]; %cost of objectivve function
 A = [1 2 ;1 1;0 1];
 B=[20;15;6];  %RHS of the constraint
 for i = 1:size(B,1)
     if B(i,:)<0
         A(i,:)=-A(i,:);
         B(i)=-B(i);
     end
 end

 % Phase 2 Identify <= or >= types constraint
 Ineqsign=[1 1 -1]; %-1 for greater than sign 1 for less than sign
 % Introduce slack and surplus variable
 s = eye(size(A,1));
 index = find(Ineqsign<0);
 s(index,:) = -s(index,:);

% Phase 4 To write standard form
objfun= array2table(c);
objfun.Properties.VariableNames(1:size(c,2))={'x1','x2'};
Mat=[A s B];
const=array2table(Mat);
const.Properties.VariableNames(1:size(Mat,2))={'x1','x2','s1','s2','s3','sol'};
disp("The cost function:")
objfun
disp("The function in Standard Form:")
const
```

```
The cost function:

objfun =

  1×2 table

    x1    x2
    __    __

    3     5

The function in Standard Form:

const =

  3×6 table

    x1    x2    s1    s2    s3    sol
    __    __    __    __    __    ___

    1     2     1     0      0    20
    1     1     0     1      0    15
    0     1     0     0     -1     6
```

Convert the problem into standard form

Max z = x1-3x2+2x3 s.t. 3x1-x2+2x3<=7 -2x1+4x2<=2 -4x1+3x2+2x3>=4 x1,x2,x3>=0

```
clc
clear all
close all
format short
 % phase 1 input parameteres
 c=[1,-3,2]; %cost of objectivve function
 A = [3,-1,2;-2,4,0;-4,3,2];
 B=[7;2;4];  %RHS of the constraint
 for i = 1:size(B,1)
     if B(i,:)<0
         A(i,:)=-A(i,:);
         B(i)=-B(i);
     end
 end

 % Phase 2 Identify <= or >= types constraint
 Ineqsign=[1,1,-1]; %-1 for greater than sign 1 for less than sign
 % Introduce slack and surplus variable
 s = eye(size(A,1));
 index = find(Ineqsign<0);
 s(index,:) = -s(index,:);

% Phase 4 To write standard form
objfun= array2table(c);
objfun.Properties.VariableNames(1:size(c,2))={'x1','x2','x3'};
Mat=[A s B];
const=array2table(Mat);
```

```
const.Properties.VariableNames(1:size(Mat,2))={'x1','x2','x3','s1','s2','s3','sol'};
disp("The cost function:")
objfun
disp("The function in Standard Form:")
const
```

```
The cost function:

objfun =

  1×3 table

    x1    x2    x3
    __    __    __

    1     -3    2

The function in Standard Form:

const =

  3×7 table

    x1    x2    x3    s1    s2    s3    sol
    __    __    __    __    __    __    ___

     3    -1    2     1     0      0     7
    -2     4    0     0     1      0     2
    -4     3    2     0     0     -1     4
```

Convert the problem into standard form

Min z = 40x1+24x2+12x3 s.t. 20x1+50x2+10x3>=480 8x1+5x2+2x3<=72 4x1+5x2+3x3<=7 x1,x2,x3>=0

```
clc
clear all
close all
format short
 % phase 1 input parameteres
 c=[40,24,12]; %cost of objectivve function
 A = [20,50,10;8,5,2;4,5,3];
 B=[480;72;7];  %RHS of the constraint
 for i = 1:size(B,1)
     if B(i,:)<0
         A(i,:)=-A(i,:);
         B(i)=-B(i);
     end
 end

 % Phase 2 Identify <= or >= types constraint
 Ineqsign=[-1 1 1]; %-1 for greater than sign 1 for less than sign
 % Introduce slack and surplus variable
 s = eye(size(A,1));
 index = find(Ineqsign<0);
```

```
 s(index,:) = -s(index,:);

% Phase 4 To write standard form
objfun= array2table(c);
objfun.Properties.VariableNames(1:size(c,2))={'x1','x2','x3'};
Mat=[A s B];
const=array2table(Mat);
const.Properties.VariableNames(1:size(Mat,2))={'x1','x2','x3','s1','s2','s3','sol'};
disp("The cost function:")
objfun
disp("The function in Standard Form:")
const
```

```
The cost function:

objfun =

  1×3 table

    x1    x2    x3
    __    __    __

    40    24    12

The function in Standard Form:

const =

  3×7 table

    x1    x2    x3    s1    s2    s3    sol
    __    __    __    __    __    __    ___

    20    50    10    -1    0     0     480
     8     5     2     0    1     0      72
     4     5     3     0    0     1       7
```

Published with MATLAB® R2023b