MAT2003 (Optimization Techniques) Lab Assignment-8

KHAN MOHD OWAIS RAZA
20BCD7138

Question-1

```
%% KHAN MOHD OWAIS RAZA (20BCD7138)
%% Optimization Techniques (MAT2003) Lab
%% Lab Assignment-8
Variables={'x_1','x_2','s_2','s_3','A_1','A_2','Sol'};
M = 1000;
Cost = [-3000 -1000 0 0 -M -M 0];
A = [ 1 1 1 0 0 0 10; 2 3 0 1 0 0 20];
s = eye(size(A,1));
BV = [];
for j=1:size(s,2)
for i=1:size(A,2)
if A(:,i)==s(:,j)
BV=[BV i]
end
end
end
B=A(:,BV);
A=inv(B)*A;
zjcj=Cost(BV)*A-Cost;
zcj=[zjcj;A];
SimpTable=array2table(zcj)
SimpTable.Properties.VariableNames(1:size(zcj,2))=Variables
RUN = true;
while RUN
zc = zjcj(:,1:end-1);
if any(zc<0)
fprintf(' The Current BFS is NOT Optimal \n ');
[Entval, pvt_col] = min(zc);
fprintf('Entering Column = %d \n',pvt_col);
sol = A(:,end); Column = A(:,pvt_col);
if all (Column)<=0
fprintf('Solution is UNBOUNDED');
else
Har = find(Column >0);
ratio = inf.*ones(1,length(sol));
ratio(Har)=sol(Har)./Column(Har);
for i=1:size(Column,1)
if Column(i)>0
ratio(i)=sol(i)./Column(i);
else
ratio(i)=inf;
end
end
[MinRatio,pvt_row]=min(ratio);
fprintf(' LEAVING Row = %d \n',pvt_row);
BV(pvt_row)=pvt_col;
```

```
B=A(:,BV); A=inv(B)*A;
zjcj = Cost(BV)*A - Cost;
zcj=[zjcj;A];
TABLE = array2table(zcj);
TABLE.Properties.VariableNames(1:size(zcj,2))=Variables
end
else
RUN = false;
fprintf(' ====== CURRENT BFS IS OPTIMAL====== \n');
end
end
FINAL_BFS = zeros(1,size(A,2));
FINAL_BFS(BV) = A(:,end);
FINAL_BFS(end)= sum(FINAL_BFS.*Cost);
OptimalBFS= array2table(FINAL_BFS);
OptimalBFS.Properties.VariableNames(1:size(OptimalBFS,2))=Variables
```

CURRENT FOLDE

WORKSPACE

Command Window

```
>> OT_Lab8_1

BV =

     3


BV =

     3     4


SimpTable =

  3×7 table

    zcj1    zcj2    zcj3    zcj4    zcj5    zcj6    zcj7
    ____    ____    ____    ____    ____    ____    ____

    3000    1000     0       0      1000    1000      0
       1       1     1       0         0       0     10
       2       3     0       1         0       0     20
SimpTable =

  3×7 table

    x_1     x_2     s_2     s_3     A_1     A_2     Sol
    ____    ____    ___     ___     ____    ____    ___

    3000    1000     0       0      1000    1000     0
       1       1     1       0         0       0    10
       2       3     0       1         0       0    20
 ====== CURRENT BFS IS OPTIMAL======

OptimalBFS =

  1×7 table

    x_1    x_2    s_2    s_3    A_1    A_2    Sol
    ___    ___    ___    ___    ___    ___    ___

     0      0     10     20      0      0      0
```

Question-2

```
%% KHAN MOHD OWAIS RAZA (20BCD7138)
%% Optimization Techniques (MAT2003) Lab
%% Lab Assignment-8
clc
clear all
Variables={'x_1','x_2','s_1','s_2','Sol'};
Cost = [-3 -1 0 0 0 ];
INFO = [1 2;1 1];
b = [10;20];
s=eye(size(INFO,1))
A=[INFO s b];
BV=[];
for j=1:size(s,2)
for i=1:size(A,2)
if A(:,i)==s(:,j)
BV =[BV i];
end
end
end
fprintf('Basic Variables (BV) =')
disp(Variables(BV));
B=A(:,BV);
A= inv(B)*A;
ZjCj=Cost(BV)*A-Cost;
Zcj=[ZjCj;A]
SimpleTable=array2table(Zcj);
SimpleTable.Properties.VariableNames(1:size(Zcj,2))=Variables
RUN= true;
while RUN
SOL=A(:,end);
if any(SOL<0);
fprintf('The current BFS is not FEASIBLE \n')
[LeaVal,pvt_row]=min(SOL);
fprintf('Leaving Row = %d \n',pvt_row);
ROW=A(pvt_row,1:end-1);
ZJ= ZjCj(:,1:end-1);
for i=1:size(ROW,2)
if ROW(i)<0
ratio(i)=abs(ZJ(i)./ROW(i));
else
ratio(i)=inf;
end
end
[minVAL, pvt_col]=min(ratio);
fprintf('Entering Variable = %d \n',pvt_col);
fprintf('Basic Variables (BV) =')
BV(pvt_row) = pvt_col;
disp(Variables(BV));
pvt_key=A(pvt_row,pvt_col);
A(pvt_row,:) = A(pvt_row,:)./pvt_key;
for i=1:size(A,1)
if i~=pvt_row
```

```
1
A(i,:)=A(i,:)-A(i,pvt_col).*A(pvt_row,:);
end
end
ZjCj=Cost(BV)*A-Cost;
Zcj=[ZjCj;A]
SimpleTable=array2table(Zcj);
SimpleTable.Properties.VariableNames(1:size(Zcj,2))=Variables
else
RUN=false;
fprintf('The current BFS is FEASIBLE and OPTIMAL\n')
end
end
```

CURRENT FOLDE

WORKSPACE

Command Window

```
s =

     1     0
     0     1

Basic Variables (BV) =    {'s_1'}    {'s_2'}


Zcj =

     3     1     0     0     0
     1     2     1     0    10
     1     1     0     1    20


SimpleTable =

  3×5 table

    x_1    x_2    s_1    s_2    Sol
    ___    ___    ___    ___    ___

     3      1      0      0      0
     1      2      1      0     10
     1      1      0      1     20

The current BFS is FEASIBLE and OPTIMAL
>>
```

Question-3

```
%% KHAN MOHD OWAIS RAZA (20BCD7138)
%% Optimization Techniques (MAT2003) Lab
%% Lab Assignment-8
clc;
clear all;
arr=[7 5 4 9;5 6 8 10;4 2 8 7;9 11 10 6]
disp('cost matrix');
disp(arr);
b=arr;
for i=1:size(arr,1)
 sub=min(arr(i,:));
 arr(i,:) = arr(i,:)-sub;
end
for i=1:size(arr,2)
 sub=min(arr(:,i));
 arr(:,i) = arr(:,i)-sub;
end
disp('after subtracting row minimum and column minimum');
disp(arr);
while true
 temp=arr;
 lines = 0;
 while true
 minZ=inf;
 for i=1:size(temp,1)
 count=size(find(temp(i,:)==0),2);
 disp('count in row is:');
 disp(count);
 if(count>0 && count < minZ)
 minZ=count;
 d=1;
 y=find(temp(i,:)==0,1);
 disp('y1 is:');
 disp(y);
 end
 end
 for i=1:size(temp,2)
 count=size(find(temp(:,i)==0),1);
 disp('count in col is:');
 disp(count);
 if(count>0 && count < minZ)
 minZ=count;
 d=0;
 y=find(temp(:,i)==0,1);
 disp('y2s is:');
 disp(y);
 end
 disp('y is:');
 disp(y);
 end
 if minZ==inf
 break;
```

```
 end
 if d==1
 temp(:,y)=inf;
 else
 temp(y,:)=inf;
 end
 lines = lines + 1;
 disp('lines is:');
 disp(lines);
 end
 sub = min(min(temp));
 if(lines~=4)
 for i=1:size(arr,1)
 for j=1:size(arr,2)
 if(temp(i,j)~=inf)
 arr(i,j) = arr(i,j)-sub;
 elseif((size(find(temp(i,:)==inf),2)==4) &&
(size(find(temp(:,j)==inf),1)==4))
 arr(i,j) = arr(i,j)+sub;
 end
 end
 end
 end
 if(lines==4)
 break;
 end
end
disp('Modified cost matrix');
disp(arr);
totalc=0;
for i=1:size(arr,1)
 for j=1:size(arr,2)
 if(arr(i,j)==0)
 totalc=totalc+b(i,j);
 for k=j+1:size(arr,2)
 if(arr(i,k)==0)
 arr(i,k)=inf;
 end
 end
 for k=i+1:size(arr,1)
 if(arr(k,j)==0)
 arr(k,j)=inf;
 end
 end
 end
 end
end
disp('Total Cost');
disp(totalc);
```

CURRENT FOLDE

WORKSPACE

Command Window

```
arr =

     7     5     4     9
     5     6     8    10
     4     2     8     7
     9    11    10     6

cost matrix
     7     5     4     9
     5     6     8    10
     4     2     8     7
     9    11    10     6

after subtracting row minimum and column minimum
     3     1     0     5
     0     1     3     5
     2     0     6     5
     3     5     4     0

count in row is:
     1

y1 is:
     3

count in row is:
     1

count in row is:
     1

count in row is:
     1

count in col is:
     1

y is:
     3

count in col is:
     1

y is:
     3

count in col is:
     1

y is:
     3

count in col is:
     1

y is:
     3

lines is:
     1

count in row is:
     0
```

```
count in row is:
     1

y1 is:
     1

count in row is:
     1


count in row is:
     1

count in col is:
     1

y is:
     1

count in col is:
     1

y is:
     1

count in col is:
     0

y is:
     1

count in col is:
     1

y is:
     1

lines is:
     2

count in row is:
     0

count in row is:
     0


count in row is:
     1

y1 is:
     2

count in row is:
     1

count in col is:
     0

y is:
     2

count in col is:
     1
```

```
y is:
     2

count in col is:
     0

y is:
     2


count in col is:
     1

y is:
     2

lines is:
     3

count in row is:
     0

count in row is:
     0

count in row is:
     0

count in row is:
     1

y1 is:
     4

count in col is:
     0


y is:
     4

count in col is:
     0

y is:
     4

count in col is:
     0

y is:
     4

count in col is:
     1

y is:
     4

lines is:
     4

count in row is:
     0
```

```
count in row is:
     0

count in row is:
     0

count in row is:
     0
count in col is:
     0

y is:
     4

count in col is:
     0


y is:
     4

count in col is:
     0

y is:
     4

count in col is:
     0

y is:
     4
Modified cost matrix
     3     1     0     5
     0     1     3     5
     2     0     6     5
     3     5     4     0

Total Cost
    17

>>
```