NAME: Shreeya Chatterji
CLASS: 3CO16
ROLL NO: 102103447

OT LAB ASSIGNMENT 5

Question 1

$$\text{Maximize} \quad Z = 2x_1 + 3x_2 + 4x_3 + 7x_4$$

$$\begin{aligned} \text{s.t.} \quad 2x_1 + 3x_2 - x_3 + 4x_4 &= 8, \\ x_1 - 2x_2 + 6x_3 - 7x_4 &= -3, \\ x_i \geq 0 \ \forall \ i = 1 - 4. \end{aligned}$$

Code:

```
%% QUESTION 1
% TO OBTAIN BFS USING ALGEBRAIC METHOD
% Question
% Max Z= 2x1+3x2+4x3+7x4
% st: 2x1+3x2-x3+4x4=8
% x1-2x2+6x3-7x4=-3
% %xi>=0; i=1,2,3,4
clc
clear all
format short
% PHASE-1: Input the parameter
c=[2,3,4,7]; %Objective function
A=[2 3 -1 4; -1 2 -6 7]; %Coefficient Matrix
B=[8;3];%RHS of const
objective=1; %1 for max and -1 for minimization problem
%Number of possible solutions: nCm:nchoosek

% PHASE-2: Number of constraint and variable
m=size(A,1); %number of constraints
n=size(A,2); % number of variables

% PHASE-3: Compute the ncm Basic Solutions: The max number of basic
% solutions will always be nCm
nab=nchoosek(n,m); %total number of atmost basic solution
t=nchoosek(1:n,m); %from this we can extract our set of variables that we need to
equate to zero

% PHASE-4:Construct the basic solution
% for this n>m must be satisfied
sol=[]; %default solution is zero (Empty Matrix)
if n>=m %if this is not statisfied then we can not have solutions
    for i=1:nab
    y=zeros(n,1);
    %selecting all rows for a specific column where for t we are taking all
columns for a
    % specific row (which is basically the variables that are equated to zero)
    X=(A(:,t(i,:)))\B;
    %fetching values from A matrix for the rows correspond
    %checking feasibility condition
    if all(X>=0 & X~=inf & X~=-inf)
        y(t(i,:))=X;
        sol=[sol y];
    end
```

NAME: Shreeya Chatterji
CLASS: 3CO16
ROLL NO: 102103447

```
    end
    disp("Solution: ");
    disp(sol);
else
    error('No. of variables is less than number of constraints')
end
if any(X == 0)
        fprintf("DEGENERATE SOLUTION");
else
    fprintf('NON-DEGENERATE SOLUTION\n');
end
%PHASE 5: To find optimal solution
Z=c*sol; %finding the values corresponding to each point
if(objective==1)
    [Zmax,Zindex]=max(Z);%storing the max value of Z and the col in which this max
value resides
else
    [Zmax,Zindex]=min(Z);%storing the min value of Z and the col in which this min
value resides
end
BFS=sol(:,Zindex);%basic feasible solution
[Optimal_Value]=[BFS' Zmax];
Optimal_bfs=array2table(Optimal_Value);
Optimal_bfs.Properties.VariableNames(1:size(Optimal_bfs,2))={'x1','x2','x3','x4','
Optimal Value of Z'};
disp(Optimal_bfs);
```

Output:

```
Solution:
    1.0000    2.4444         0         0
    2.0000         0    2.8125         0
         0         0    0.4375    2.5882
         0    0.7778         0    2.6471

NON-DEGENERATE SOLUTION
    x1    x2      x3        x4      Optimal Value of Z
    __    __    ______    ______    __________________

    0     0     2.5882    2.6471          28.882
```

Question 2

$$\text{Maximize} \quad Z = -x_1 + 2x_2 - x_3$$

$$\begin{aligned} \text{s.t.} \quad x_1 &\leq 4, \\ x_2 &\leq 4, \\ -x_1 + x_2 &\leq 6, \\ -x_1 + 2x_3 &\leq 4, \\ x_1, x_2, x_3 &\geq 0 \end{aligned}$$

NAME: Shreeya Chatterji
CLASS: 3CO16
ROLL NO: 102103447

Code:

```
%% QUESTION 2
% TO OBTAIN BFS USING ALGEBRAIC METHOD
% Question 2
% Max Z= -x1+2x2-x3
% st: x1+s1=4
% x2+s2=4
% -x1+x2+s3=6
% -x1+2x3+s4=4
% x1,x2,x3>=0
clc
clear all
format short
% PHASE-1: Input the parameter
c=[-1,2,-1,0,0,0,0]; %Objective function
A=[1,0,0,1,0,0,0;0,1,0,0,1,0,0;-1,1,0,0,0,1,0;-1,0,2,0,0,0,1]; %Coefficient Matrix
B=[4;4;6;4];%RHS of const
objective=1; %1 for max and -1 for minimization problem
%Number of possible solutions: nCm:nchoosek

% PHASE-2: Number of constraint and variable
m=size(A,1); %number of constraints
n=size(A,2); % number of variables

% PHASE-3: Compute the ncm Basic Solutions: The max number of basic
% solutions will always be nCm
nab=nchoosek(n,m); %total number of atmost basic solution
t=nchoosek(1:n,m); %from this we can extract our set of variables that we need to
equate to zero

% PHASE-4:Construct the basic solution
% for this n>m must be satisfied
sol=[]; %default solution is zero (Empty Matrix)
if n>=m %if this is not statisfied then we can not have solutions
    for i = 1:nab
        y = zeros(n, 1);
        % Check if the selected variables form a singular matrix
        if rank(A(:, t(i, :))) == m
            X = A(:, t(i, :)) \ B; % Solve for basic variables
            if all(X >= 0)
                y(t(i, :)) = X;
                sol = [sol y];
            end
        end
    end
    disp("Solution: ");
    disp(sol);
else
    error('No. of variables is less than number of constraints')
end
if any(X == 0)
        fprintf("DEGENERATE SOLUTION");
else
    fprintf('NON-DEGENERATE SOLUTION\n');
end
%PHASE 5: To find optimal solution
Z=c*sol; %finding the values corresponding to each point
if(objective==1)
```

```
    [Zmax,Zindex]=max(Z);%storing the max value of Z and the col in which this max
value resides
else
    [Zmax,Zindex]=min(Z);%storing the min value of Z and the col in which this min
value resides
end
BFS=sol(:,Zindex);%basic feasible solution
[Optimal_Value]=[BFS' Zmax];
Optimal_bfs=array2table(Optimal_Value);
Optimal_bfs.Properties.VariableNames(1:size(Optimal_bfs,2))={'x1','x2','x3','s1','
s2','s3','s4','Optimal Value of Z'};
disp(Optimal_bfs);
```

Output:

```
Solution:
     4     4     4     4     0     0     0     0
     4     4     0     0     4     4     0     0
     4     0     4     0     2     0     2     0
     0     0     0     0     4     4     4     4
     0     0     4     4     0     0     4     4
     6     6    10    10     2     2     6     6
     0     8     0     8     0     4     0     4

NON-DEGENERATE SOLUTION
    x1    x2    x3    s1    s2    s3    s4    Optimal Value of Z
    __    __    __    __    __    __    __    __________________

    0     4     0     4     0     2     4             8
```

Question 3:

$$\text{Minimize} \quad Z = 5x_2 - 2x_1$$

$$\begin{aligned} \text{s.t.} \quad 2x_1 + 5x_2 &\leq 8, \\ x_1 + x_2 &\leq 2, \\ x_1, x_2 &\geq 0 \end{aligned}$$

Code:

```
%% QUESTION 3
% TO OBTAIN BFS USING ALGEBRAIC METHOD
% Question
% Min Z= 5x2-2x1
% st: 2x1+5x2+s1=8
% x1+x2+s2=2
% %xi>=0
clc
clear all
format short
% PHASE-1: Input the parameter
c=[-2,5,0,0]; %Objective function
A=[2,5,1,0;1,1,0,1]; %Coefficient Matrix
B=[8;2];%RHS of const
objective=-1; %1 for max and -1 for minimization problem
```

```
%Number of possible solutions: nCm:nchoosek

% PHASE-2: Number of constraint and variable
m=size(A,1); %number of constraints
n=size(A,2); % number of variables

% PHASE-3: Compute the ncm Basic Solutions: The max number of basic
% solutions will always be nCm
nab=nchoosek(n,m); %total number of atmost basic solution
t=nchoosek(1:n,m); %from this we can extract our set of variables that we need to
equate to zero

% PHASE-4:Construct the basic solution
% for this n>m must be satisfied
sol=[]; %default solution is zero (Empty Matrix)
if n>=m %if this is not statisfied then we can not have solutions
    for i=1:nab
    y=zeros(n,1);
    %selecting all rows for a specific column where for t we are taking all
columns for a
    % specific row (which is basically the variables that are equated to zero)
    X=(A(:,t(i,:)))\B;
    %fetching values from A matrix for the rows correspond
    %checking feasibility condition
    if all(X>=0 & X~=inf & X~=-inf)
        y(t(i,:))=X;
        sol=[sol y];
    end
    end
    disp("Solution: ");
    disp(sol);
else
    error('No. of variables is less than number of constraints')
end
if any(X == 0)
        fprintf("DEGENERATE SOLUTION\n");
else
    fprintf('NON-DEGENERATE SOLUTION\n');
end

%PHASE 5: To find optimal solution
Z=c*sol; %finding the values corresponding to each point
if(objective==1)
    [Zmax,Zindex]=max(Z);%storing the max value of Z and the col in which this max
value resides
else
    [Zmax,Zindex]=min(Z);%storing the min value of Z and the col in which this min
value resides
end
%Optimal BFS
BFS=sol(:,Zindex);%basic feasible solution
[Optimal_Value]=[BFS' Zmax];
Optimal_bfs=array2table(Optimal_Value);
Optimal_bfs.Properties.VariableNames(1:size(Optimal_bfs,2))={'x1','x2','s1','s2','
Optimal Value of Z'};
disp(Optimal_bfs);
```

Output:

NAME: Shreeya Chatterji
CLASS: 3CO16
ROLL NO: 102103447

```
Solution:
    0.6667    2.0000         0         0
    1.3333         0    1.6000         0
         0    4.0000         0    8.0000
         0         0    0.4000    2.0000

NON-DEGENERATE SOLUTION
    x1     x2     s1     s2     Optimal Value of Z
    __     __     __     __     __________________

    2      0      4      0             -4
```

Question 4:

$$\text{Maximize} \quad Z = x_1 + x_2 + x_3$$

$$\begin{aligned} \text{s.t.} \quad x_1 + x_2 &\leq 1, \\ -x_2 + x_3 &\leq 0, \\ x_1, x_2, x_3 &\geq 0 \end{aligned}$$

Code:

```
%% QUESTION 4
% TO OBTAIN BFS USING ALGEBRAIC METHOD
% Question
% Max Z= x1+x2+x3+0s1+0s2
% st: x1+x2+s1=1
% -x2+x3+s2=0
% %xi>=0
clc
clear all
format short

% PHASE-1: Input the parameter
c=[1,1,1,0,0]; %Objective function
A=[1,1,0,1,0;0,-1,1,0,1]; %Coefficient Matrix
B=[1;0];%RHS of const
objective=1; %1 for max and -1 for minimization problem
%Number of possible solutions: nCm:nchoosek

% PHASE-2: Number of constraint and variable
m=size(A,1); %number of constraints
n=size(A,2); % number of variables

% PHASE-3: Compute the ncm Basic Solutions: The max number of basic
% solutions will always be nCm
nab=nchoosek(n,m); %total number of atmost basic solution
t=nchoosek(1:n,m); %from this we can extract our set of variables that we need to
equate to zero

% PHASE-4:Construct the basic solution
% for this n>m must be satisfied
```

```
sol=[]; %default solution is zero (Empty Matrix)
if n>=m %if this is not statisfied then we can not have solutions
    for i = 1:nab
        y = zeros(n, 1);
        % Check if the selected variables form a singular matrix
        if rank(A(:, t(i, :))) == m
            X = A(:, t(i, :)) \ B; % Solve for basic variables
            if all(X >= 0)
                y(t(i, :)) = X;
                sol = [sol y];
            end
        end
    end
    disp("Solution: ");
    disp(sol);
else
    error('No. of variables is less than number of constraints')
end
if any(X == 0)
        fprintf("DEGENERATE SOLUTION\n");
else
    fprintf('NON-DEGENERATE SOLUTION\n');
end
%PHASE 5: To find optimal solution
Z=c*sol; %finding the values corresponding to each point
if(objective==1)
    [Zmax,Zindex]=max(Z);%storing the max value of Z and the col in which this max
value resides
else
    [Zmax,Zindex]=min(Z);%storing the min value of Z and the col in which this min
value resides
end
BFS=sol(:,Zindex);%basic feasible solution
[Optimal_Value]=[BFS' Zmax];
Optimal_bfs=array2table(Optimal_Value);
Optimal_bfs.Properties.VariableNames(1:size(Optimal_bfs,2))={'x1','x2','x3','s1','
s2','Optimal Value of Z'};
disp(Optimal_bfs);
```

Output:

```
Solution:
     1     1     1     0     0     0     0     0
     0     0     0     1     0     1     0     0
     0     0     0     1     0     0     0     0
     0     0     0     0     1     0     1     1
     0     0     0     0     0     1     0     0

DEGENERATE SOLUTION
    x1    x2    x3    s1    s2    Optimal Value of Z
    __    __    __    __    __    __________________

    0     1     1     0     0             2
```