MAT2003 (Optimization Techniques) Lab-2 01/09/22

KHAN MOHD. OWAIS RAZA (20BCD7138)

Code with errors :-

```
%%  KHAN MOHD OWAIS RAZA (20BCD7138)
%%  MAT2003 (Optimization Techniques) Lab
% Code with errors
Clc
Clear
format short
%%%%%% Stage 1: %%%%%%%%
C= [3 5];
A= [ 1 2 ; 1 1 ; 0 1 ];
b= [2000; 1500; 600];
%%%%%%%%% Stage 2: ploting the constraints in 2d graph%%%%%%%%%%
y1= 0:1: max(b);
x21= (b(1) - A(1,1) .*y1)./A(1,2);
X22= (b(2) - A(2,1) .*y1)./A(2,2);
X23= (b(3) - A(3,1) .*y1)./A(3,2);
X21= max(0,X21);
X22= max(0,X22);
X23= max(0,X23);
plot(y1,X21, 'r', y1,X22,'k', y1, X23, 'b');
xlabel( 'value of x1');
ylabel( 'value of x2');
title('x1 vs x2');
legend('x1+2x2=2000', ' x1+x2=1500', 'x2=600')
%%%%%%%%%%%%Phase 3 Find the corner point i.e., pt of intersections
Cx1=find(y1==0);
C1 = find(X21==0);
Line1= [y1(:, [C1 Cx1]) ; X21(:, [C1 Cx1])]';
C2 = find(X22==0);
Line2= [y1(:, [C2 Cx1]) ; X22(:, [C2 Cx1])]';
C3 = find(X23==0);
Line3= [y1(:, [C3 Cx1]) ; X23(:, [C3 Cx1])]';
Corpt= unique([Line1;Line2;Line3],'row');
%%%%%%%%Stage 4 Find the intersection points%%%%%%%%
HG=[0;0];
for i=1:size(A,1)
Hg1=A(i,: );
B1=b(i,: ) ;
for j=i+1: size(A,1)
Hg2= A(j,: );
B2= b (j, : );
Aa= [Hg1; Hg2];
Bb= [B1;B2];
Xx= Aa\Bb;
HG=[ HG Xx];
```

```
end
end
Pt = HG';
%%%%%%%%%%%%%%%% Stage 5 write all points, i.e., corner +
intersection points ********
Allpt = [Pt; Corpt];
%%%%%%%%%%%%%%%% stage 6: find the feasible region %%%%%%%%%
PT= constraint(Allpt);
PT= unique(PT,'row');
%%%%%%% stage 7*************
for i=1: size(PT,1)
FX(i, : ) = sum (PT (i,: ).*C);
End
%%%%%%%% final Stage optimal Solution %%%%%%%%%%
vert_fns = [PT FX];
[fxval, indfx]= max(FX);
optval = vert_fns(indfx, :);
optimal_bfs= array2table( Optval);
disp(" x value is")
disp(optimal_bfs(1,[1]))
disp("y value is")
disp(optimal_bfs(1,[1]))
disp("max value is")
disp(optimal_bfs(1,[1]))
function X = constraint(X)
%%%%%%% write the first constraint here%%%%%%%
X1= X(: , 1);
X2= X(: , 2);
Cons1 = X1+2.*X2-2000;
H1=find(Cons1>0);
X(H1,: )=[];
%%%%%%% write the Second constraint here%%%%%%%
X1= X(: , 1);
X2= X(: , 2);
Cons2 = X1+X2-1500;
H2=find(Cons2>0);
X(H2,: )=[];
%%%%%%% write the Third constraint here%%%%%%%
X1= X(: , 1);
X2= X(: , 2);
Cons3 = X2-1500;
H3=find(Cons3>0);
X(H3,: )=[];
end
```

Error:-

```
Function definition are not supported in this context. Functions can only be
created as local or nested functions in code files.
```

```
Command Window
>> MAT2003_Lab2
Error: File: MAT2003_Lab2.m Line: 68 Column: 1
Function definition are not supported in this context. Functions can only be created as local or nested functions in code files.
```

Corrected code and output :-

```
%% KHAN MOHD OWAIS RAZA (20BCD7138)
%% Optimization Techniques (MAT2003)
% Corrected code
clc
clear
format short
%%%%%% Stage 1: %%%%%%%%
C= [3 5];
A= [ 1 2 ; 1 1 ; 0 1 ];
b= [2000; 1500; 600];
%%%%%%%%% Stage 2: ploting the constraints in 2d graph%%%%%%%%%%
y1= 0:1: max(b);
X21= (b(1) - A(1,1) .*y1)./A(1,2);
X22= (b(2) - A(2,1) .*y1)./A(2,2);
X23= (b(3) - A(3,1) .*y1)./A(3,2);
X21= max(0,X21);
X22= max(0,X22);
X23= max(0,X23);
plot(y1,X21, 'r', y1,X22,'k', y1, X23, 'b');
xlabel( 'value of x1');
ylabel( 'value of x2');
title('x1 vs x2');
legend('x1+2x2=2000', ' x1+x2=1500', 'x2=600')
%%%%%%%%%%%%Phase 3 Find the corner point i.e., pt of intersections
Cx1=find(y1==0);
C1 = find(X21==0);
Line1= [y1(:, [C1 Cx1]) ; X21(:, [C1 Cx1])]';
C2 = find(X22==0);
Line2= [y1(:, [C2 Cx1]) ; X22(:, [C2 Cx1])]';
C3 = find(X23==0);
Line3= [y1(:, [C3 Cx1]) ; X23(:, [C3 Cx1])]';
Corpt= unique([Line1;Line2;Line3],'row');
%%%%%%%%Stage 4 Find the intersection points%%%%%%%%
HG=[0;0];
for i=1:size(A,1)
Hg1=A(i,: );
B1=b(i,: );
for j=i+1: size(A,1)
Hg2= A(j,: );
B2= b (j,: );
Aa= [Hg1; Hg2];
Bb= [B1;B2];
Xx= Aa\Bb;
HG=[ HG Xx];
end
end
```

```
Pt = HG';
%%%%%%%%%%%%%%%% Stage 5 write all points, i.e., corner +
intersection points ********
Allpt = [Pt; Corpt];
%%%%%%%%%%%%%%%% stage 6: find the feasible region %%%%%%%%%
PT= constraint(Allpt);
PT= unique(PT,'row');
%%%%%%% stage 7*************
for i=1: size(PT,1)
FX(i, : ) = sum (PT (i,: ).*C);
end
%%%%%%%% Final Stage optimal Solution %%%%%%%%%%
vert_fns = [PT FX];
[fxval, indfx]= max(FX);
optval = vert_fns(indfx, :);
optimal_bfs= array2table( optval);
disp(" x value is")
disp(optimal_bfs(1,1))
disp("y value is")
disp(optimal_bfs(1,1))
disp("max value is")
disp(optimal_bfs(1,1))
function X = constraint(X)
%%%%%%% First constraint %%%%%%%
X1= X(: , 1);
X2= X(: , 2);
Cons1 = X1+2.*X2-2000;
H1=find(Cons1>0);
X(H1,: )=[];
%%%%%%% Second constraint %%%%%%%
X1= X(: , 1);
X2= X(: , 2);
Cons2 = X1+X2-1500;
H2=Cons2>0;
X(H2,: )=[];
%%%%%%% Third constraint %%%%%%%
X1= X(: , 1);
X2= X(: , 2);
Cons3 = X2-1500;
H3=Cons3>0;
X(H3,: )=[];
end
```

Output :-

Command Window

```
 x value is
    optval1
    _______

     1000

y value is
    optval1
    _______

     1000

max value is
    optval1
    _______

     1000
```