```
%%%%%%%%%%%%%%%%%%%%  Problem 03  %%%%%%%%%%%%%%%%%%%%

clc
clear
a=[2 4 4 1; 10 3 7 7; 6 7 20 5];d=[50 100 150
 200];s=[150;200;150];m=3;n=4;
Given_Problem=[a s;d sum(d)]
fprintf("The initial Matrix,")  %%%%%%   Initial function created  %%%
[b,Initial_LCM_Cost]=LCM(a,d,s,m,n)%Initial_Solution=[b s;d sum(d)]

fprintf("\n\n\nRoundwise Cij-Ui-Vj Matrix,")
c=optCheck(a,b,m,n) %%%%%%%%%% Optimality   %%%%%%%%%%
while(sum(sum(c<0))~=0)%%%%%%%%%%%%%%   Subtraction Logic without DFS
 %%%%%%%%%%
[p,q]=find(min(min(c))==c,1);
T=0;
for i=1:m
    for j=1:n
        if(b(i,j)~=0 && (i==p||j==q))
            if(i==p)
                for k=1:m
                    if(b(k,q)~=0 && b(k,j)~=0)
                          T=-5;
                          dsub=min(b(k,q),b(i,j));
                          b(p,q)=b(p,q)+dsub;
                          b(k,q)=b(k,q)-dsub;
                          b(k,j)=b(k,j)+dsub;
                          b(i,j)=b(i,j)-dsub;
                          break
                    end
                end
            else
                for k=1:n
                    if(b(k,q)~=0 && b(k,j)~=0)
                          T=-5;
                          dsub=min(b(i,j),b(p,k));
                          b(p,q)=b(p,q)-dsub;
                          b(i,j)=b(i,j)-dsub;
                          b(i,k)=b(i,k)-dsub;
                          b(p,k)=b(p,k)-dsub;
                          break
                    end
                end

            end
            break
        end
    end
    if(T==-5)
        break
    end
end
```

```
fprintf("\n\n\nROundwise Cij-Ui-Vj Matrix,")
c=optCheck(a,b,m,n) %%%%%%%%%% Optimality   %%%%%%%%%%
fprintf("Solution Matrix,")
b
end
fprintf("The optimality condition satisfied and hence the cost is: %d
\n\n",sum(sum(b.*a)))



function[c]=optCheck(a,b,m,n)
c=zeros(m,n);
[p q]=find(0~=b);
u(1:m)=intmax;
v(1:n)=intmax;
u(p(1))=0;%%% Initialisationfor u v
while(sum(intmax==u)+sum(intmax==v)~=0)
    [u v]=solv(u,v,p,q,a); %Solve with a simple function
end
for i=1:m %%%%% Optimality Checking at empty cell
    for j=1:n
        if(b(i,j)==0)
            c(i,j)=a(i,j)-u(i)-v(j);
        end
    end
end
end

function[u v]=solv(u,v,p,q,a)
k=length(p);
for i=1:k
    if(u(p(i))~=intmax)
        v(q(i))=a(p(i),q(i))-u(p(i));
    end
% end
% for i=1:k
    if(v(q(i))~=intmax)
        u(p(i))=a(p(i),q(i))-v(q(i));
    end
end
end

function [outputArg2,outputArg1] = LCM(Cost,d,s,m,n)
a=Cost;
b=zeros(m,n);
while((sum(d)+sum(s)>0))
    mm=min(min(a));
    [p,q] = find(a==mm,1);
    b(p,q)=min(s(p),d(q));
    s(p)=s(p)-b(p,q);d(q)=d(q)-b(p,q);
    if(s(p)==0)
        a(p,:)=intmax;
    else
        a(:,q)=intmax;
```

```
    end
end
sm=sum(sum(Cost.*b));
outputArg1 = sm;
outputArg2 = b;
end
```

```
Given_Problem =

     2     4     4     1   150
    10     3     7     7   200
     6     7    20     5   150
    50   100   150   200   500

The initial Matrix,
b =

     0     0     0   150
     0   100   100     0
    50     0    50    50


Initial_LCM_Cost =

        2700



Roundwise Cij-Ui-Vj Matrix,
c =

     0    -8   -12     0
    17     0     0    15
     0    -9     0     0



ROundwise Cij-Ui-Vj Matrix,
c =

     0     4     0     0
     5     0     0     3
     0     3    12     0

Solution Matrix,
b =

     0     0    50   100
     0   100   100     0
    50     0     0   100
```

```
The optimality condition satisfied and hence the cost is: 2100
```

Published with MATLAB® R2018b