**Exp 7**

**Code:**

//MAIN PROGRAM

#include<stdio.h>

int main()

{

int allocation[10][5],max[10][5],need[10][5],available[3],flag[10],sq[10];

int n,r,i,j,k,count,count1=0;

printf("\n Input the number of processes running ( <10 )..");

scanf("%d",&n);

for(i=0;i<10;i++)

flag[i]=0;

printf("\n Input the number of resources ( <5 )..");

scanf("%d",&r);

printf("\n Input the allocation matrix for the processes in row major order..\n");

for(i=0;i<n;i++)

{

printf("\n Process %d\n",i);

for(j=0;j<r;j++)

{

printf("\n Resource %d\n",j);

scanf("%d",&allocation[i][j]);

}

}

printf("\n Input the no. of resources that a process can maximum have..\n");

for(i=0;i<n;i++)

{

printf("\n Process %d\n",i);

for(j=0;j<r;j++)

{

printf("\n Resource %d\n",j);

scanf("%d",&max[i][j]);

}

}

printf("\n Input the no. of available instances of each resource..\n");

for(i=0;i<r;i++)

{

printf("\n Resource %d : ",i);

scanf("%d",&available[i]);

}

printf("\n The need matrix is as follows : \n");

for(i=0;i<n;i++)

{

for(j=0;j<r;j++)

{

need[i][j]= max[i][j]-allocation[i][j];

printf("\t %d",need[i][j]);

}

printf("\n");

}

do{

for(k=0;k<n;k++)

{

for(i=0;i<n;i++)

{

if(flag[i]==0)

{

count=0;

for(j=0;j<r;j++)

{

if(available[j]>=need[i][j])

count++;

}

if(count==r)

{

count1++;

flag[i]=1;

sq[count1-1]=i;

for(j=0;j<r;j++)

{

available[j]=available[j]+allocation[i][j];

}

break;

}

}

}

}

if(count1!=n)

{

printf("\n---------------IT'S AN UNSAFE STATE---------------");

break;

}

}while(count1!=n);

if(count1==n)

{

printf("\n \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*IT'S A SAFE STATE\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n The safe sequence is....\n");

for(i=0;i<n;i++)

printf("\t P%d",sq[i]);

printf("\n");

printf("\n The available matrix is now : ");

for(i=0;i<r;i++)

printf("\t %d",available[i]);

}

return(0);

}

**Output**:

etc@CETCL04-13:~$ cd Desktop

etc@CETCL04-13:~/Desktop$ gcc bankers.c

etc@CETCL04-13:~/Desktop$ ./a.out

Input the number of processes running ( <10 )..5

Input the number of resources ( <5 )..3

Input the allocation matrix for the processes in row major order..

Process 0

Resource 0

0

Resource 1

1

Resource 2

0

Process 1

Resource 0

2

Resource 1

0

Resource 2

0

Process 2

Resource 0

3

Resource 1

0

Resource 2

2

Process 3

Resource 0

2

Resource 1

1

Resource 2

1

Process 4

Resource 0

0

Resource 1

0

Resource 2

2

Input the no. of resources that a process can maximum have..

Process 0

Resource 0

7

Resource 1

5

Resource 2

3

Process 1

Resource 0

3

Resource 1

2

Resource 2

2

Process 2

Resource 0

9

Resource 1

0

Resource 2

2

Process 3

Resource 0

2

Resource 1

2

Resource 2

2

Process 4

Resource 0

4

Resource 1

3

Resource 2

3

Input the no. of available instances of each resource..

Resource 0 : 3

Resource 1 : 3

Resource 2 : 2

The need matrix is as follows :

7 4 3

1 2 2

6 0 0

0 1 1

4 3 1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*IT'S A SAFE STATE\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

The safe sequence is....

P1 P3 P0 P2 P4

The available matrix is now : 10 5 7