DEADLOCK DETECTION

PROGRAM:

#include<stdio.h>

static int mark[20];

int i, j, np, nr;

int main()

{

int alloc[10][10],request[10][10],avail[10],r[10],w[10];

printf ("\nEnter the no of the process: ");

scanf("%d",&np);

printf ("\nEnter the no of resources: ");

scanf("%d",&nr);

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

{

printf("\nTotal Amount of the Resource R % d: ",i+1);

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

}

printf("\nEnter the request matrix:");

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

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

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

printf("\nEnter the allocation matrix:");

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

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

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

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

{

avail[j]=r[j];

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

{

avail[j]-=alloc[i][j];

}

}

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

{

int count=0;

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

{

if(alloc[i][j]==0)

count++;

else

break;

}

if(count==nr)

mark[i]=1;

}

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

w[j]=avail[j];

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

{

int canbeprocessed= 0;

if(mark[i]!=1)

{

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

{

if(request[i][j]<=w[j])

canbeprocessed=1;

else

{

canbeprocessed=0;

break;

}

}

if(canbeprocessed)

{

mark[i]=1;

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

w[j]+=alloc[i][j];

}

}

}

int deadlock=0;

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

if(mark[i]!=1)

deadlock=1;if

(deadlock)

printf("\n Deadlock detected");

else

printf("\n No Deadlock possible");

}

OUTPUT:

