**#include<stdio.h>**

**#include<stdlib.h>**

**int avail[10],maximum[20][20],allocated[20][20],m,n,need[20][20];**

**void input()**

**{**

**int i,j,val;**

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

**{**

**for(j=0;j<m;j++)**

**{**

**printf("\nENTER THE MAX NUMBER OF INSTANCES for P%d,R%d: ",i,j);**

**loop: scanf("%d",&val);**

**if(val>avail[j])**

**{**

**printf("ERROR!!THE ENTERED VALUE IS GREATER THAN THE**

**AVAILABLE VALUE i.e %d .PLZ ENTER THE VALUE AGAIN: ",avail[j]);**

**goto loop;**

**}**

**Else**

**maximum[i][j]=val;**

**}**

**}**

**}**

**void rollback(int roll[],int process)**

**{**

**int k; for(k=0;k<m;k++)**

**{**

**avail[k]=avail[k]+roll[k];**

**allocated[process][k]=allocated[process][k]-roll[k];**

**need[process][k]=allocated[process][k]+roll[k];**

**printf("\nWE ARE ROLLING BACK");**

}

}

int safety()

{

**int i,j,work[10],finish[10],flag=0,k,cnt=0,cn,flag1=0,req;**

**//char req;**

**printf("\nTHE ALTERED allocatedCATION TABLE IS:\n");**

**printf("processCESS");**

**for(i=0;i<m;i++)**

**{**

**printf("\tR%d\t",i);**

**}**

**printf("\n");**

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

**{**

**printf("\nP%d\t",i);**

**for(j=0;j<m;j++)**

**{**

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

**}**

**}**

**j=0;**

**for(i=0;i<m;i++)**

**{ work[j]=avail[i];**

**j++;**

**}**

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

**{**

**finish[i]=0;**

**}**

**while(cnt<=10)**

**{**

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

**{**

**if(finish[i]==0)**

**{**

**for(j=0;j<m;j++)**

**{**

**if(need[i][j]<=work[j])**

**{**

**flag=1;**

**}**

**else { flag=0; break;**

**}**

**} if(flag==1)**

**{**

**printf("\nTHE processCESS P%d RAN SUCCESSFULLY",i);**

**finish[i]=1;**

**for(k=0;k<m;k++)**

**{ work[k]=work[k]+allocated**[i][k];

**}**

**}**

**}**

**}**

**for(cn=0;cn<n;cn++)**

**{**

**if(finish[cn]==1)**

**{**

**flag1=1;**

**}**

**Else**

**{**

**flag1=0;**

**break;**

**}**

**}**

**cnt++;**

**if(flag1==1)**

**printf("\nTHE SYSTEM IS IN SAFE STATE\n");**

**req=0; break;**

**}**

**}//end of while**

**if(flag1==0)**

**{**

**printf("\nTHE SYSTEM IS NOT IN SAFE STATE\n");**

**req=1;**

**}**

**return req;**

**}**

**void check()**

**{**

**int process,val,i,req1,k,roll[10];**

**char req;**

**while(1)**

**{**

**printf("\nDO YOU HAVE A check?(y/n)");**

**scanf("%c",&req);**

**scanf("%c",&req);**

**k=0;**

**if(req=='y'||req=='Y')**

**{**

**printf("ENTER THE processCESS checkING(0,1,2,3..): ");**

**loop0:**

**scanf("%d",&process);**

**if(process<=m)**

**{**

**for(i=0;i<m;i++)**

**{**

**printf("ENTER THE REQUIRED VALUE OF RESOURCE R%d: ",i);**

**loop1: scanf("%d",&val);**

**if(val>maximum[process][i])**

**{**

**printf("ERROR!!THE ENTERED VALUE IS GREATER THAN THE MAX VALUE REQD. i.e %d .PLZ ENTER THE VALUE AGAIN: ",maximum[process][i]);**

**goto loop1;**

**}**

**Else**

**{**

**allocated[process][i]=allocated[process][i]+val;**

**avail[i]=avail[i]-val;**

**need[process][i]=need[process][i]-val;**

**roll[k]=val;**

**k++;**

**}**

**}**

**}**

**Else**

**{**

**printf("THE processCESS NUMBER ENTERED IS INVALID !! RE ENTER THE VALUE BY OBSERVING THE TABLE");**

**goto loop0;**

**}**

**req1=safety();**

**if(req1==1)**

**{**

**rollback(roll,process);**

**}**

**}**

**else**

**{**

**exit(0);**

**}**

**}**

**}**

**int main()**

**{**

**int i,val,j;**

**printf("\nWELCOME TO USER Ms.REENA \n");**

**printf("\nENTER THE NUMBER OF RESOURCES: ");**

**scanf("%d",&m); printf("\nENTER THE NUMBER OF processCESS(s): ");**

**scanf("%d",&n); for(i=0;i<m;i++)**

**{**

**printf("\nENTER THE MAX AVAILABLE VALUE FOR RESOURCE R%d: ",i);**

**scanf("%d",&val);**

**avail[i]=val;**

**}**

**input();**

**printf("\nTHE TABLE ENTERED FOR MAX NEEDED IS:\n");**

**printf("processCESS");**

**for(i=0;i<m;i++)**

**{**

**printf("\tR%d\t",i);**

**}**

**printf("\n");**

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

**{**

**printf("\nP%d\t",i);**

**for(j=0;j<m;j++)**

**{**

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

**}**

**}**

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

**{**

**for(j=0;j<m;j++)**

**{**

**need[i][j]=maximum[i][j]-allocated[i][j];**

**}**

**}**

**check();**

**return 0;**

**}**

**Output:** WELCOME TO USER Ms.REENA

ENTER THE NUMBER OF RESOURCES: 2

ENTER THE NUMBER OF processCESS(s): 2

ENTER THE MAX AVAILABLE VALUE FOR RESOURCE R0: 3

ENTER THE MAX AVAILABLE VALUE FOR RESOURCE R1: 4

ENTER THE MAX NUMBER OF INSTANCES for P0,R0: 5

ERROR!!THE ENTERED VALUE IS GREATER THAN THE AVAILABLE VALUE i.e 3

.PLZ ENTER THE VALUE AGAIN: 2

ENTER THE MAX NUMBER OF INSTANCES for P0,R1: 3

ENTER THE MAX NUMBER OF INSTANCES for P1,R0: 2

ENTER THE MAX NUMBER OF INSTANCES for P1,R1: 4

THE TABLE ENTERED FOR MAX NEEDED IS:

processCESS R0 R1

P0 2 3 P1 2 4 DO YOU HAVE A check?(y/n)y

ENTER THE processCESS checkING(0,1,2,3..): 0

ENTER THE REQUIRED VALUE OF RESOURCE R0: 1

ENTER THE REQUIRED VALUE OF RESOURCE R1: 2

THE ALTERED allocatedCATION TABLE IS:

processCESS R0 R1

P0 1 2

P1 0 0

THE processCESS P0 RAN SUCCESSFULLY

THE processCESS P1 RAN SUCCESSFULLY

THE SYSTEM IS IN SAFE STATE

DO YOU HAVE A check?(y/n)n

logout

Saving session...

...copying shared history...

...saving history...truncating history files...

...completed.

[Process completed]