WEEK 6 OS LAB

1BM21CS247

q)Use bankers algo given here to check if the following state is safe/unsafe:

source code:

#include <stdio.h>

int main()

{

int n, m, i, j, k;

int alloc[50][50];

int max[50][50];

int avail[50];

printf("Enter the Number of processes:\n");

scanf("%d",&n);

printf("Enter the Number of resources:\n");

scanf("%d",&m);

printf("Enter the allocation matrix:\n");

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

{

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

{

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

}

}

printf("Enter the Max matrix:\n");

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

{

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

{

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

}

}

printf("Enter the available resources:\n");

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

{

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

}

int f[n], ans[n], ind = 0;

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

f[k] = 0;

}

int need[n][m];

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

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

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

}

int y = 0;

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

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

if (f[i] == 0) {

int flag = 0;

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

if (need[i][j] > avail[j]){

flag = 1;

break;

}

}

if (flag == 0) {

ans[ind++] = i;

for (y = 0; y < m; y++)

avail[y] += alloc[i][y];

f[i] = 1;

}

}

}

}

int flag = 1;

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

{

if(f[i]==0)

{

flag=0;

printf("The following system is not safe");

break;

}

}

if(flag==1)

{

printf("Following is the SAFE Sequence\n");

for (i = 0; i < n - 1; i++)

printf(" P%d ->", ans[i]);

printf(" P%d", ans[n-1]);

}

return (0);

}

OUTPUT:-

