BANKERS ALGORITHIM

#include <stdio.h>

int main()

{

int processes = 4;

int resources = 3;

int available[resources] = {9, 3, 6};

int claim[processes][resources] =

{

{3, 2, 2},

{6, 1, 3},

{3, 1, 4},

{4, 2, 2}

};

int allocation[processes][resources] =

{

{1, 0, 0},

{6, 1, 2},

{2, 1, 1},

{0, 0, 2}

};

int work[resources];

int finish[processes];

int i, j, k, safe, count;

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

{

work[i] = available[i];

}

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

{

finish[i] = 0;

}

safe = 0;

count = 0;

while (count < processes)

{

safe = 0;

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

{

if (!finish[i])

{

safe = 1;

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

{

if (claim[i][j] - allocation[i][j] > work[j])

{

safe = 0;

break;

}

}

if (safe)

{

finish[i] = 1;

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

{

work[k] += allocation[i][k];

}

count++;

break;

}

}

}

if (!safe)

{

break;

}

}

if (count == processes) {

printf("System is in a safe state\n");

} else {

printf("System is in an unsafe state\n");

}

return 0;

}

OUTPUT:

