Write a C program to simulate: (Any one)
a) Bankers' algorithm for the purpose of deadlock avoidance.

```
#include <stdio.h>
#include <stdbool.h>

#define MAX 10

int main() {
   int n, m;
   int alloc[MAX][MAX], max[MAX][MAX], avail[MAX];
   int need[MAX][MAX];
   int finish[MAX] = {0}, safeSeq[MAX];

   printf("Enter number of processes: ");
   scanf("%d", &n);
   printf("Enter number of resources: ");
   scanf("%d", &m);

   printf("Enter allocation matrix:\n");
   for (int i = 0; i < n; i++)
      for (int j = 0; j < m; j++)
         scanf("%d", &alloc[i][j]);

   printf("Enter maximum matrix:\n");
   for (int i = 0; i < n; i++)
      for (int j = 0; j < m; j++)
         scanf("%d", &max[i][j]);

   printf("Enter available resources:\n");
   for (int i = 0; i < m; i++)
      scanf("%d", &avail[i]);

   for (int i = 0; i < n; i++)
      for (int j = 0; j < m; j++)
         need[i][j] = max[i][j] - alloc[i][j];
```

```
    int count = 0;
    while (count < n) {
        bool found = false;
        for (int i = 0; i < n; i++) {
            if (!finish[i]) {
                bool canAllocate = true;
                for (int j = 0; j < m; j++) {
                    if (need[i][j] > avail[j]) {
                        canAllocate = false;
                        break;
                    }
                }
                if (canAllocate) {
                    for (int j = 0; j < m; j++)
                        avail[j] += alloc[i][j];
                    safeSeq[count++] = i;
                    finish[i] = 1;
                    found = true;
                }
            }
        }
        if (!found) {
            printf("System is not in a safe state.\n");
            return 0;
        }
    }

    printf("System is in a safe state.\nSafe sequence is: ");
    for (int i = 0; i < n; i++)
        printf("P%d ", safeSeq[i]);
    printf("\n");

    return 0;
}
```

OUTPUT

SAFE STATE:

```
Enter number of processes: 5
Enter number of resources: 3
Enter allocation matrix:
0 1 0
2 0 0
3 0 2
2 1 1
0 0 2
Enter maximum matrix:
7 5 3
3 2 2
9 0 2
2 2 2
4 3 3
Enter available resources:
3 3 2
System is in a safe state.
Safe sequence is: P1 P3 P4 P0 P2
```

NOT SAFE STATE

```
Enter number of processes: 5
Enter number of resources: 3
Enter allocation matrix:
0 1 0
2 0 0
3 0 2
2 1 1
0 0 2
Enter maximum matrix:
7 5 3
3 2 2
9 0 2
2 2 2
4 3 3
Enter available resources:
1 1 0
System is not in a safe state.
```