```
//Banker’s Algorithm

#include <stdio.h>

int main() {
    int n = 5; // Number of processes
    int m = 3; // Number of resources

    // Allocation Matrix
    int alloc[5][3] = {
        {0, 1, 0},
        {2, 0, 0},
        {3, 0, 2},
        {2, 1, 1},
        {0, 0, 2}
    };

    // Maximum Matrix
    int max[5][3] = {
        {7, 5, 3},
        {3, 2, 2},
        {9, 0, 2},
        {2, 2, 2},
        {4, 3, 3}
    };

    // Available Resources
    int avail[3] = {3, 3, 2};

    int ans[5];
    int f[5] = {0};

    for (int k = 0; k < n; k++) {
        for (int i = 0; i < n; i++) {
            if (f[i] == 0) {
                int flag = 0;
                for (int j = 0; j < m; j++) {
                    if (max[i][j] - alloc[i][j] > avail[j]) {
                        flag = 1;
                        break;
                    }
                }
                if (flag == 0) {
                    ans[k] = i;
                    for (int y = 0; y < m; y++) {
                        avail[y] += alloc[i][y];
                    }
                    f[i] = 1;
                    break;
                }
            }
        }
```

```
    }

    int flag = 1;

    // Check if sequence is safe or not
    for (int i = 0; i < n; i++) {
        if (f[i] == 0) {
            flag = 0;
            printf("The given sequence is not safe\n");
            break;
        }
    }

    if (flag == 1) {
        printf("Following is the SAFE Sequence\n");
        for (int i = 0; i < n - 1; i++) {
            printf(" P%d ->", ans[i]);
        }
        printf(" P%d\n", ans[n - 1]);
    }

    return 0;
}



//OUTPUT:-

avcoe@avcoe-HP-ProDesk-400-G1-SFF:~$ gcc Banker.c
avcoe@avcoe-HP-ProDesk-400-G1-SFF:~$ ./a.out
Following is the SAFE Sequence
 P1 -> P3 -> P0 -> P2 -> P4
```