```
#include <stdio.h>

#include <stdbool.h>

#define MAX 10

#define RESOURCE_TYPES 10

bool request_granted(int n, int m, int request[], int available[], int allocation[]
[RESOURCE_TYPES], int need[][RESOURCE_TYPES], int process_id) {

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

if (request[i] > need[process_id][i]) {

return false;

}

if (request[i] > available[i]) {

return false;

}

}

return true;

}

void update_resources(int n, int m, int request[], int available[], int allocation[]
[RESOURCE_TYPES], int need[][RESOURCE_TYPES], int process_id) {

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

available[i] -= request[i];

allocation[process_id][i] += request[i];

need[process_id][i] -= request[i];

}

}

bool safety_algorithm(int n, int m, int available[], int allocation[]
[RESOURCE_TYPES], int need[][RESOURCE_TYPES], int safe_sequence[]) {

int work[RESOURCE_TYPES];

bool finish[MAX] = {false};
```

```
int count = 0;
for (int i = 0; i < m; i++) {
work[i] = available[i];
}
while (count < n) {
bool progress_made = false;
for (int i = 0; i < n; i++) {
if (!finish[i]) {
bool can_finish = true;
for (int j = 0; j < m; j++) {
if (need[i][j] > work[j]) {
can_finish = false;
break;
}
}
if (can_finish) {
for (int j = 0; j < m; j++) {
work[j] += allocation[i][j];
}
safe_sequence[count++] = i;
finish[i] = true;
progress_made = true;
break;
}
}
}
```

```
if (!progress_made) {
return false;
}
}
return true;
}
int main() {
printf("Yash Sigchi-23BAI1242:\n\n");
int n, m;
printf("Enter the number of processes(23BAI1242): ");
scanf("%d", &n);
printf("Enter the number of resource types(23BAI1242): ");
scanf("%d", &m);
int allocation[MAX][RESOURCE_TYPES], maximum[MAX]
[RESOURCE_TYPES], need[MAX][RESOURCE_TYPES],
available[RESOURCE_TYPES];
int safe_sequence[MAX];
printf("Enter the allocation matrix:\n");
for (int i = 0; i < n; i++) {
printf("Process %d allocation: ", i);
for (int j = 0; j < m; j++) {
scanf("%d", &allocation[i][j]);
}
}
printf("Enter the maximum matrix:\n");
for (int i = 0; i < n; i++) {
printf("Process %d maximum: ", i);
```

```
for (int j = 0; j < m; j++) {
scanf("%d", &maximum[i][j]);
}
}
for (int i = 0; i < n; i++) {
for (int j = 0; j < m; j++) {
need[i][j] = maximum[i][j] - allocation[i][j];
}
}
printf("Enter the available resources: ");
for (int i = 0; i < m; i++) {
scanf("%d", &available[i]);
}
bool is_safe = safety_algorithm(n, m, available, allocation, need, safe_sequence);
if (is_safe) {
printf("The system is in a safe state.\n(23BAI1242)\n");
printf("Safe sequence: ");
for (int i = 0; i < n; i++) {
printf("P%d ", safe_sequence[i]);
}
printf("\n");
} else {
printf("The system is in an unsafe state.\n(23BAI1242)");
}
int process_id;
printf("Enter the process id requesting resources: ");
```

```
scanf("%d", &process_id);
int request[RESOURCE_TYPES];
printf("Enter the resource request for process %d: ", process_id);
for (int i = 0; i < m; i++) {
scanf("%d", &request[i]);
}
if (request_granted(n, m, request, available, allocation, need, process_id)) {
printf("Request can be granted.\n");
update_resources(n, m, request, available, allocation, need, process_id);
is_safe = safety_algorithm(n, m, available, allocation, need, safe_sequence);
if (is_safe) {
printf("The system is still in a safe state after the request.\n");
} else {
printf("The system is in an unsafe state after the request. Rolling back.\n");
for (int i = 0; i < m; i++) {
available[i] += request[i];
allocation[process_id][i] -= request[i];
need[process_id][i] += request[i];
}
}
} else {
printf("Request cannot be granted. It exceeds the need or available
resources.\n");
}
r
```

```
student@admin-55:~/Desktop/23BAI1242/Ex-6$ ./a.out
Yash Sigchi-23BAI1242:

Enter the number of processes(23BAI1242): 5
Enter the number of resource types(23BAI1242): 3
Enter the allocation matrix:
Process 0 allocation: 0 1 0
Process 1 allocation: 2 0 0
Process 2 allocation: 3 0 2
Process 3 allocation: 2 1 1
Process 4 allocation: 0 0 2
Enter the maximum matrix:
Process 0 maximum: 7 5 3
Process 1 maximum: 3 2 2
Process 2 maximum: 9 0 2
Process 3 maximum: 4 2 2
Process 4 maximum: 5 3 3
Enter the available resources: 10 5 7
The system is in a safe state.
(23BAI1242)
Safe sequence: P0 P1 P2 P3 P4
Enter the process id requesting resources: 4
Enter the resource request for process 4: 6 4 10
Request cannot be granted. It exceeds the need or available resources.
student@admin-55:~/Desktop/23BAI1242/Ex-6$
```

eturn 0;

}

```
File  Edit  View  Search  Terminal  Help
student@admin-55:~$ cd Desktop
student@admin-55:~/Desktop$ mkdir 23BAI1242
student@admin-55:~/Desktop$ cd 23BAI1242
student@admin-55:~/Desktop/23BAI1242$ mkdir Ex-6
student@admin-55:~/Desktop/23BAI1242$ cd Ex-6
student@admin-55:~/Desktop/23BAI1242/Ex-6$ touch Bankers.c
student@admin-55:~/Desktop/23BAI1242/Ex-6$ gedit Bankers.c
student@admin-55:~/Desktop/23BAI1242/Ex-6$ gcc Bankers.c
student@admin-55:~/Desktop/23BAI1242/Ex-6$ ./a.out
Yash Sigchi-23BAI1242:

Enter the number of processes(23BAI1242): 5
Enter the number of resource types(23BAI1242): 3
Enter the allocation matrix:
Process 0 allocation: 0 1 0
Process 1 allocation: 2 0 0
Process 2 allocation: 3 0 2
Process 3 allocation: 2 1 1
Process 4 allocation: 0 0 2
Enter the maximum matrix:
Process 0 maximum: 7 5 3
Process 1 maximum: 3 2 2
Process 2 maximum: 9 0 2
Process 3 maximum: 4 2 2
Process 4 maximum: 5 3 3
Enter the available resources: 10 5 7
The system is in a safe state.
(23BAI1242)
Safe sequence: P0 P1 P2 P3 P4
Enter the process id requesting resources: 2
Enter the resource request for process 2: 2 0 0
Request can be granted.
The system is still in a safe state after the request.
student@admin-55:~/Desktop/23BAI1242/Ex-6$
```

```
student@admin-55:~/Desktop/23BAI1242/Ex-6$ ./a.out
Yash Sigchi-23BAI1242:

Enter the number of processes(23BAI1242): 4
Enter the number of resource types(23BAI1242): 3
Enter the allocation matrix:
Process 0 allocation: 0 1 0
Process 1 allocation: 2 0 0
Process 2 allocation: 3 0 2
Process 3 allocation: 2 1 0
Enter the maximum matrix:
Process 0 maximum: 7 5 3
Process 1 maximum: 3 2 2
Process 2 maximum: 7 0 2
Process 3 maximum: 2 2 2
Enter the available resources: 3 3 2
The system is in a safe state.
(23BAI1242)
Safe sequence: P1 P2 P3 P0
Enter the process id requesting resources: 1
Enter the resource request for process 1: 1 2 3
Request cannot be granted. It exceeds the need or available resources.
student@admin-55:~/Desktop/23BAI1242/Ex-6$
```