

OS LAB 07

1) Implement the above code and paste the screen shot of the output.

CODE:

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

```
int current[5][5], maximum_claim[5][5], available[5];
```

```
int allocation[5] = {0};
```

```
int maxres[5], running[5], safe = 0;
```

```
int counter = 0, i, j, exec, resources, processes;
```

```
int main() {
```

```
    printf("\nEnter number of processes: ");
```

```
    scanf("%d", &processes);
```

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

```
        running[i] = 1;
```

```
        counter++;
```

```
    }
```

```
    printf("\nEnter number of resources: ");
```

```
    scanf("%d", &resources);
```

```
    printf("\nEnter Claim Vector:");
```

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

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

```
    }
```

```
    printf("\nEnter Allocated Resource Table:\n");
```

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

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

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

```
        }
```

```
    }
```

```
    printf("\nEnter Maximum Claim Table:\n");
```

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

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

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

```
        }
```

```
    }
```

```
    printf("\nAllocated resources:");
```

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

```
        for (j = 0; j < processes; j++) {
```

```
            allocation[i] += current[j][i];
```

```
        }
```

```
        printf("\t%d", allocation[i]);
```

```
    }
```

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

```
        available[i] = maxres[i] - allocation[i];
```

```
    }
```

```
    printf("\nAvailable resources:");
```

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

```
        printf("\t%d", available[i]);
```

```
    }
```

```
    printf("\n");
```

```
    while (counter != 0) {
```

```
        safe = 0;
```

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

```
            if (running[i]) {
```

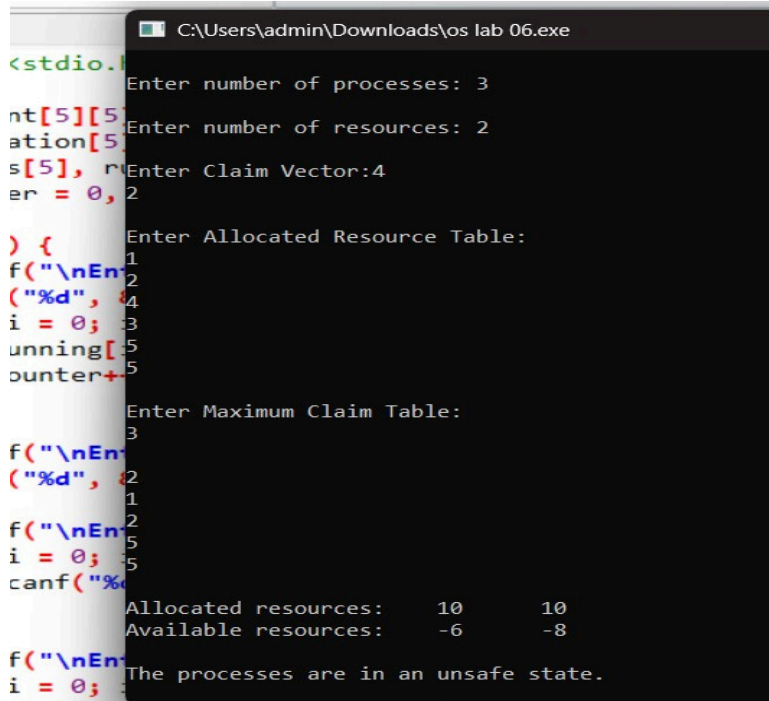
```
                exec = 1;
```

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

OS LAB 07

```
        if (maximum_claim[i][j] - current[i][j] > available[j]) {
            exec = 0;
            break;
        }
    }
    if (exec) {
        printf("\nProcess %d is executing\n", i + 1);
        running[i] = 0;
        counter--;
        safe = 1;
        for (j = 0; j < resources; j++) {
            available[j] += current[i][j];
        }
        break;
    }
}
}
if (!safe) {
    printf("\nThe processes are in an unsafe state.\n");
    break;
} else {
    printf("\nThe process is in a safe state\n");
    printf("Available vector:");
    for (i = 0; i < resources; i++) {
        printf("\t%d", available[i]);
    }
    printf("\n");
}
}
return 0;
}
```

OUTPUT



```
C:\Users\admin\Downloads\os lab 06.exe
Enter number of processes: 3
Enter number of resources: 2
Enter Claim Vector:4
Enter Allocated Resource Table:
1
2
4
3
5
5
Enter Maximum Claim Table:
3
2
2
5
5
Allocated resources:    10    10
Available resources:   -6    -8
The processes are in an unsafe state.
```