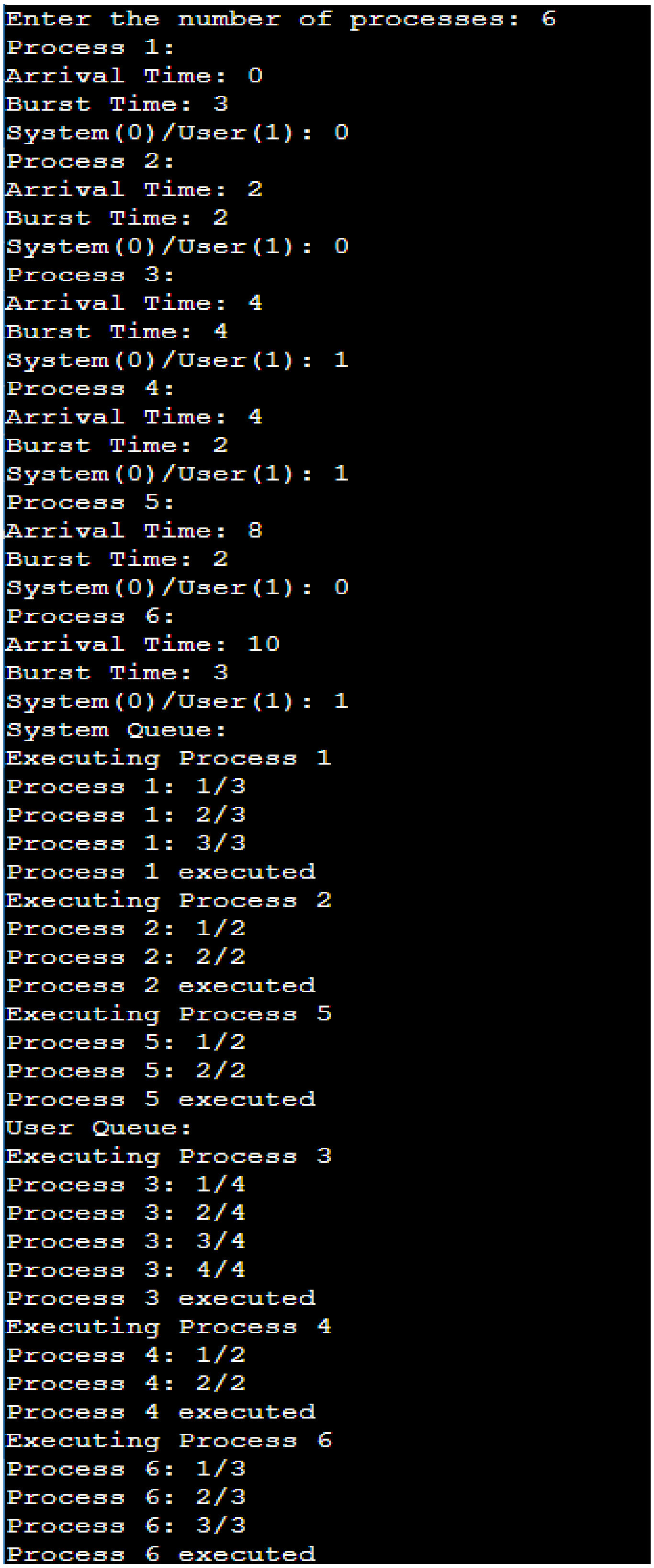
Multilevel Sheduling

```
#include <stdio.h>
#define MAX_QUEUE_SIZE 100

typedef struct {
    int processID;
    int arrivalTime;
    int burstTime;
    int priority;
} Process;

void executeProcess(Process process) {
    printf("Executing Process %d\n", process.processID);
    for (int i = 1; i <= process.burstTime; i++) {
        printf("Process %d: %d/%d\n", process.processID, i, process.burstTime);
    }
    printf("Process %d executed\n", process.processID);
}

void scheduleFCFS(Process queue[], int size) {
    for (int i = 0; i < size; i++) {
        executeProcess(queue[i]);
    }
}

int main() {
    int numProcesses;
    Process processes[MAX_QUEUE_SIZE];
    printf("Enter the number of processes: ");
    scanf("%d", &numProcesses);
    for (int i = 0; i < numProcesses; i++) {
        printf("Process %d:\n", i + 1);
        printf("Arrival Time: ");
        scanf("%d",&processes[i].arrivalTime);
        printf("Burst Time: ");
        scanf("%d",&processes[i].burstTime);
        printf("System(0)/User(1): ");
        scanf("%d",&processes[i].priority);
        processes[i].processID = i + 1;
    }
    Process systemQueue[MAX_QUEUE_SIZE];
    int systemQueueSize = 0;
    Process userQueue[MAX_QUEUE_SIZE];
    int userQueueSize = 0;
    for (int i = 0; i < numProcesses; i++) {
        if (processes[i].priority == 0) {
            systemQueue[systemQueueSize++] = processes[i];
        } else {
            userQueue[userQueueSize++] = processes[i];
        }
    }
    printf("System Queue:\n");
    scheduleFCFS(systemQueue, systemQueueSize);
    printf("User Queue:\n");
    scheduleFCFS(userQueue, userQueueSize);
    return 0;
}
```

OUTPUT :



```
Enter the number of processes: 6
Process 1:
Arrival Time: 0
Burst Time: 3
System(0)/User(1): 0
Process 2:
Arrival Time: 2
Burst Time: 2
System(0)/User(1): 0
Process 3:
Arrival Time: 4
Burst Time: 4
System(0)/User(1): 1
Process 4:
Arrival Time: 4
Burst Time: 2
System(0)/User(1): 1
Process 5:
Arrival Time: 8
Burst Time: 2
System(0)/User(1): 0
Process 6:
Arrival Time: 10
Burst Time: 3
System(0)/User(1): 1
System Queue:
Executing Process 1
Process 1: 1/3
Process 1: 2/3
Process 1: 3/3
Process 1 executed
Executing Process 2
Process 2: 1/2
Process 2: 2/2
Process 2 executed
Executing Process 5
Process 5: 1/2
Process 5: 2/2
Process 5 executed
User Queue:
Executing Process 3
Process 3: 1/4
Process 3: 2/4
Process 3: 3/4
Process 3: 4/4
Process 3 executed
Executing Process 4
Process 4: 1/2
Process 4: 2/2
Process 4 executed
Executing Process 6
Process 6: 1/3
Process 6: 2/3
Process 6: 3/3
Process 6 executed
```