5) Write a C program to simulate multi-level queue scheduling algorithm considering the following scenario. All the processes in the system are divided into two categories - system processes and user processes. System processes are to be given higher priority than user processes. Use FCFS scheduling for the processes in each queue.

```
#include<stdio.h>
void main()
{
  int p[20], bt[20], su[20], wt[20], tat[20], at[20];
  int ct[20], i, k, n, temp;
  float wtavg, tatavg;
  printf(" Enter the numbers of processes --- ");
  scanf("%d", &n);
  for(i=0; i<n; i++)
  {
    p[i] = i;
    printf(" enter the arrival time of process %d --- ", i+1);
    scanf("%d", &at[i]);
    printf(" enter the burst time of process %d --- ", i+1);
    scanf("%d", &bt[i]);
    printf(" Queue (0/1) ? --- ");
    scanf("%d", &su[i]);
  }
  // sort based on arrival time first, then by
     queue process if arrival times are the same
  for(i=0; i<n; i++)
  {
    for(k=i+1; k<n; k++) {
```

```
if (at[i] > at[k] || (at[i] == at[k] && pr[i] > pr[k]))
{
    // swap process ids
    temp = p[i];
    p[i] = p[k];
    p[k] = temp;

    // swap burst times
    temp = bt[i];
    bt[i] = bt[k];
    bt[k] = temp;

    // swap Queue identifiers
    temp = bt[i];
    bt[i] = bt[k];
    bt[k] = temp;

    // swap arrival times
    temp = at[i];
    at[i] = at[k];
    at[k] = temp;
  }
 }
}
wtavg = wt[0] = 0;
tatavg = tat[0] = bt[0];
ct[0] = at[0] + bt[0];
for(i=1; i<n; i++){
 if (ct[i-1] < at[i]){
  ct[i] = at[i] + bt[i];
 }
```

```
else{
    ct[i] = ct[i-1] + bt[i];
  }
wt[i] = ct[i] - at[i] - bt[i];
tat[i] = ct[i] - at[i];
wtavg += wt[i];
tatavg += tat[i];
}
printf("\n Process\t\t Arrival time\t System/user process\t
        Burst time\t waiting time\t turnaround time");
for(i=0; i<n; i++){
printf("\n %d\t\t %d\t\t %d\t\t %d\t\t %d\t\t %d", p[i], at[i], su[i], bt[i], wt[i], tat[i]);
}
printf("\n Average Waiting Time is --- %f", wtavg/n);
printf("\n Average Turnaround time is --- %f", tatavg/n);
}
```

Output:

```
Enter the number of processes --- 3
Enter the Arrival time of process --- 0
Enter the Burst time of process --- 2
Queue or not? --- 1
Enter the Arrival time of process --- 2
Enter the Burst time of process --- 3
Queue or not? --- 1
Enter the Arrival time of process --- 3
Enter the Burst time of process --- 4
Queue or not? --- 0
```

Process	Arrival Time	System/User Process	BT	WT	TAT
0	0	1	2	0	2
1	2	1	3	0	3
2	3	0	4	2	6

Average Waiting Time is ---0.666667

Average Turnaround Time is ---3.666667

6/6/24