LAB 5: PROCESS SCHEDULING ALGORITHMS

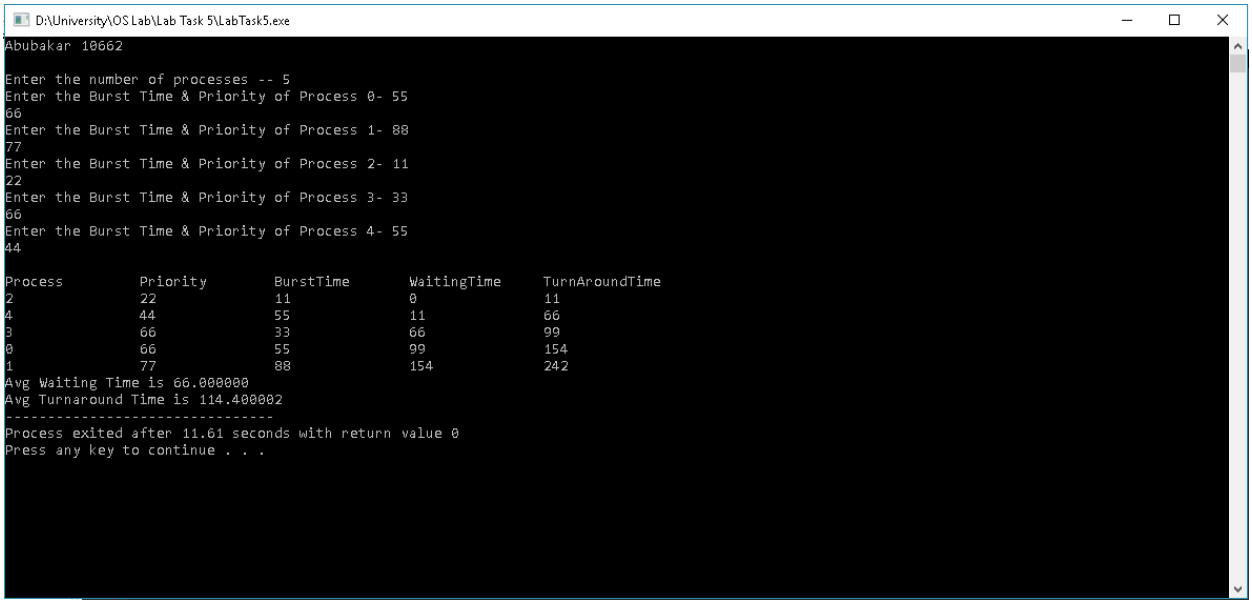
Abubakar Jamil 10662

Q1:

Code:

```c
//Abubakar 10662
int main() {
    printf("Abubakar 10662\n\n");
    int p[20],bt[20],pri[20], wt[20],tat[20],i, k, n, temp;
    float wtavg, tatavg;
    printf("Enter the number of processes -- ");
    scanf("%d",&n);
    for(i=0;i<n;i++) {
        p[i] = i;
        printf("Enter the Burst Time & Priority of Process %d- ",i);
        scanf("%d %d",&bt[i], &pri[i]); }
    for(i=0;i<n;i++)
        for(k=i+1;k<n;k++)
            if(pri[i] > pri[k]) {
                temp=p[i];
                p[i]=p[k];
                p[k]=temp;
                temp=bt[i];
                bt[i]=bt[k];
                bt[k]=temp;
                temp=pri[i];
                pri[i]=pri[k];
                pri[k]=temp;
}
wtavg = wt[0] = 0;
tatavg = tat[0] = bt[0];
for(i=1;i<n;i++) {
    wt[i] = wt[i-1]+ bt[i-1];
    tat[i] = tat[i-1] + bt[i];
    wtavg = wtavg + wt[i];
    tatavg = tatavg + tat[i];
}
printf("\nProcess\t\tPriority\tBurstTime\tWaitingTime\tTurnAroundTime");
for(i=0;i<n;i++)
    printf("\n%d\t\t%d\t\t%d\t\t%d\t\t%d ",p[i],pri[i],bt[i],wt[i],tat[i]);
printf("\nAvg Waiting Time is %f",wtavg/n);
printf("\nAvg Turnaround Time is %f",tatavg/n); }
```

Output:



X----------X----------X

Q2: Which of the following process scheduling algorithm may lead to starvation?

a) FIFO.

b) Round Robin.

c) Shortest Job First.

d) None of the Above.

Explanation:

Shortest job next may lead to process starvation for processes which will require a long time to complete if short processes are continually added.