OPERATING SYSTEM - CS23431

EXP 6(A)

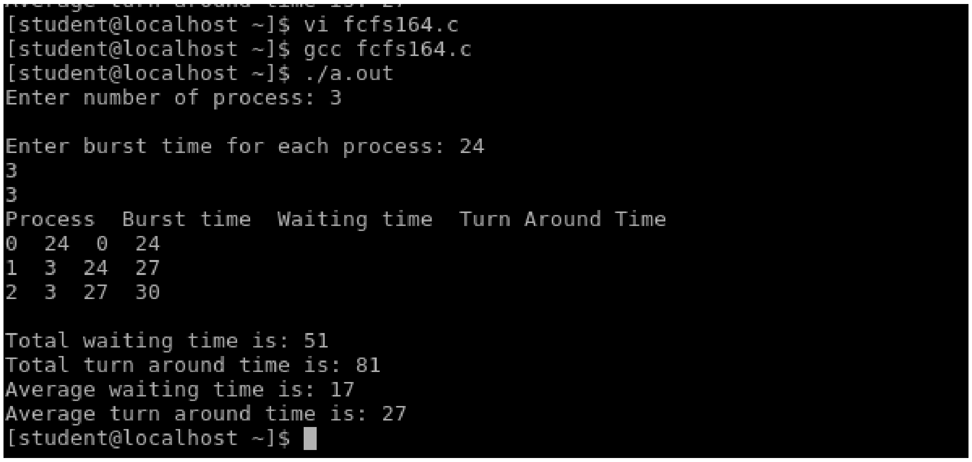
FIRST COME FIRST SERVE

NAME: S.Manicka Meenakshi **ROLL NO: 230701173**

PROGRAM:

```
#include <stdio.h>

int main() {

int n, i; printf("Enter number of processes: "); scanf("%d", &n);

int bt[n], wt[n], tat[n];

printf("\nEnter burst time for each process:\n");
for (i = 0; i < n; i++) {
   printf("P[%d]: ", i + 1);
   scanf("%d", &bt[i]);
}

wt[0] = 0;
for (i = 1; i < n; i++) {
   wt[i] = wt[i - 1] + bt[i - 1];
}

for (i = 0; i < n; i++) {
   tat[i] = bt[i] + wt[i];
}

int total_wt = 0, total_tat = 0;
for (i = 0; i < n; i++) {
   total_wt += wt[i];
   total_tat += tat[i];
}

printf("\nProcess\tBurst Time\tWaiting Time\tTurnaround Time\n");
for (i = 0; i < n; i++) {
   printf("P[%d]\t%5d\t\t%5d\t\t%5d\n",
        i + 1, bt[i], wt[i], tat[i]);
}


printf("\nTotal waiting time     = %d\n", total_wt);
printf("Total turnaround time = %d\n", total_tat);
printf("Average waiting time     = %.2f\n", (float)total_wt / n);
printf("Average turnaround time = %.2f\n", (float)total_tat / n);

return 0;


}
```

OUTPUT:



```
Average turn around time is: 27
[student@localhost ~]$ vi fcfs164.c
[student@localhost ~]$ gcc fcfs164.c
[student@localhost ~]$ ./a.out
Enter number of process: 3

Enter burst time for each process: 24
3
3
Process  Burst time  Waiting time  Turn Around Time
0  24  0  24
1  3  24  27
2  3  27  30

Total waiting time is: 51
Total turn around time is: 81
Average waiting time is: 17
Average turn around time is: 27
[student@localhost ~]$
```

OPERATING SYSTEM - CS23431

EXP 6(B)

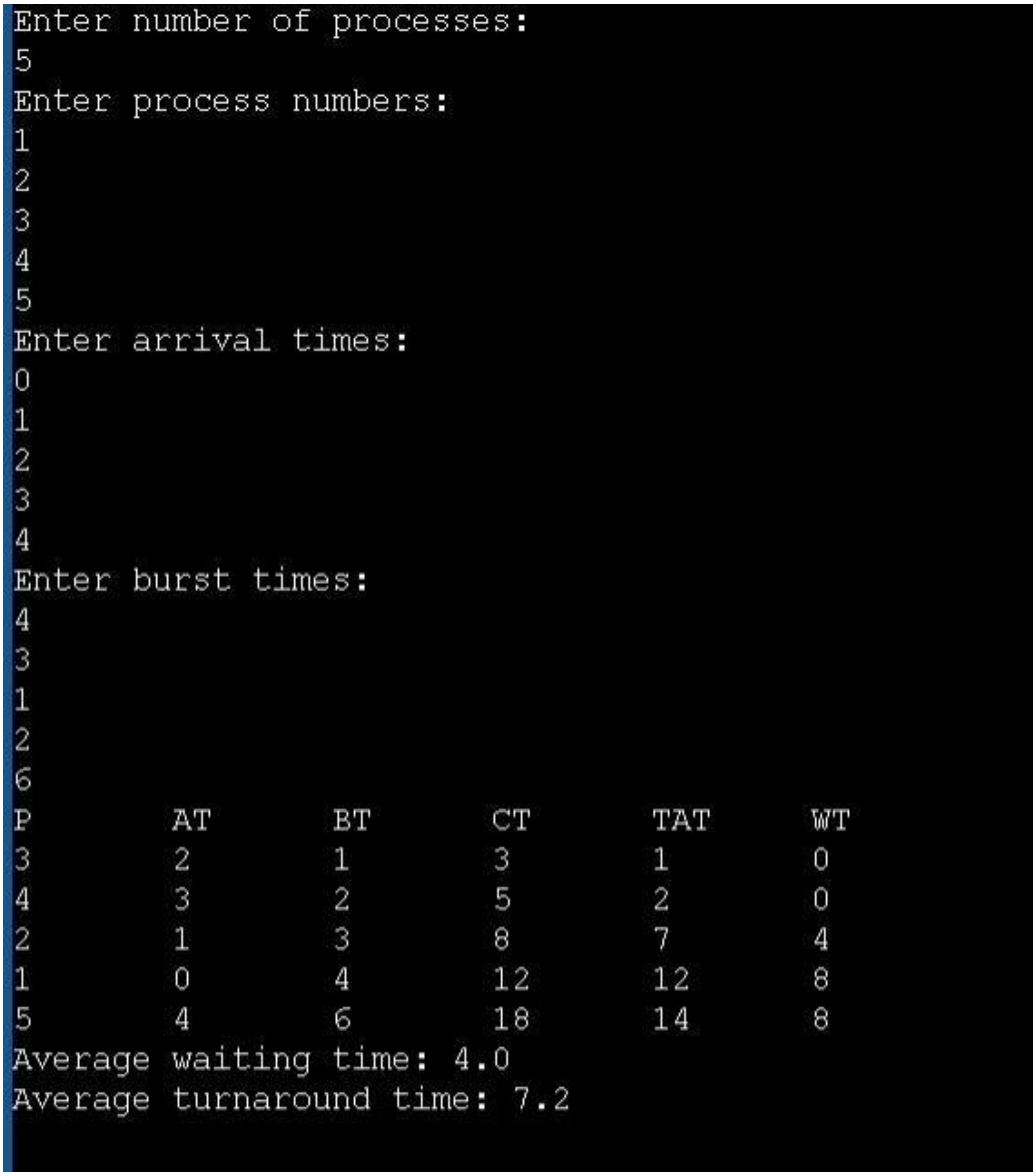
SHORTEST JOB FIRST

NAME: S.Manicka Meenakshi **ROLL NO: 230701173**

PROGRAM:

```
#include <stdio.h>

int main() {
    int n, totalwt = 0, totaltat = 0;

    printf("Enter number of processes: \n");
    scanf("%d", &n);

    int p[n], at[n], bt[n], ct[n], tat[n], wt[n];

    printf("Enter process numbers: \n");
    for (int i = 0; i < n; i++) {
        scanf("%d", &p[i]);
    }

    printf("Enter arrival times: \n");
    for (int i = 0; i < n; i++) {
        scanf("%d", &at[i]);
    }

    printf("Enter burst times: \n");
    for (int i = 0; i < n; i++) {
        scanf("%d", &bt[i]);
    }

    int temp;
    for (int i = 0; i < n - 1; i++) {
        for (int j = i + 1; j < n; j++) {
            if ((bt[j] < bt[i]) || (bt[j] == bt[i] && at[j] < at[i])) {
                temp = at[i];
                at[i] = at[j];
                at[j] = temp;

                temp = bt[i];
                bt[i]  =  bt[j];
                bt[j] = temp;

                temp = p[i];
                p[i] = p[j];
                p[j] = temp;
            }
        }
    }

    ct[0] = at[0] + bt[0];
    for (int i = 1; i < n; i++) {
        if (at[i] > ct[i - 1]) {
            ct[i] = at[i] + bt[i];
        } else {
            ct[i] = ct[i - 1] + bt[i];
        }
    }

    for (int i = 0; i < n; i++) {
        tat[i] = ct[i] - at[i];
        totaltat += tat[i];

        wt[i] = tat[i] - bt[i];
        totalwt += wt[i];
    }

    printf("P\tAT\tBT\tCT\tTAT\tWT\n");
    for (int i = 0; i < n; i++) {
        printf("%d\t%d\t%d\t%d\t%d\t%d\n", p[i], at[i], bt[i], ct[i], tat[i], wt[i]);
    }

    printf("Average waiting time: %.1f\n", (float)(totalwt) / n);
    printf("Average turnaround time: %.1f\n", (float)(totaltat) / n);

    return 0;
}
```

OUTPUT:



```
Enter number of processes:
5
Enter process numbers:
1
2
3
4
5
Enter arrival times:
0
1
2
3
4
Enter burst times:
4
3
1
2
6
P       AT      BT      CT      TAT     WT
3       2       1       3       1       0
4       3       2       5       2       0
2       1       3       8       7       4
1       0       4       12      12      8
5       4       6       18      14      8
Average waiting time: 4.0
Average turnaround time: 7.2
```

OPERATING SYSTEM - CS23431

EXP 6(C)

PRIORITY SCHEDULING

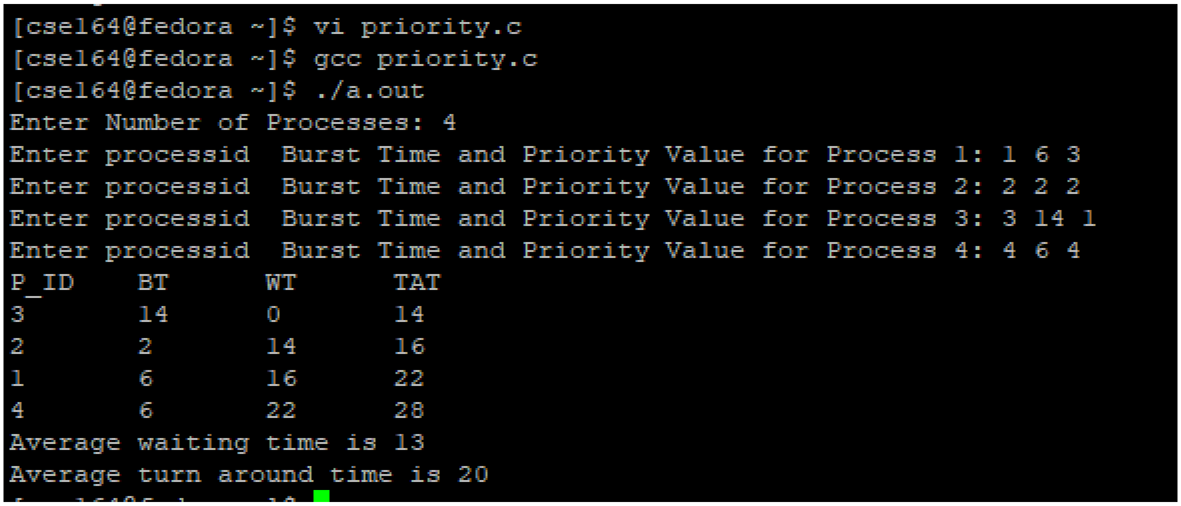
NAME: S.Manicka Meenakshi **ROLL NO: 230701173**

PROGRAM:

```
#include <stdio.h>

int main() {

int n; printf("Enter Number of Processes: "); scanf("%d", &n);

int pid[n], b[n], p[n];
for (int i = 0; i < n; i++) {
   printf("Enter processid  Burst Time and Priority Value for Process %d: ", i + 1);
   scanf("%d %d %d", &pid[i], &b[i], &p[i]);
}

for (int i = 0; i < n; i++) {
   int max_priority = p[i];
   int max_index = i;
   int swapped = 0;

    for (int j = i + 1; j < n; j++) {
      if (p[j] < max_priority) {
          max_priority = p[j];
           max_index = j;
           swapped = 1;
       }
   }

   if (swapped) {
       int temp = p[i];
       p[i] = p[max_index];
       p[max_index] = temp;

       temp = b[i];
       b[i] = b[max_index];
```

```
        b[max_index] = temp;


        temp = pid[i];
        pid[i] = pid[max_index];
        pid[max_index] = temp;
    }
}


int wait_time = 0, totalwt = 0, totalturn = 0;
printf("P_ID\tBT\tWT\tTAT\n");
for (int i = 0; i < n; i++) {
    int tat = wait_time + b[i];
    printf("%d\t%d\t%d\t%d\n", pid[i], b[i], wait_time, tat);
    totalwt += wait_time;
    totalturn += tat;
    wait_time += b[i];
}


printf("Average waiting time is %d\n", totalwt / n);
printf("Average turn around time is %d\n", totalturn / n);

return 0;


}
```

OUTPUT:



```
[cse164@fedora ~]$ vi priority.c
[cse164@fedora ~]$ gcc priority.c
[cse164@fedora ~]$ ./a.out
Enter Number of Processes: 4
Enter processid  Burst Time and Priority Value for Process 1: 1 6 3
Enter processid  Burst Time and Priority Value for Process 2: 2 2 2
Enter processid  Burst Time and Priority Value for Process 3: 3 14 1
Enter processid  Burst Time and Priority Value for Process 4: 4 6 4
P_ID    BT      WT      TAT
3       14      0       14
2       2       14      16
1       6       16      22
4       6       22      28
Average waiting time is 13
Average turn around time is 20
```

OPERATING SYSTEM - CS23431

EXP 6(D)

ROUND ROBIN CHEDULING

NAME: S.Manicka Meenakshi **ROLL NO: 230701173**

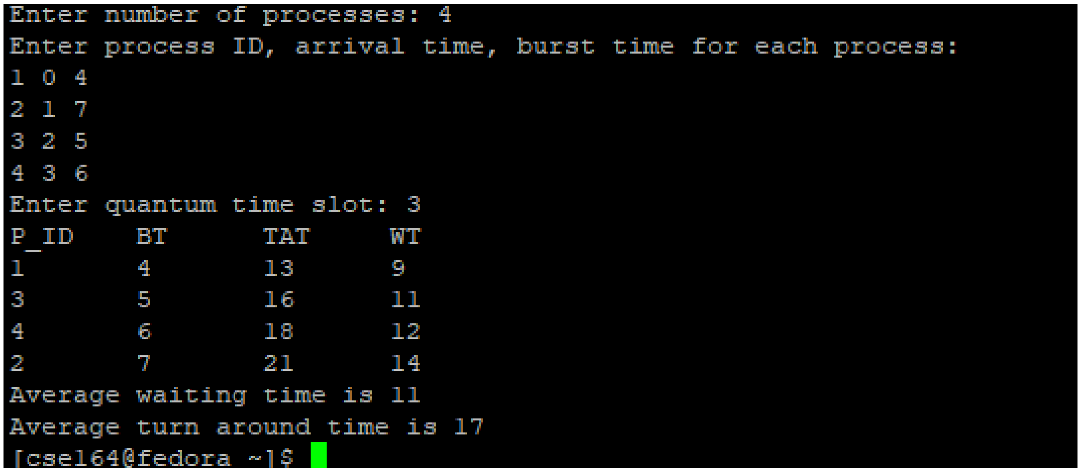
PROGRAM:

```
#include <stdio.h>

int main() {
    int n;
    printf("Enter number of processes: ");
    scanf("%d", &n);

    int p[n], a[n], bt[n], temptbt[n], slot;
    printf("Enter process ID, arrival time, burst time for each process:\n");
    for (int i = 0; i < n; i++) {
        scanf("%d %d %d", &p[i], &a[i], &bt[i]);
        temptbt[i] = bt[i];
    }

    printf("Enter quantum time slot: ");
    scanf("%d", &slot);

    int totalwt = 0, totalturn = 0, totaltime = 0;
    int i = 0, count = 0, completed = 0;

    printf("P_ID\tBT\tTAT\tWT\n");
```

```
while (completed != n) {
    if (temptbt[i] <= slot && temptbt[i] > 0) {
        totaltime += temptbt[i];
        temptbt[i] = 0;
        count = 1;
    }
    else if (temptbt[i] > 0) {
        totaltime += slot;
        temptbt[i] -= slot;
    }

    if (temptbt[i] == 0 && count == 1) {
        completed++;
        int tat = totaltime - a[i];
        int wt  = totaltime - a[i] - bt[i];
        printf("%d\t%d\t%d\t%d\n", p[i], bt[i], tat, wt);
        totalwt += wt;
        totalturn += tat;
        count = 0;
    }

    if (i == n - 1)
        i = 0;
    else
        i++;
}
```

```
    printf("Average waiting time is %d\n", totalwt / n);

    printf("Average turn around time is %d\n", totalturn / n);



    return 0;

}
```

OUTPUT:



```
Enter number of processes: 4
Enter process ID, arrival time, burst time for each process:
1 0 4
2 1 7
3 2 5
4 3 6
Enter quantum time slot: 3
P_ID    BT      TAT     WT
1       4       13      9
3       5       16      11
4       6       18      12
2       7       21      14
Average waiting time is 11
Average turn around time is 17
[cse164@fedora ~]$
```