

## **EXPERIMENT: 4**

**NAME: KRISHNA KUMAR MAHTO**

**REG. NO. :- 16BIT0453**



### **First come first serve algorithm**

```
#include <stdio.h>
#include <conio.h>
int wt[10], bt[10], at[10], tat[10], n;
float awt, atat;
void input() {
    printf("Enter Number of processes:");
    scanf("%d", &n);
    inti;
    for(i=0; i<n; i++)
    {
        printf("Enter Burst Time of process %d:", i+1);
        scanf("%d", &bt[i]);
        printf("Enter Arrival Time of process %d:", i+1);
        scanf("%d", &at[i]);
    }
}
void calculate() {
    wt[0]=0;
    atat=tat[0]=bt[0];
    int btt=bt[0];
    inti;
    for(i=1; i<n; i++) { wt[i]=btt-at[i];
        btt+=bt[i];
        awt+=wt[i];
        tat[i]= wt[i]+bt[i];
        atat+=tat[i];
    }
    atat/=n;
    awt/=n;
}
void display() {
    inti;
    printf("SR.\tA.T.\tB.T.\tW.T.\tT.A.T.\n");
    for(i=0; i<n; i++)
    {
        printf("%3d\t%3d\t%3d\t%3d\t%4d\n", i+1, at[i], bt[i], wt[i], tat[i]);
    }
    printf("Average Waiting Time: %f\nAverage Turn Around Time:%f", awt, atat);
}
int main() {
    input();
    calculate();
    display();
    getch();
}
```

```
fcfs.c:(.text+0x276): undefined reference to `getch'
collect2: ld returned 1 exit status
ubuntu@ubuntu:~$ vi fcfs.c
ubuntu@ubuntu:~$ cc fcfs.c
ubuntu@ubuntu:~$ ./a.out;
Enter Number of processes:5
Enter Burst Time of process 1:5
Enter Arrival Time of process 1:1
Enter Burst Time of process 2:3
Enter Arrival Time of process 2:4
Enter Burst Time of process 3:6
Enter Arrival Time of process 3:0
Enter Burst Time of process 4:4
Enter Arrival Time of process 4:2
Enter Burst Time of process 5:2
Enter Arrival Time of process 5:3
SR.      A.T.      B.T.      W.T.      T.A.T.
 1         1         5         0         5
 2         4         3         1         4
 3         0         6         8        14
 4         2         4        12        16
 5         3         2        15        17
Average Waiting Time: 7.200000
Average Turn Around Time:11.200000ubuntu@ubuntu:~$
```

### Shortest job first algorithm non-preemptive

```
#include<stdio.h>
intmain()
{
inttime, bt[10], at[10], sum_bt=0, smallest, n, i;
intsum_turnaround=0, sum_wait=0;
printf("Enter no of processes : ");
scanf("%d", &n); for(i=0; i<n; i++)
{
printf("Enter arrival time for process P%d : ", i+1);
scanf("%d", &at[i]);
printf("Enter burst time for process P%d : ", i+1);
scanf("%d", &bt[i]);
sum_bt+=bt[i];
}
bt[9]=9999;
printf("\n\nProcess\t|Turnaround Time| Waiting Time\n\n");
for(time=0; time<sum_bt; )
{
smallest=9;
for(i=0; i<n; i++)
{
if(at[i]<=time &&bt[i]>0 &&bt[i]<bt[smallest])
smallest=i;
}
if(smallest==9)
```

```

{
time++;
continue;
}
printf("P[%d]\t\t%d\t\t%d\n",smallest+1,time+bt[smallest]-at[smallest],time-at[smallest]);
sum_turnaround+=time+bt[smallest]-at[smallest];
sum_wait+=time-at[smallest];
time+=bt[smallest];
bt[smallest]=0;
}
printf("\n\n average waiting time = %f",sum_wait*1.0/n);printf("\n\n average turnaround time = %f",sum_turnaround*1.0/n);
return 0;
}

```

```

File Edit View Terminal Help
Average Waiting Time: 7.200000
Average Turn Around Time:11.200000ubuntu@ubuntu:~$ vi sjf.c
ubuntu@ubuntu:~$ cc sjf.c
ubuntu@ubuntu:~$ ./a.out
Enter no of processes : 5
Enter arrival time for process P1 : 1
Enter burst time for process P1 : 5
Enter arrival time for process P2 : 4
Enter burst time for process P2 : 3
Enter arrival time for process P3 : 0
Enter burst time for process P3 : 6
Enter arrival time for process P4 : 2
Enter burst time for process P4 : 4
Enter arrival time for process P5 : 3
Enter burst time for process P5 : 2

I

Process |Turnaround Time| Waiting Time
P[3]    |      6      |      0
P[5]    |      5      |      3
P[2]    |      7      |      4
P[4]    |     13      |      9
P[1]    |     19      |     14

average waiting time = 6.000000
ubuntu@ubuntu:~$

```

### Shortest remaining time first algorithm(preemptive)

```

#include<stdio.h>
int main()
{
int at[10],bt[10],rt[10],endTime,i,smallest;
int remain=0,n,time,sum_wait=0,sum_turnaround=0;
printf("Enter no of Processes : ");
scanf("%d",&n);
for(i=0;i<n;i++)
{printf("Enter arrival time for Process P%d : ",i+1);
scanf("%d",&at[i]);
printf("Enter burst time for Process P%d : ",i+1);
scanf("%d",&bt[i]);
rt[i]=bt[i];
}

```

```

printf("\n\nProcess\t|Turnaround Time| Waiting Time\n\n");
rt[9]=9999;
for(time=0;remain!=n;time++)
{
smallest=9;
for(i=0;i<n;i++)
{
if(at[i]<=time && rt[i]<rt[smallest] && rt[i]>0)
{
smallest=i;
}
}
rt[smallest]--;
if(rt[smallest]==0)
{
remain++;
endTime=time+1;
printf("\nP[%d]\t|\t%d\t|\t%d",smallest+1,endTime-at[smallest],endTime-bt[smallest]-
at[smallest]);
sum_wait+=endTime-bt[smallest]-at[smallest];
sum_turnaround+=endTime-at[smallest];
}
}
printf("\n\nAverage waiting time = %f\n",sum_wait*1.0/n);
printf("Average Turnaround time = %f",sum_turnaround*1.0/5);return 0;
}

```

```

Avg waiting time = 7.200000
Avg turnaround time = 11.200000ubuntu@ubuntu:~$ vi round.c
ubuntu@ubuntu:~$ vi srtf.c
ubuntu@ubuntu:~$ cc srtf.c
ubuntu@ubuntu:~$ ./a.out
Enter no of Processes : 5
Enter arrival time for Process P1 : 0
Enter burst time for Process P1 : 5
Enter arrival time for Process P2 : 1
Enter burst time for Process P2 : 3
Enter arrival time for Process P3 : 3
Enter burst time for Process P3 : 2
Enter arrival time for Process P4 : 2
Enter burst time for Process P4 : 6
Enter arrival time for Process P5 : 4
Enter burst time for Process P5 : 6

Process |Turnaround Time| Waiting Time

P[2]    |      3      |      0
P[3]    |      3      |      1
P[1]    |     10      |      5
P[4]    |     14      |      8
P[5]    |     18      |     12

Average waiting time = 5.200000

```

## Preemptive priority algorithm

```
#include<stdio.h>
intmain()
{inti,j,n,time,sum_wait=0,sum_turnaround=0,smallest;
int at[10],bt[10],pt[10],rt[10],remain;
printf("Enter no of Processes : ");
scanf("%d",&n);
remain=n;
for(i=0;i<n;i++)
{
printf("Enter arrival time, burst time and priority for process p%d :",i+1);
scanf("%d",&at[i]);
scanf("%d",&bt[i]);
scanf("%d",&pt[i]);
rt[i]=bt[i];
}
pt[9]=11;
printf("\n\nProcess\t|Turnaroundtime|waiting time\n");
for(time=0;remain!=0;time++)
{
smallest=9;
for(i=0;i<n;i++)
{
if(at[i]<=time &&pt[i]<pt[smallest] &&rt[i]>0)
{
smallest=i;
}
}
rt[smallest]--;
if(rt[smallest]==0)
{
remain--;
printf("P[%d]\t|t%d\t|t%d\n",smallest+1,time+1-at[smallest],time+1-at[smallest]-
bt[smallest]);sum_wait+=time+1-at[smallest];
sum_turnaround+=time+1-at[smallest]-bt[smallest];
}
}
printf("\nAvg waiting time = %f\n",sum_wait*1.0/n);
printf("Avg turnaround time = %f",sum_turnaround*1.0/n);
return 0;
}
```

```

ubuntu@ubuntu:~$ vi pre_pri.c
ubuntu@ubuntu:~$ cc pre_pri.c
ubuntu@ubuntu:~$ ./a.out
Enter no of Processes : 4
Enter arrival time, burst time and priority for process p1 :1 2 2
Enter arrival time, burst time and priority for process p2 :2 0 4
Enter arrival time, burst time and priority for process p3 :0 3 1
Enter arrival time, burst time and priority for process p4 :3 1 3

Process | Turnaround time | waiting time
P[3]    | 3               | 0
P[1]    | 4               | 2
P[4]    | 3               | 2
P[10]   | -134511292     | -134511293

Avg waiting time = -33627820.500000
Avg turnaround time = -33627822.250000ubuntu@ubuntu:~$

```

## Non-preemptive priority algorithm

```

#include<stdio.h>
intmain()
{
    inti,j,n,time,sum_wait=0,sum_turnaround=0;
    intsmallest,at[10],bt[10],priority[10],remain;
    printf("Enter no of Processes : ");
    scanf("%d",&n);
    remain=n;for(i=0;i<n;i++)
    {
        printf("Enter arrival time, burst time and priority for process p%d :",i+1);
        scanf("%d",&at[i]);
        scanf("%d",&bt[i]);
        scanf("%d",&priority[i]);
    }
    priority[9]=11;
    printf("\n\nProcess\t|Turnaroundtime|waiting time\n");
    for(time=0;remain!=0;)
    {
        smallest=9;
        for(i=0;i<n;i++)
        {
            if(at[i]<=time && priority[i]<priority[smallest] &&bt[i]>0)
            {
                smallest=i;
            }
        }
        time+=bt[smallest];
        remain--;
        printf("P[%d]\t|t%d\t|t%d\n",smallest+1,time-at[smallest],time-at[smallest]-bt[smallest]);
        sum_wait+=time-at[smallest]-bt[smallest];
        sum_turnaround+=time-at[smallest];
        bt[smallest]=0;
    }
}

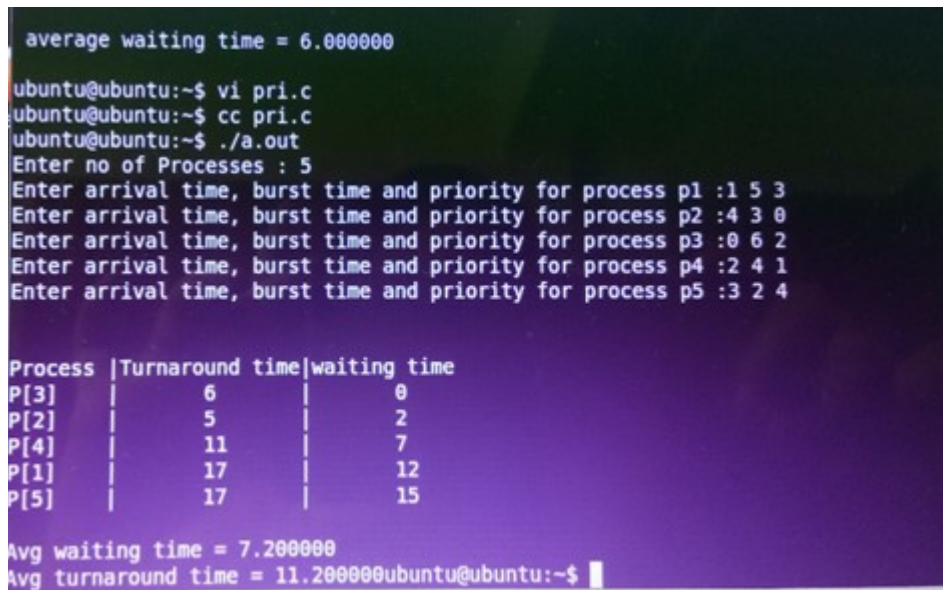
```



```

printf("\nAvg waiting time = %f\n",sum_wait*1.0/n);
printf("Avg turnaround time = %f",sum_turnaround*1.0/n);
return 0;
}

```



```

average waiting time = 6.000000

ubuntu@ubuntu:~$ vi pri.c
ubuntu@ubuntu:~$ cc pri.c
ubuntu@ubuntu:~$ ./a.out
Enter no of Processes : 5
Enter arrival time, burst time and priority for process p1 :1 5 3
Enter arrival time, burst time and priority for process p2 :4 3 0
Enter arrival time, burst time and priority for process p3 :0 6 2
Enter arrival time, burst time and priority for process p4 :2 4 1
Enter arrival time, burst time and priority for process p5 :3 2 4

Process | Turnaround time | waiting time
P[3]    | 6               | 0
P[2]    | 5               | 2
P[4]    | 11              | 7
P[1]    | 17              | 12
P[5]    | 17              | 15

Avg waiting time = 7.200000
Avg turnaround time = 11.200000ubuntu@ubuntu:~$

```

## Round robin algorithm

```

#include<stdio.h>
int main()
{
    int count,j,n,time,remain,flag=0,time_quantum;
    int wait_time=0,turnaround_time=0,at[10],bt[10],rt[10];
    printf("Enter Total Process:\t ");
    scanf("%d",&n);
    remain=n;
    for(count=0;count<n;count++)
    {
        printf("Enter Arrival Time and Burst Time for Process Process Number %d :",count+1);
        scanf("%d",&at[count]);
        scanf("%d",&bt[count]);
        rt[count]=bt[count];
    }
    printf("Enter Time Quantum:\t");scanf("%d",&time_quantum);
    printf("\n\nProcess\t|TurnaroundTime|Waiting Time\n\n");
    for(time=0,count=0;remain!=0;)
    {
        if(rt[count]<=time_quantum&&rt[count]>0)
        {
            time+=rt[count];
            rt[count]=0;
            flag=1;
        }
        else if(rt[count]>0)
        {
            rt[count]-=time_quantum;

```

```
File Edit View Terminal Help
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$ cc round.c
ubuntu@ubuntu:~$ ./a.out
Enter Total Process: 4
Enter Arrival Time and Burst Time for Process Process Number 1 :1 24
Enter Arrival Time and Burst Time for Process Process Number 2 :0 3
Enter Arrival Time and Burst Time for Process Process Number 3 :2
4
Enter Arrival Time and Burst Time for Process Process Number 4 :3 5
Enter Time Quantum: 3

Process |Turnaround Time|Waiting Time
P[2] | 6 | 3
P[3] | 14 | 10
P[4] | 15 | 10
P[1] | 35 | 11

Average Waiting Time= 8.500000
Avg Turnaround Time = 17.500000ubuntu@ubuntu:~$
```

```
File Edit View Terminal Help
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$
ubuntu@ubuntu:~$ cc round.c
ubuntu@ubuntu:~$ ./a.out
Enter Total Process: 4
Enter Arrival Time and Burst Time for Process Process Number 1 :1 24
Enter Arrival Time and Burst Time for Process Process Number 2 :0 3
Enter Arrival Time and Burst Time for Process Process Number 3 :2
4
Enter Arrival Time and Burst Time for Process Process Number 4 :3 5
Enter Time Quantum: 3

Process |Turnaround Time|Waiting Time
P[2] | 6 | 3
P[3] | 14 | 10
P[4] | 15 | 10
P[1] | 35 | 11

Average Waiting Time= 8.500000
Avg Turnaround Time = 17.500000ubuntu@ubuntu:~$
```