DATE: PAGE:

LAB - 2

1) FCFS

```
#include <stdio.h>
void main()
{
    int p[10], wt[10], bt[10], tt[10], tat[10], wt[10], i, j,
    temp=0, n;
    float awt=0, atat=0;
    printf("Enter no. of processes:");
    scanf("%d", &n);
    for(i=0; i<n; i++)
        p[i]=i+1;
    printf("Enter %d arrival time:", n);
    for(i=0; i<n; i++)
        scanf("%d", &wt[i]);
    printf("Enter %d burst time:", n);
    for(i=0; i<n; i++)
        scanf("%d", &bt[i]);
    for(i=0; i<n; i++)
    {
        for(j=i; j<n; j++)
        {
            if(wt[j]<wt[i])
            {
                temp=p[j];
                p[j]=p[i];
                p[i]=temp;
                temp=wt[j];
```

```
            wt[j] = wt[i];
            wt[i] = temp;
            temp = bt[j];
            bt[j] = bt[i];
            bt[i] = temp;
        }
    }
}
ct[0] = wt[0] + bt[0];
for(i=1; i<n; i++)
{
    temp=0;
    if (ct[i-1] < wt[i])
    {
        temp = wt[i] - ct[i-1];
    }
    ct[i] = ct[i-1] + bt[i] + temp;
}
printf("\n P\t AT\t BT\t CT\t TAT\t WT");
for(i=0; i<n; i++)
{
    tat[i] = ct[i] - wt[i];
    wt[i] = tat[i] - bt[i];
    atat += tat[i];
    awt += wt[i];
}
atat = atat/n;
awt = awt/n;
for(i=0; i<n; i++)
{
    printf("\n P%d\t %d\t %d\t %d\t %d\t %d", p[i], wt[i], bt[i], ct[i], tat[i], wt[i]);
}
```

```
    printf("\n Average TAT = %.f", att);
    printf("\n Average WT = %.f", awt);
}
```

OUTPUT:

Enter no. of processes : 4
Enter 4 arrival times : 0 8 3 5
Enter 4 burst times : 7 3 4 6

P	AT	BT	CT	TAT	WT
P1	0	7	7	7	0
P3	3	4	11	8	4
P4	5	6	17	12	6
P2	8	3	20	12	9

Average TAT = 4.750000
Average WT = 4.750000

---

2) SJF (Non-preemptive)

```
#include<stdio.h>
void main()
{
    int time, bt[10], at[10], sum_bt=0, smallest, n, i;
    int sumt=0, sumw=0;
    printf("Enter no. of processes:");
    scanf("%d", &n);
    for(i=0; i<n; i++)
    {
        printf("Enter arrival time for P%d:", i+1);
        scanf("%d", &at[i]);
```

DATE: PAGE:

```
printf("Enter burst time for P%d :", i+1);
scanf("%d", &bt[i]);
}
bt[9] = 9999;
printf("\nP \t AT \t BT \t CT \t TAT \t WT");
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;
        }
    }
printf("\nP%d \t %d \t %d \t %d \t %d \t %d", smallest+1, at[smallest], bt[smallest], time+bt[smallest],
time + bt[smallest] - at[smallest], time - at[smallest]);
sumt += time + bt[smallest] - at[smallest];
sumw += time - at[smallest];
time += bt[smallest];
bt[smallest] = 0;
}
printf("\n Average TAT = %f", sumt/n);
printf("\n Average WT = %f", sumw/n);
}
```

OUTPUT:

```
Enter no. of processes: 4
Enter arrival time for P1: 0
Enter burst time for P1: 7
```

```
Enter arrival time for P2: 8
Enter burst time for P2: 3
Enter arrival time for P3: 3
Enter burst time for P3: 4
Enter arrival time for P4: 5
Enter burst time for P4: 6
```

P	AT	BT	CT	TAT	WT
P1	0	7	7	7	0
P3	3	4	11	8	4
P2	8	3	14	6	3
P4	5	6	20	15	9

```
Average TAT: 9.000000
Average WT: 4.000000
```

3) SJF (Preemptive)

```
#include<stdio.h>
Void main()
{
    int a[10], b[10], x[10];
    int wt[10], tat[10], ct[10];
    int i, j, smallest, count=0, time, n;
    float sum=0, sumwt=0, end;
    printf("Enter the no. of processes:");
    scanf("%d", &n);
    for(i=0; i<n; i++)
    {
        printf("\nEnter arrival time of P%d", i+1);
        scanf("%d", &a[i]);
    }
```

```
for (i=0; i<n; i++)
{
    printf("\nEnter burst time of P%d", i+1);
    scanf("%d", &bt[i]);
}
for (i=0; i<n; i++)
    x[i] = bt[i];
bt[9] = 9999;
for (time=0; count!=n; time++)
{
    smallest = 9;
    for (i=0; i<n; i++)
    {
        if (at[i]<=time && bt[i]<bt[smallest] && bt[i]>0)
        {
            smallest = i;
        }
        bt[smallest]--;
        if (bt[smallest]==0)
        {
            count++;
            end = time+1;
            ct[smallest] = end;
            tat[smallest] = end - at[smallest];
            wt[smallest] = tat[smallest] - x[smallest];
        }
    }
printf("\nP\tAT\tBT\tCT\tTAT\tWT");
for (i=0; i<n; i++)
{
    printf("P%d\t%d\t%d\t%d\t%d\t%d", i+1, at[i], x[i], ct[i], tat[i], wt[i]);
```

```
        sumt = sumt + tat[i];
        sumw = sumw + wt[i];
    }
    printf("Average TAT = %f", sumt/n);
    printf("Average WT = %f", sumw/n);
}
```

OUTPUT:

```
Enter the no. of Processes: 4
Enter arrival time of P1
Enter arrival time of P2:
Enter arrival time of P3:
Enter arrival time of P4:
Enter burst time of P1:
Enter burst time of P2:
Enter burst time of P3:
Enter burst time of P4:
```

P	AT	BT	CT	TAT	WT
P1	0	7	9	9	2
P2	8	3	12	4	1
P3	3	2	5	2	0
P4	5	6	18	13	7

```
Average TAT = 7.000000
Average WT : 2.500000
```