

Task1:

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int n=5, total = 0, i, j, small, temp, procs[100], k, waiting[10], finish[10];
```

```
    float tavg = 0.0, wavg = 0.0;
```

```
    int ari[5]={0,2,3,4,5};
```

```
    int bur[5]={5,2,7,4,5};
```

```
    for (i = 0; i < n; i++)
```

```
    {
```

```
        waiting[i] = 0;
```

```
        total += bur[i];
```

```
    }
```

```
    for (i = 0; i < n; i++)
```

```
    {
```

```
        for (j = i + 1; j < n; j++)
```

```
        {
```

```
            if (ari[i] > ari[j])
```

```
            {
```

```
                temp = ari[i];
```

```
                ari[i] = ari[j];
```

```
                ari[j] = temp;
```

```
                temp = bur[i];
```

```
                bur[i] = bur[j];
```

```
                bur[j] = temp;
```

```
            }
```

```
        }
```

```
    }
```

```

for (i = 0; i < total; i++)
{
    small = 3200;
    for (j = 0; j < n; j++)
    {
        if ((bur[j] != 0) && (ari[j] <= i) && (bur[j] < small))
        {
            small = bur[j];
            k = j;
        }
    }
    bur[k]--;
    procs[i] = k;
}
k = 0;
for (i = 0; i < total; i++)
{
    for (j = 0; j < n; j++)
    {
        if (procs[i] == j)
        {
            finish[j] = i;
            waiting[j]++;
        }
    }
}

printf("Process  Finish time  Turnaround time  Waiting time\n");
for (i = 0; i < n; i++)
{
    printf("%d\t%d\t%d\t%d\n", i + 1, finish[i] + 1, (finish[i] - ari[i]) + 1, (((finish[i] + 1) -
waiting[i]) - ari[i]));
}

```

```

    wavg = wavg + (((finish[i] + 1) - waiting[i]) - ari[i]);

    tavg = tavg + ((finish[i] - ari[i]) + 1);

}

printf("\n Average Wating Time: %f\n Average Turnaround Time%f\n", (wavg / n), (tavg / n));

return 0;

}

```

```

PS C:\Users\skrbm\Downloads\Docs\CSE321 Lab\Lab 4\Codes> cd "c:\Users\skrbm\Downloads\Docs\CSE321 Lab\Lab 4\Codes\" ; if ($?) { gcc task1.c -o task1 } ; if ($?) { .\task1 }
Process  Finish time  Turnaround time  Waiting time
1          7           7           2
2          4           2           0
3         23          20          13
4         11           7           3
5         16          11           6

Average Wating Time: 4.800000
Average Turnaround Time: 9.400000

```

Task 2:

```
#include <stdio.h>
```

```

int main() {

    int WT[4], TAT[4], FT[4];

    int process[4] = {1, 2, 3, 4};

    int burst[4]= {53, 17, 68, 24};

    int temp[4];

    int q = 20, time = 0, flag = 0, n=4;

    int i;

    for (i = 0; i < n; i++)

    {

        temp[i] = burst[i];

    }

    for(;;)

```

```

{

    if (flag == 1) {
        break;
    }

    flag = 1;
    int i;
    for (i = 0; i < n; i++)
    {
        if (temp[i] != 0) {

            if (temp[i] > q)
            {
                time += q;
                temp[i] -= q;
            }
            else
            {
                time += temp[i];
                WT[i] = time - burst[i];
                temp[i] = 0;
            }
            flag = 0;
        }
    }
}

printf("Process\t Finishing Time\t Waiting Time\t Turnaround Time\n");
for (int i = 0; i < n; i++) {
    TAT[i] = burst[i] + WT[i];
}

```

```

        FT[i] = TAT[i] - 0;

        printf("%d\t\t%d\t\t%d\t\t%d\n", i+1, FT[i], WT[i], TAT[i]);

    }

}

```

```

PS C:\Users\skrbm\Downloads\Docs\CSE321 Lab\Lab 4\Codes> cd "c:\Users\skrbm\Downloads\Docs\CSE321 Lab\Lab 4"
Process Finishing Time Waiting Time Turnaround Time
1 134 81 134
2 37 20 37
3 162 94 162
4 121 97 121
PS C:\Users\skrbm\Downloads\Docs\CSE321 Lab\Lab 4\Codes> 

```

Task 3:

```
#include<stdio.h>
```

```
struct process
```

```
{
    int WT,AT,BT,TAT,PT;
};
```

```
struct process a[10];
```

```
int main()
```

```
{
    int n=5,temp[10],t,count=0,short_p;
    float total_WT=0,total_TAT=0,Avg_WT,Avg_TAT;
```

```
    printf("Enter the arrival time , burst time and priority of the process\n");
    printf("AT BT P\n");
```

```
    for(int i=0;i<n;i++)
```

```
    {
        scanf("%d%d%d",&a[i].AT,&a[i].BT,&a[i].PT);
```

```
temp[i]=a[i].BT;
```

```
}
```

```
a[9].PT=10000;
```

```
for(t=0;count!=n;t++)
```

```
{
```

```
    short_p=9;
```

```
    for(int i=0;i<n;i++)
```

```
    {
```

```
        if(a[short_p].PT>a[i].PT && a[i].AT<=t && a[i].BT>0)
```

```
        {
```

```
            short_p=i;
```

```
        }
```

```
    }
```

```
a[short_p].BT=a[short_p].BT-1;
```

```
if(a[short_p].BT==0)
```

```
{
```

```
    count++;
```

```
    a[short_p].WT=t+1-a[short_p].AT-temp[short_p];
```

```
    a[short_p].TAT=t+1-a[short_p].AT;
```

```
    total_WT=total_WT+a[short_p].WT;
```

```
    total_TAT=total_TAT+a[short_p].TAT;
```

```
}
```

```
}
```

```
Avg_WT=total_WT/n;
```

```

Avg_TAT=total_TAT/n;

printf("Process Finishing time Waiting time Turnaround time\n");
for(int i=0;i<n;i++)
{
    printf("%d\t%d\t%d\t%d\n",i+1,a[i].AT+a[i].TAT,a[i].WT,a[i].TAT);
}

printf("Avg waiting time of the process is %f\n",Avg_WT);
printf("Avg turn around time of the process is %f\n",Avg_TAT);

return 0;
}

```

```

Enter the arrival time , burst time and priority of the process
AT BT P
0 15 2
14 5 4
3 10 0
9 22 3
7 16 1
Process Finishing time Waiting time Turnaround time
1 41 26 41
2 68 49 54
3 13 0 10
4 63 32 54
5 29 6 22
Avg waiting time of the process is 22.600000
Avg turn around time of the process is 36.200001

```