Task-1:

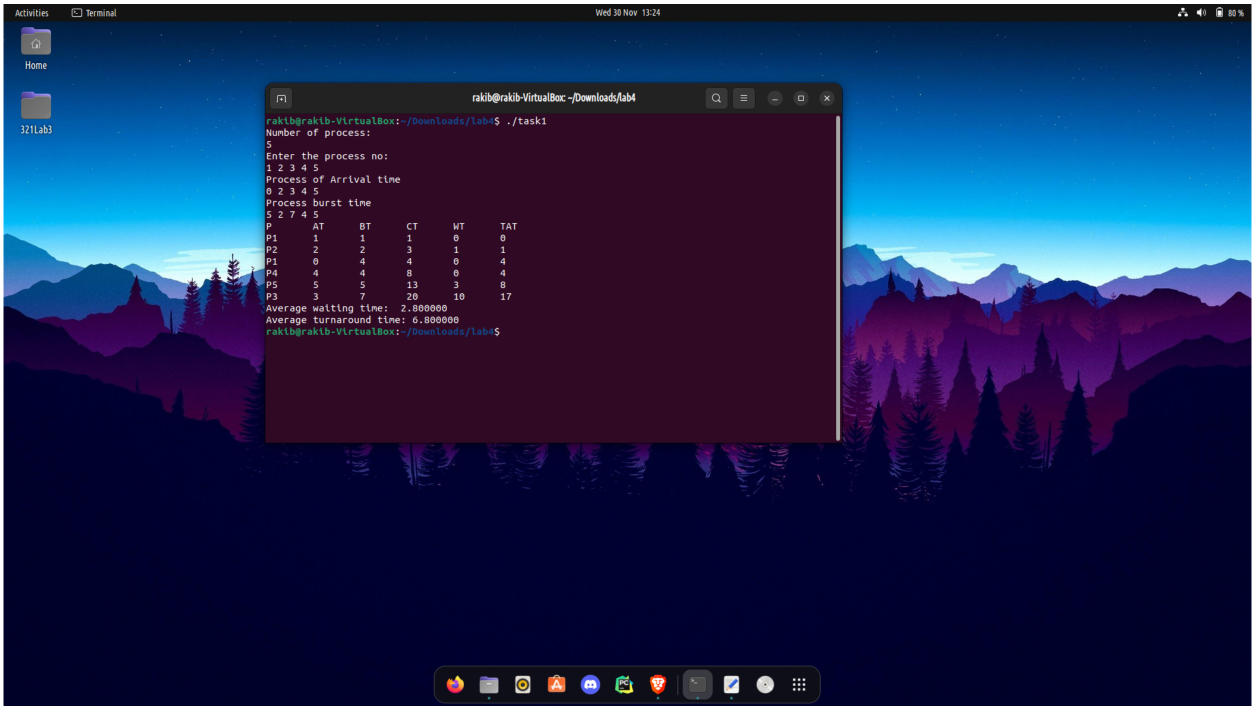
```
#include<stdio.h>
    #include<stdlib.h>
    int main()
    {
        int pn,pno[20],at[20],bt[20],i,tat[20],com[20],wt[20],temp,j,bttemp[20]={},temp1=0,temp2;
        float avg,avg2,avgtat,avgwt;
        printf("Number of process:\n");
        scanf("%d",&pn);
        printf("Enter the process no:\n");
        for(i=0;i<pn;i++)
            scanf("%d",&pno[i]);
        printf("Process Arrival time:\n");
        for(i=0;i<pn;i++)
            scanf("%d",&at[i]);
        printf("Process burst time:\n");
        for(i=0;i<pn;i++)
            scanf("%d",&bt[i]);

        for(i=0;i<pn;i++)
        {
            for(j=i+1;j<pn;j++)
            {
                if(at[i]>at[j])
                {
                    temp=at[i];
                    at[i]=at[j];
                    at[j]=temp;

                    temp=pno[i];
                    pno[i]=pno[j];
                    pno[j]=temp;

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

                }
            }
        }

        temp2=1;
        temp1=bttemp[0]=bt[0]-1;
    com[0]=at[0]+1;

    for(i=1;i<pn;i++)
    {
        bttemp[i]=bt[i];
    }


if(bttemp[0]>bttemp[1])
{
    bttemp[pn]=temp2;

    for(i=0;i<=pn;i++)
    {
        for(j=i+1;j<=pn;j++)
        {
            if(bttemp[i]>bttemp[j])
            {
                temp=at[i];
                at[i]=at[j];
                at[j]=temp;

                temp=pno[i];
                pno[i]=pno[j];
                pno[j]=temp;

                temp=bttemp[i];
                bttemp[i]=bttemp[j];
                bttemp[j]=temp;

            }
        }
    }


    pno[0]=bttemp[0];
    at[0]=bttemp[0];


    if(at[1]<at[2])
    {
        com[1]=com[0]+bttemp[1];
```

```
for(i=0;i<=pn;i++)
{
        for(j=i+1;j<=pn;j++)
        {
                if(at[i]>at[j])
                {
                        temp=at[i];
                        at[i]=at[j];
                        at[j]=temp;

                        temp=pno[i];
                        pno[i]=pno[j];
                        pno[j]=temp;

                        temp=bttemp[i];
                        bttemp[i]=bttemp[j];
                        bttemp[j]=temp;

                }
        }
}

com[2]=com[1]+bttemp[1];

for(i=0;i<=pn;i++)
{
        for(j=i+1;j<=pn;j++)
        {
                if(bttemp[i]>bttemp[j])
                {
                        temp=at[i];
                        at[i]=at[j];
                        at[j]=temp;

                        temp=pno[i];
                        pno[i]=pno[j];
                        pno[j]=temp;

                        temp=bttemp[i];
                        bttemp[i]=bttemp[j];
                        bttemp[j]=temp;
```

```
            }
        }
    }

tat[0]=0;
tat[1]=bttemp[0];


wt[0]=0;
for(i=3;i<=pn;i++)
    com[i]=abs(com[i-1]+bttemp[i]);

for(i=1;i<=pn;i++)
    tat[i]=abs(com[i]-at[i]);


for(i=1;i<=pn;i++)
{
    wt[i]=abs(tat[i]-bttemp[i]);
    if(wt[i]==temp1)
        wt[i]=wt[i]-1;

}
}
else
{
    for(i=0;i<=pn;i++)
    {
        for(j=i+1;j<=pn;j++)
        {
            if(at[i]>at[j])
            {
                temp=at[i];
                at[i]=at[j];
                at[j]=temp;

                temp=pno[i];
                pno[i]=pno[j];
                pno[j]=temp;

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

com[1]=com[0]+bttemp[2];

for(i=0;i<=pn;i++)
{
        for(j=i+1;j<=pn;j++)
        {
                if(bttemp[i]>bttemp[j])
                {
                        temp=at[i];
                        at[i]=at[j];
                        at[j]=temp;

                        temp=pno[i];
                        pno[i]=pno[j];
                        pno[j]=temp;

                        temp=bttemp[i];
                        bttemp[i]=bttemp[j];
                        bttemp[j]=temp;

                }
        }
}


com[2]=com[1]+bttemp[1];


if(at[1]>at[2])
{
        temp=at[1];
        at[1]=at[2];
        at[2]=temp;

        temp=pno[1];
        pno[1]=pno[2];
        pno[2]=temp;

        temp=bttemp[1];
        bttemp[1]=bttemp[2];
```

```
                bttemp[2]=temp;
        }


tat[0]=0;
tat[1]=bttemp[0];


wt[0]=0;
for(i=3;i<=pn;i++)
        com[i]=abs(com[i-1]+bttemp[i]);

for(i=1;i<=pn;i++)
        tat[i]=abs(com[i]-at[i]);


for(i=1;i<=pn;i++)
{
        if(temp1==bttemp[i])
                bttemp[i]+=1;
        wt[i]=abs(tat[i]-bttemp[i]);
        if(wt[i]==temp1)
                wt[i]=wt[i]-1;

}

}



for(i=0;i<=pn;i++)
        avg=avg+tat[i];

avgtat=avg/pn;

for(i=0;i<=pn;i++)
        avg2=avg2+wt[i];

avgwt=avg2/pn;


printf("P\tAT\tBT\tCT\tWT\tTAT\t\n");
for(i=0;i<=pn;i++)
{
```

```
printf("P%d\t%d\t%d\t%d\t%d\t%d\n",pno[i],at[i],bttemp[i],com[i],wt[i],tat[i]);
                }

                printf("Average waiting time:  %f\nAverage turnaround time:
%f\n",avgwt,avgtat);
            }
            else
            {

                bttemp[0]=bttemp[0]+temp2;
                com[0]=bttemp[0];
                tat[0]=bttemp[0];
                wt[0]=0;

                for(i=1;i<pn;i++)
                {
                    for(j=i+1;j<pn;j++)
                    {
                        if(bttemp[i]>bttemp[j])
                        {
                            temp=at[i];
                            at[i]=at[j];
                            at[j]=temp;

                            temp=pno[i];
                            pno[i]=pno[j];
                            pno[j]=temp;

                            temp=bttemp[i];
                            bttemp[i]=bttemp[j];
                            bttemp[j]=temp;

                        }
                    }
                }


                for(i=1;i<pn;i++)
                    com[i]=abs(com[i-1]+bttemp[i]);

                for(i=0;i<pn;i++)
                    tat[i]=abs(com[i]-at[i]);
                for(i=0;i<pn;i++)
                    wt[i]=abs(tat[i]-bttemp[i]);

                for(i=0;i<pn;i++)
                    avg=avg+tat[i];

                avgtat=avg/pn;

                for(i=0;i<pn;i++)
                    avg2=avg2+wt[i];

                avgwt=avg2/pn;


                printf("P\tAT\tBT\tCT\tWT\tTAT\t\n");
                for(i=0;i<pn;i++)
                {

printf("P%d\t%d\t%d\t%d\t%d\t%d\n",pno[i],at[i],bttemp[i],com[i],wt[i],tat[i]);
                }

                printf("Average waiting time:  %f\nAverage turnaround time:
%f\n",avgwt,avgtat);
            }



        }
```



```
rakib@rakib-VirtualBox:~/Downloads/lab4$ ./task1
Number of process:
5
Enter the process no:
1 2 3 4 5
Process of Arrival time
0 2 3 4 5
Process burst time
5 2 7 4 5
P       AT      BT      CT      WT      TAT
P1      1       1       1       0       0
P2      2       2       3       1       1
P1      0       4       4       0       4
P4      4       4       8       0       4
P5      5       5       13      3       8
P3      3       7       20      10      17
Average waiting time:  2.800000
Average turnaround time: 6.800000
rakib@rakib-VirtualBox:~/Downloads/lab4$
```

Task-2:

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
        int pn,pno[50]={0},at[50],bt[50],i,tat[50],com[50],wt[50],temp,j,qm,bttemp[50],btttemp[50]={0},tattemp[50]={0},tatat[50]={},tatttemp[50]={},wtbt[50],wttbt[50]={};
        float avg,avg2,avgtat,avgwt;
        printf("Number of process:\n");
        scanf("%d",&pn);
        printf("Enter the process:\n");
        for(i=0;i<pn;i++)
                scanf("%d",&pno[i]);
        printf("Process Arrival time\n");
        for(i=0;i<pn;i++)
                scanf("%d",&at[i]);
        printf("Process burst time\n");
        for(i=0;i<pn;i++)
                scanf("%d",&bt[i]);
        printf("Time quantum:\n");
                scanf("%d",&qm);

        for(i=0;i<pn;i++)
        {
```

```
	for(j=i+1;j<pn;j++)
	{
		if(at[i]>at[j])
		{
			temp=at[i];
			at[i]=at[j];
			at[j]=temp;

			temp=pno[i];
			pno[i]=pno[j];
			pno[j]=temp;

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

		}
	}
}


for(i=0;i<pn;i++)
{
	bttemp[i]=bt[i]-qm;
	if(bttemp[i]<0)
	{
		bttemp[i]=bt[i];
		btttemp[i]=bttemp[i];
		bttemp[i]=0;
	}
	if(bt[i]<=qm)
		bttemp[i]=bt[i]-bt[i];

}


for(i=0;i<pn;i++)
{
	if(bttemp[i]==bttemp[i+1])
	{
		com[0]=bt[0]+at[0];
		tat[0]=bt[0];
		wt[0]=0;
		for(i=1;i<pn;i++)
```

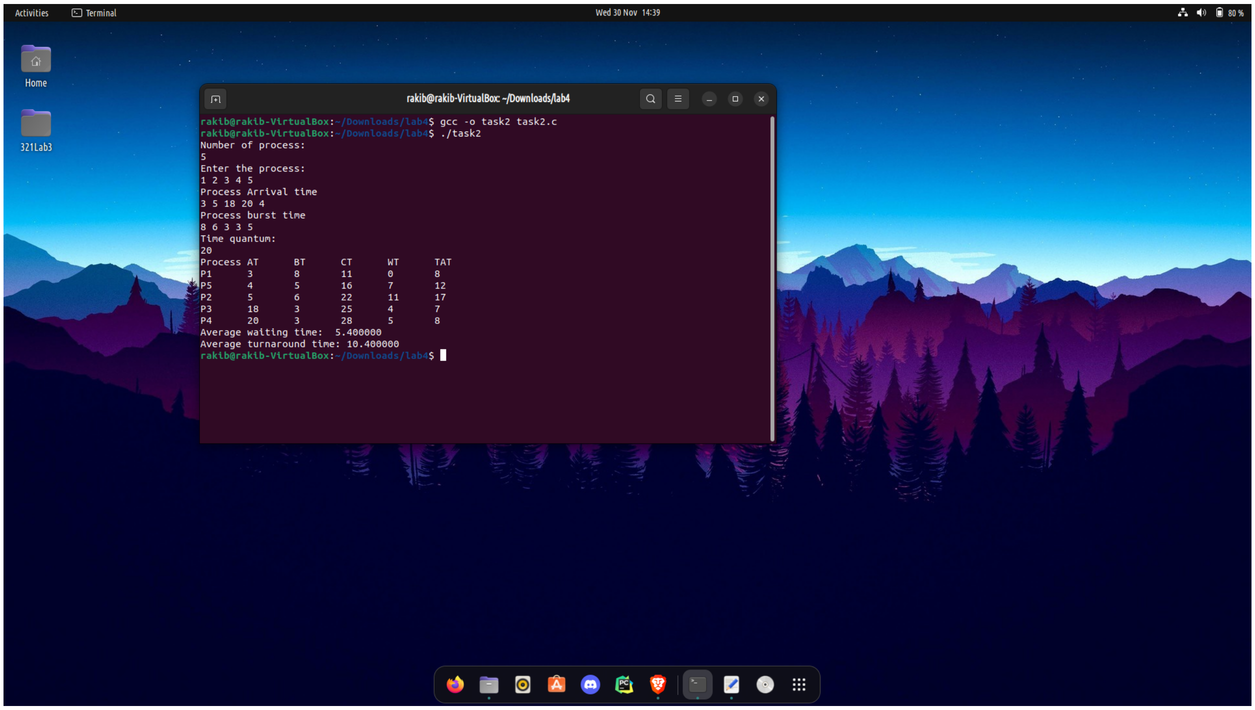
```
                        com[i]=abs(com[i-1]+bt[i]);
                    for(i=0;i<pn;i++)
                        tat[i]=abs(com[i]-at[i]);


                    for(i=0;i<pn;i++)
                        wt[i]=abs(tat[i]-bt[i]);

                    for(i=0;i<pn;i++)
                        avg=avg+tat[i];

                    avgtat=avg/pn;

                    for(i=0;i<pn;i++)
                        avg2=avg2+wt[i];

                    avgwt=avg2/pn;


                    printf("Process\tAT\tBT\tCT\tWT\tTAT\t\n");
                    for(i=0;i<pn;i++)
                    {

printf("P%d\t%d\t%d\t%d\t%d\t%d\n",pno[i],at[i],bt[i],com[i],wt[i],tat[i]);
                    }

                    printf("Average waiting time:  %f\nAverage turnaround
time: %f\n",avgwt,avgtat);
                    }
                else if(bttemp[i]>bttemp[i+1]||bttemp[i]<bttemp[i+1])
                {
                    if(bttemp[0]>=qm)
                        com[0]=at[0]+qm;
                    else
                        com[0]=at[0]+bt[0];

                    for(i=1;i<pn;i++)
                    {
                        if(bt[i]>=qm)
                            com[i]=com[i-1]+qm;
                        else
                            com[i]=com[i-1]+bt[i];

                    }
```

```
for(i=0;i<pn;i++)
{

    if(bttemp[i]>0)
    {
        pno[i+pn]=pno[i];
        at[i+pn]=at[i];
        bt[i+pn]=bt[i];
        bttemp[i+pn]=bttemp[i]-qm;
        if(bttemp[i+pn]<0)
        {
            bttemp[i+pn]=bttemp[i];
            btttemp[i]=bttemp[i+pn];
            bttemp[i+pn]=0;
        }
        if(bttemp[i]<=qm)
            bttemp[i+pn]=bttemp[i]-bttemp[i];

    }
    else
    {
        pno[i+pn]=0;
        at[i+pn]=0;
        bt[i+pn]=0;
        bttemp[i+pn]=0;
    }
}


for(i=0;i<pn;i++)
{
    if(bttemp[i+pn]!=-1)
    {
        if(bttemp[i+pn]>=qm)
            com[i+pn]=com[i+pn-1]+qm;
        else
            com[i+pn]=com[i+pn-1]+btttemp[i];
    }
    else
    {
        com[i+pn]=0;
    }
}
for(i=pn;i<pn+pn-1;i++)
{

          if(bttemp[i]>0)
          {
                    pno[i+pn-1]=pno[i];
                    at[i+pn-1]=at[i];
                    bt[i+pn-1]=bt[i];
                    bttemp[i+pn-1]=bttemp[i]-qm;
                    if(bttemp[i+pn-1]<0)
                    {
                              bttemp[i+pn-1]=bttemp[i];
                              btttemp[i]=bttemp[i+pn-1];
                              bttemp[i+pn-1]=0;
                    }
                    if(bttemp[i]<=qm)
                              bttemp[i+pn-1]=bttemp[i]-bttemp[i];
          }
          else
          {
                    pno[i+pn-1]=0;
                    at[i+pn-1]=0;
                    bt[i+pn-1]=0;
                    bttemp[i+pn-1]=0;
          }

}

for(i=pn;i<pn+pn-1;i++)
{
          if(bttemp[i+pn-1]!=-1)
          {
                    if(bttemp[0+pn]>=qm)
                              com[i+pn-1]=com[i+pn-1]+qm;
                    else
                              com[i+pn-1]=com[i+pn-1]+btttemp[i];
          }
          else
          {
                    com[i+pn-1]=0;
          }
}
```

```
                    for(i=pn+pn-1;i<pn+pn+1;i++)
                    {
                        if(bttemp[i]>0)
                        {
                                pno[i+1]=pno[i];
                                at[i+1]=at[i];
                                bt[i+1]=bt[i];
                                bttemp[i+1]=bttemp[i]-qm;
                                if(bttemp[i+1]<0)
                                {
                                    bttemp[i+1]=bttemp[i];
                                    btttemp[i]=bttemp[i+1];
                                    bttemp[i+1]=0;
                                }
                                if(bttemp[i]<=qm)

bttemp[i+1]=bttemp[i]-bttemp[i];
                        }
                        else
                        {
                            pno[i+1]=0;
                            at[i+1]=0;
                            bt[i+1]=0;
                            bttemp[i+1]=0;
                        }

                    }

                    for(i=pn+pn-1;i<pn+pn+1;i++)
                    {
                        if(bttemp[i+1]!=-1)
                        {
                            if(bttemp[i]>=qm)
                                com[i+1]=com[i]+qm;
                            else
                                com[i+1]=com[i]+btttemp[i];
                        }
                        else
                        {
                            com[i+1]=0;
                        }
                    }

                    for(i=0;i<pn+pn+1;i++)
```

```
{
    tattemp[i]=com[i];
    tatat[i]=at[i];
    wtbt[i]=bt[i];
}


for(i=0;i<pn+pn+1;i++)
{
    for(j=i+1;j<pn+pn+1;j++)
    {
        if(tatat[i]>tatat[j])
        {
            temp=tatat[i];
            tatat[i]=tatat[j];
            tatat[j]=temp;

            temp=tattemp[i];
            tattemp[i]=tattemp[j];
            tattemp[j]=temp;


            temp=wtbt[i];
            wtbt[i]=wtbt[j];
            wtbt[j]=temp;

        }
    }
}

for(i=0;i<pn+pn+1;i++)
{
        if(tatat[i]!=tatat[i+1])
        {
                tatttemp[i]=tattemp[i];
                wttbt[i]=wtbt[i];

        }
}

for(i=0;i<pn+pn+1;i++)
{
        tat[i]=tatttemp[i]-tatat[i];
        if(tat[i]<0)
```

```
                                tat[i]=0;
                            wt[i]=tat[i]-wttbt[i];
                            if(wt[i]<0)
                                wt[i]=0;
                    }

                    for(i=0;i<pn+pn+1;i++)
                        avg=avg+tat[i];
                    avgtat=avg/pn;

                    for(i=0;i<pn+pn+1;i++)
                        avg2=avg2+wt[i];
                    avgwt=avg2/pn;


                    printf("Process\tAT\tBT\tCT\tWT\tTAT\t\n");
                    for(i=0;i<pn+pn+1;i++)
                    {
printf("P%d\t%d\t%d\t%d\t%d\t%d\n",pno[i],at[i],bt[i],com[i],wt[i],tat[i]);
                    }

                    printf("Average waiting time:  %f\nAverage turnaround
time: %f\n",avgwt,avgtat);
                }
            }

        }
```



```
rakib@rakib-VirtualBox:~/Downloads/lab4$ gcc -o task2 task2.c
rakib@rakib-VirtualBox:~/Downloads/lab4$ ./task2
Number of process:
5
Enter the process:
1 2 3 4 5
Process Arrival time
3 5 18 20 4
Process burst time
8 6 3 3 5
Time quantum:
20
Process AT      BT      CT      WT      TAT
P1      3       8       11      0       8
P5      4       5       16      7       12
P2      5       6       22      11      17
P3      18      3       25      4       7
P4      20      3       28      5       8
Average waiting time:  5.400000
Average turnaround time: 10.400000
rakib@rakib-VirtualBox:~/Downloads/lab4$ 
```

Task-3:

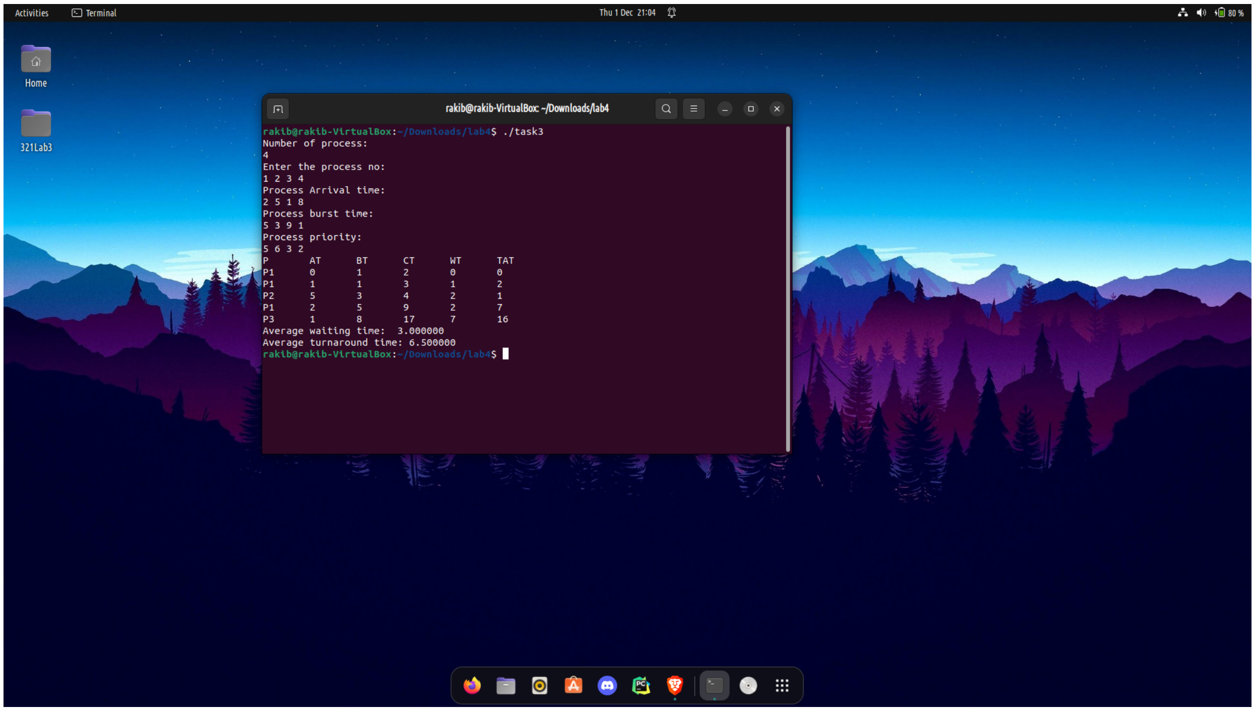
```
#include<stdio.h>
#include<stdlib.h>
int main()
{
        int pn,pno[20],at[20],bt[20],pr[20],i,tat[20],com[20],wt[20],temp,j,bttemp[20]={},temp1=0,temp2;
        float avg,avg2,avgtat,avgwt;
        printf("Number of process:\n");
        scanf("%d",&pn);
        printf("Enter the process no:\n");
        for(i=0;i<pn;i++)
                scanf("%d",&pno[i]);
        printf("Process Arrival time:\n");
        for(i=0;i<pn;i++)
                scanf("%d",&at[i]);
        printf("Process burst time:\n");
        for(i=0;i<pn;i++)
                scanf("%d",&bt[i]);
        printf("Process priority:\n");
        for(i=0;i<pn;i++)
                scanf("%d",&pr[i]);

        for(i=0;i<pn;i++)
        {
```

```
    for(j=i+1;j<pn;j++)
    {
        if(at[i]>at[j])
        {
            temp=at[i];
            at[i]=at[j];
            at[j]=temp;

            temp=pno[i];
            pno[i]=pno[j];
            pno[j]=temp;

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

        }
    }
}

temp2=1;
temp1=bttemp[0]=bt[0]-1;

    com[0]=at[0]+1;

    for(i=1;i<pn;i++)
    {
        bttemp[i]=bt[i];
    }


if(bttemp[0]>bttemp[1])
{
    bttemp[pn]=temp2;

    for(i=0;i<=pn;i++)
    {
        for(j=i+1;j<=pn;j++)
        {
            if(bttemp[i]>bttemp[j])
            {
                temp=at[i];
                at[i]=at[j];
                at[j]=temp;
            temp=pno[i];
            pno[i]=pno[j];
            pno[j]=temp;

            temp=bttemp[i];
            bttemp[i]=bttemp[j];
            bttemp[j]=temp;

        }
    }
}


pno[0]=bttemp[0];
at[0]=bttemp[0];


if(at[1]<at[2])
{
    com[1]=com[0]+bttemp[1];



    for(i=0;i<=pn;i++)
    {
        for(j=i+1;j<=pn;j++)
        {
            if(at[i]>at[j])
            {
                temp=at[i];
                at[i]=at[j];
                at[j]=temp;

                temp=pno[i];
                pno[i]=pno[j];
                pno[j]=temp;

                temp=bttemp[i];
                bttemp[i]=bttemp[j];
                bttemp[j]=temp;

            }
        }
    }

    com[2]=com[1]+bttemp[1];

    for(i=0;i<=pn;i++)
    {
        for(j=i+1;j<=pn;j++)
        {
            if(bttemp[i]>bttemp[j])
            {
                temp=at[i];
                at[i]=at[j];
                at[j]=temp;

                temp=pno[i];
                pno[i]=pno[j];
                pno[j]=temp;

                temp=bttemp[i];
                bttemp[i]=bttemp[j];
                bttemp[j]=temp;

            }
        }
    }

tat[0]=0;
tat[1]=bttemp[0];


wt[0]=0;
for(i=3;i<=pn;i++)
    com[i]=abs(com[i-1]+bttemp[i]);

for(i=1;i<=pn;i++)
    tat[i]=abs(com[i]-at[i]);


for(i=1;i<=pn;i++)
{
    wt[i]=abs(tat[i]-bttemp[i]);
    if(wt[i]==temp1)
        wt[i]=wt[i]-1;
```

```
}
}
else
{
    for(i=0;i<=pn;i++)
    {
        for(j=i+1;j<=pn;j++)
        {
            if(at[i]>at[j])
            {
                temp=at[i];
                at[i]=at[j];
                at[j]=temp;

                temp=pno[i];
                pno[i]=pno[j];
                pno[j]=temp;

                temp=bttemp[i];
                bttemp[i]=bttemp[j];
                bttemp[j]=temp;

            }
        }
    }

    com[1]=com[0]+bttemp[2];

    for(i=0;i<=pn;i++)
    {
        for(j=i+1;j<=pn;j++)
        {
            if(bttemp[i]>bttemp[j])
            {
                temp=at[i];
                at[i]=at[j];
                at[j]=temp;

                temp=pno[i];
                pno[i]=pno[j];
                pno[j]=temp;

                temp=bttemp[i];
                bttemp[i]=bttemp[j];
                bttemp[j]=temp;

            }
        }
    }


    com[2]=com[1]+bttemp[1];


    if(at[1]>at[2])
    {
        temp=at[1];
        at[1]=at[2];
        at[2]=temp;

        temp=pno[1];
        pno[1]=pno[2];
        pno[2]=temp;

        temp=bttemp[1];
        bttemp[1]=bttemp[2];
        bttemp[2]=temp;
    }


tat[0]=0;
tat[1]=bttemp[0];


wt[0]=0;
for(i=3;i<=pn;i++)
    com[i]=abs(com[i-1]+bttemp[i]);

for(i=1;i<=pn;i++)
    tat[i]=abs(com[i]-at[i]);


for(i=1;i<=pn;i++)
{
    if(temp1==bttemp[i])
        bttemp[i]+=1;
    wt[i]=abs(tat[i]-bttemp[i]);
    if(wt[i]==temp1)
```

```
                        wt[i]=wt[i]-1;

                }

                }



                for(i=0;i<=pn;i++)
                    avg=avg+tat[i];

                avgtat=avg/pn;

                for(i=0;i<=pn;i++)
                    avg2=avg2+wt[i];

                avgwt=avg2/pn;


                printf("P\tAT\tBT\tCT\tWT\tTAT\t\n");
                for(i=0;i<=pn;i++)
                {

printf("P%d\t%d\t%d\t%d\t%d\t%d\n",pno[i],at[i],bttemp[i],com[i],wt[i],tat[i]);
                }

                printf("Average waiting time:  %f\nAverage turnaround time:
%f\n",avgwt,avgtat);
            }
            else
            {

                bttemp[0]=bttemp[0]+temp2;
                com[0]=bttemp[0];
                tat[0]=bttemp[0];
                wt[0]=0;

                for(i=1;i<pn;i++)
                {
                    for(j=i+1;j<pn;j++)
                    {
                        if(bttemp[i]>bttemp[j])
                        {
                            temp=at[i];
                            at[i]=at[j];
                            at[j]=temp;

                            temp=pno[i];
                            pno[i]=pno[j];
                            pno[j]=temp;

                            temp=bttemp[i];
                            bttemp[i]=bttemp[j];
                            bttemp[j]=temp;

                        }
                    }
                }


                for(i=1;i<pn;i++)
                    com[i]=abs(com[i-1]+bttemp[i]);

                for(i=0;i<pn;i++)
                    tat[i]=abs(com[i]-at[i]);


                for(i=0;i<pn;i++)
                    wt[i]=abs(tat[i]-bttemp[i]);

                for(i=0;i<pn;i++)
                    avg=avg+tat[i];

                avgtat=avg/pn;

                for(i=0;i<pn;i++)
                    avg2=avg2+wt[i];

                avgwt=avg2/pn;


                printf("P\tAT\tBT\tCT\tWT\tTAT\t\n");
                for(i=0;i<pn;i++)
                {

printf("P%d\t%d\t%d\t%d\t%d\t%d\n",pno[i],at[i],bttemp[i],com[i],wt[i],tat[i]);
                }
```

```
                                        printf("Average waiting time:  %f\nAverage turnaround time: 
%f\n",avgwt,avgtat);
                              }



                    }
```



```
rakib@rakib-VirtualBox:~/Downloads/lab4$ ./task3
Number of process:
4
Enter the process no:
1 2 3 4
Process Arrival time:
2 5 1 8
Process burst time:
5 3 9 1
Process priority:
5 6 3 2
P       AT      BT      CT      WT      TAT
P1      0       1       2       0       0
P1      1       1       3       1       2
P2      5       3       4       2       1
P1      2       5       9       2       7
P3      1       8       17      7       16
Average waiting time:  3.000000
Average turnaround time: 6.500000
rakib@rakib-VirtualBox:~/Downloads/lab4$ 
```