**Lab Report No: 09**

**Name of the lab report: Implementation of Priority Scheduling Algorithm.**

**Name: Binodon**

**ID: IT-17046**

**Objective:** Priority Scheduling algorithm Definition & executable code in c are followed.

**Q.1 What is priority Scheduling algorithm?**

**Ans:**Priorities can be either dynamic or static. Static priorities are allocated during creation, whereas dynamic priorities are assigned depending on the behavior of the processes while in the system. To illustrate, the scheduler could favor input/output (I/O) intensive tasks, which lets expensive requests to be issued as soon as possible.

**Q.2 How to implemented in C?**

**Ans:**

#include <stdio.h>

int main()

{

intbt[20],wt[20],p[20],tat[20],priority[20];

floatavwt=0,avtat=0;

inti,j,n,temp,key;

printf("\nEnter the number of the processes: ");

scanf("%d",&n);

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

{

printf("\nEnter the burst time and priority of the process P[%d]: ",i);

scanf("%d",&bt[i]);

scanf("%d",&priority[i]);

p[i]=i;

}

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

{

key=i;

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

{

if(priority[j]<priority[key])

{

key=j;

}

}

temp=bt[i];

bt[i]=bt[key];

bt[key]=temp;

temp=priority[i];

priority[i]=priority[key];

priority[key]=temp;

temp=p[i];

p[i]=p[key];

p[key]=temp;

}

wt[0]=0;

tat[0]=bt[0];

avtat=tat[0];

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

{

wt[i]=wt[i-1]+bt[i-1];

tat[i]=tat[i-1]+bt[i];

avwt+=wt[i];

avtat+=tat[i];

}

avwt=avwt/n;

avtat=avtat/n;

printf("\n\nPROCESS\t\twaiting time\tburst time\tTurnaround time\n");

printf("\n");

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

{

printf("P[%d]\t\t%d\t\t%d\t\t%d\n",p[i],wt[i],bt[i],tat[i]);

}

printf("\n\nAverage waiting time: %.2f",avwt);

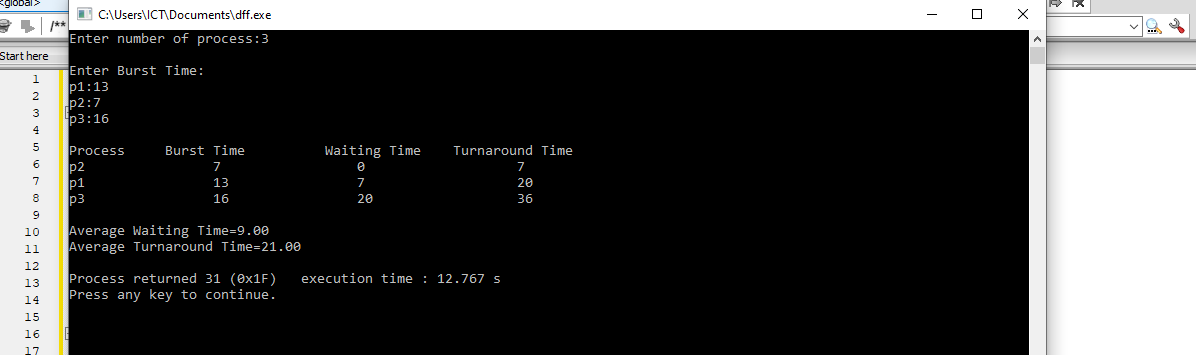
printf("\n\nAverageTurn around time is: %.2f",avtat);

printf("\n");

return 0;

}

**Output:**

****