

MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

Santosh,Tangail-1902

**LAB REPORT**

Lab Report No : 09

Lab Report name : Implementation of Priority Scheduling algorithm.

Course Title :Operating System Lab

Course Code :ICT-3110

Date of Performance :

Date of Submission : 29/9/2020

Submitted by,

Name : Farhana Afrin Shikha

ID :IT-18038

Session : 2017-18

3rd Year 1stsemester

Dept. of ICT

Submitted to,

Nazrul Islam

Assistant Professor

Dept. of ICT,

MBSTU.

**Lab Report 9: Implementation of Priority Scheduling algorithm .**

**Theory :** In priority scheduling algorithm each process has a priority associated with it and as each process hits the queue, it is stored in based on its priority so that process with higher priority are dealt with first. It should be noted that equal priority processes are scheduled in FCFS order.

Turnaround Time = Completion Time - Arrival Time

Waiting Time = Turn Around Time - Burst Time

**Corresponding code:**

#include<stdio.h>

int main()

{

int bt[20], p[20], wt[20], tat[20], pr[20],i,j,n,total=0,pos,temp,avg\_wt,avg\_tat;

printf("Enter Total Number of Process:");

scanf("%d",&n);

printf("\nEnter Burst Time and Priority\n");

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

{

printf("\nP[%d]\n",i+1);

printf("Burst Time:");

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

printf("Priority:");

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

p[i]=i+1; //contains process number

}

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

{

pos=i;

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

{

if(pr[j]<pr[pos])

pos=j;

}

temp=pr[i];

pr[i]=pr[pos];

pr[pos]=temp;

temp=bt[i];

bt[i]=bt[pos];

bt[pos]=temp;

temp=p[i];

p[i]=p[pos];

p[pos]=temp;

}

wt[0]=0; //waiting time for first process is zero

//calculate waiting time

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

{

wt[i]=0;

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

wt[i]+=bt[j];

total+=wt[i];

}

avg\_wt=total/n; //average waiting time

total=0;

printf("\nProcess\t Burst Time \tWaiting Time\tTurnaround Time");

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

{

tat[i]=bt[i]+wt[i]; //calculate turnaround time

total+=tat[i];

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

}

avg\_tat=total/n; //average turnaround time

printf("\n\nAverage Waiting Time=%d",avg\_wt);

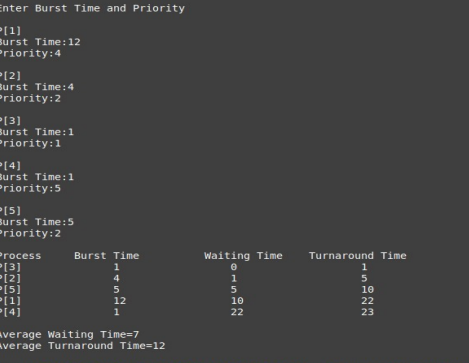
printf("\nAverage Turnaround Time=%d\n",avg\_tat);

printf("\n");

return 0;

}

**Output:**



**Conclusion:** The priority process algorithm implement and works successfully. **Priority Scheduling** is a method of scheduling processes that is based on priority. In this algorithm, the scheduler selects the tasks to work as per the priority. The processes with higher priority should be carried out first, whereas jobs with equal priorities are carried out on FCFS basis. Priority depends upon memory requirements, time requirements, etc.In this algorithm Lower the number, higher is the priority. In this type of scheduling algorithm, if a newer process arrives, that is having a higher priority than the currently running process, then the currently running process is preempted.Here are an example of priority scheduling algorithm.In this example there are five processes:p1,p2,p3,p4 and p5.The priority number of this processes are respectively: 4,2,1,5 and 2.So we can see that the priority number of p3 is the lowest and so the priority of p3 is high than other processes.Thats why p3 process come first in gannt chart.Next come p2,p5,p1,p4 respectively according to priority.