Mawlana Bhashani Science and Technology University

Lab-Report

Report No: 09

Course code: ICT-3110

Course title: Operating System Lab

Date of Performance:

Date of Submission: 09/10/2020

Submitted by

Name: Shakhera khanom

ID: IT-18033

3rd year 1st semester

Session: 2017-2018

Dept. of ICT

MBSTU.

Submitted To

Nazrul Islam

Assistant Professor

Dept. of ICT

MBSTU.

Experiment No : 09

Experiment Name: Implementation of Priority Scheduling Algorithm.

Objectives:

- What is Priority Scheduling Algorithm?
- How to implement in c?

- **Priority Scheduling Algorithm:**

  → Processes with same priority are execute on first come first served (FCFS) basis.

  → Priority scheduling is a non-preemptive algorithm and one of most common scheduling algorithm in batch systems.

  → Each process is assigned a priority process with highest priority is to be execute first and so on.

Aim: To write a c program to implement the CPU scheduling priority algorithm.

Description:

To calculate the average waiting time in the priority algorithm, sort the burst times according to their priorities and then calculate the average waiting time of the processes. The waiting time of each process is obtained by summing up the burst times of all the previous processes.

Algorithm:

Step 1: Start the process

Step 2: Accept the number of processes in the ready Queue

Step 3: For each process in the ready Q, assign the process id and accept the CPU burst time

Step 4: Sort the ready queue according to the priority number.

Step 5: Set the waiting of the first process as _0 and its burst time as its turnaround time

Step 6: Arrange the processes based on process priority

Step 7: for each process in the Ready Q calculate

a) Waiting time(n)= waiting time (n-1) + Burst time (n-1)

b) Turnaround time (n)= waiting time(n)+Burst time(n)

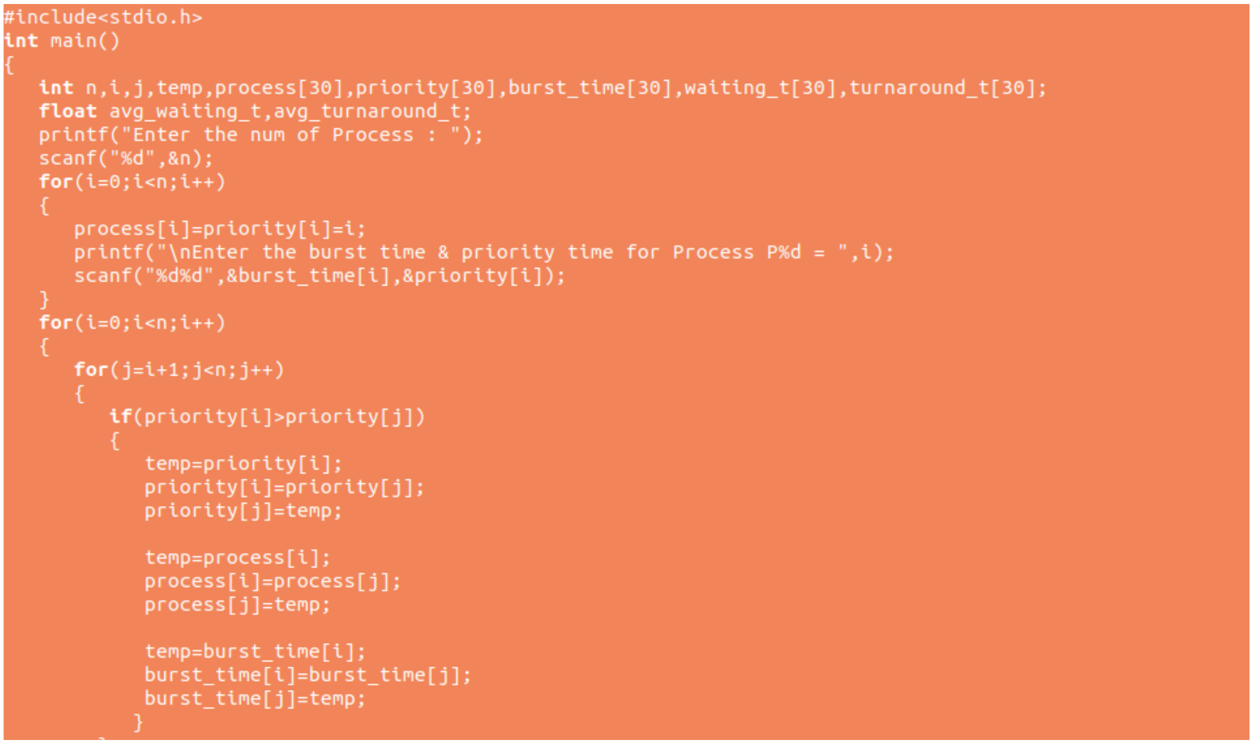
Step 8: Calculate

c) Average waiting time = Total waiting Time / Number of process

d) Average Turnaround time = Total Turnaround Time / Number of process Print the results in an order

Step 9: Stop the process

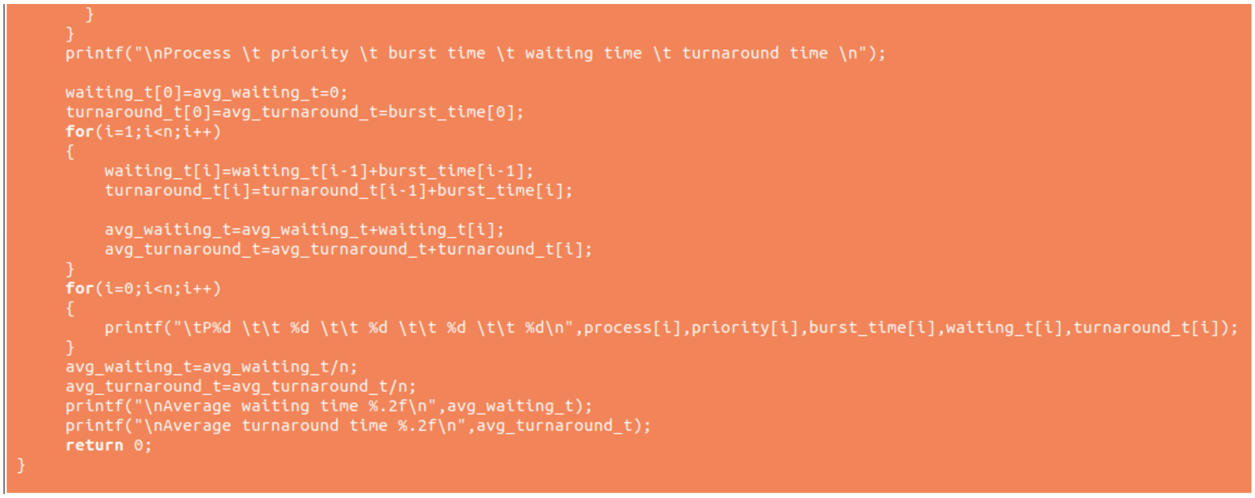
Corresponding Code:



```
#include<stdio.h>
int main()
{
   int n,i,j,temp,process[30],priority[30],burst_time[30],waiting_t[30],turnaround_t[30];
   float avg_waiting_t,avg_turnaround_t;
   printf("Enter the num of Process : ");
   scanf("%d",&n);
   for(i=0;i<n;i++)
   {
      process[i]=priority[i]=i;
      printf("\nEnter the burst time & priority time for Process P%d = ",i);
      scanf("%d%d",&burst_time[i],&priority[i]);
   }
   for(i=0;i<n;i++)
   {
      for(j=i+1;j<n;j++)
      {
         if(priority[i]>priority[j])
         {
            temp=priority[i];
            priority[i]=priority[j];
            priority[j]=temp;

            temp=process[i];
            process[i]=process[j];
            process[j]=temp;

            temp=burst_time[i];
            burst_time[i]=burst_time[j];
            burst_time[j]=temp;
           }
```

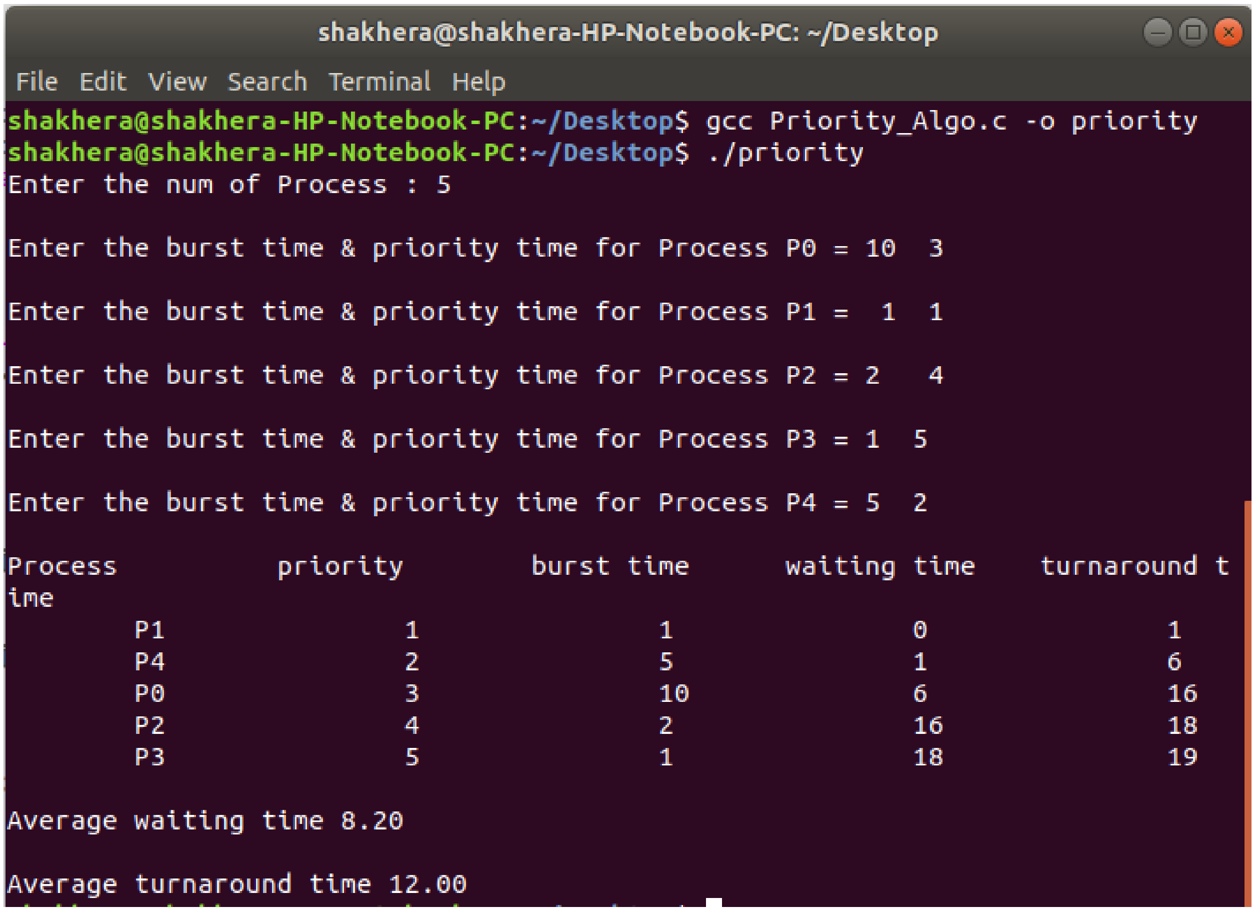


```
        }
      }
      printf("\nProcess \t priority \t burst time \t waiting time \t turnaround time \n");

      waiting_t[0]=avg_waiting_t=0;
      turnaround_t[0]=avg_turnaround_t=burst_time[0];
      for(i=1;i<n;i++)
      {
          waiting_t[i]=waiting_t[i-1]+burst_time[i-1];
          turnaround_t[i]=turnaround_t[i-1]+burst_time[i];

          avg_waiting_t=avg_waiting_t+waiting_t[i];
          avg_turnaround_t=avg_turnaround_t+turnaround_t[i];
      }
      for(i=0;i<n;i++)
      {
          printf("\tP%d \t\t %d \t\t %d \t\t %d \t\t %d\n",process[i],priority[i],burst_time[i],waiting_t[i],turnaround_t[i]);
      }
      avg_waiting_t=avg_waiting_t/n;
      avg_turnaround_t=avg_turnaround_t/n;
      printf("\nAverage waiting time %.2f\n",avg_waiting_t);
      printf("\nAverage turnaround time %.2f\n",avg_turnaround_t);
      return 0;
}
```

Output:



```
shakhera@shakhera-HP-Notebook-PC:~/Desktop$ gcc Priority_Algo.c -o priority
shakhera@shakhera-HP-Notebook-PC:~/Desktop$ ./priority
Enter the num of Process : 5

Enter the burst time & priority time for Process P0 = 10  3

Enter the burst time & priority time for Process P1 =  1  1

Enter the burst time & priority time for Process P2 = 2   4

Enter the burst time & priority time for Process P3 = 1  5

Enter the burst time & priority time for Process P4 = 5  2

Process          priority        burst time      waiting time     turnaround t
ime
        P1               1               1               0               1
        P4               2               5               1               6
        P0               3               10              6               16
        P2               4               2               16              18
        P3               5               1               18              19

Average waiting time 8.20

Average turnaround time 12.00
```

Discussion:

This lab helps to learn priority algorithm. Using this algorithm we can find waiting time and turnaround time that depend the priority.