Operating System – CS23431

Ex 6c)	Priority Scheduling
Name: B M Madhumitha	
Reg No: 230701168	

Aim:

To implement priority scheduling technique

Algorithm:

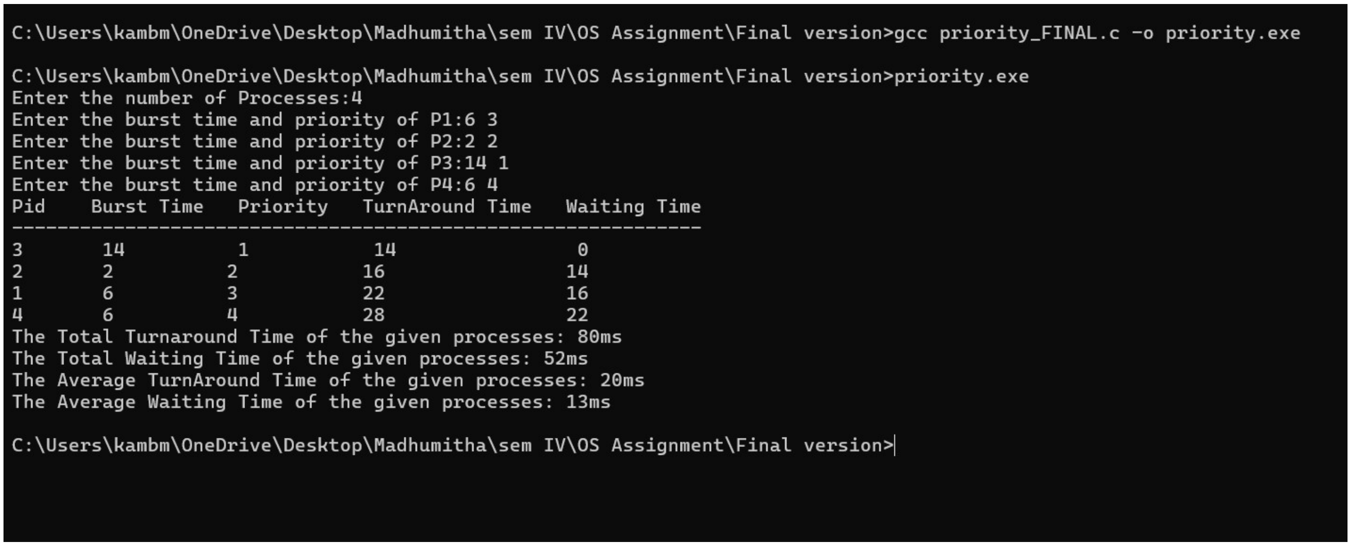
1. Get the number of processes from the user.
2. Read the process name, burst time and priority of process.
3. Sort based on burst time of all processes in ascending order based priority 4. Calculate the total waiting time and total turnaround time for each process 5. Display the process name & burst time for each process.
6. Display the total waiting time, average waiting time, turnaround time

Program Code:

```
#include<stdio.h>
#include<stdlib.h>
void sort(int bt[],int p[], int pid[],int n){
    for(int i=0;i<n-1;i++){
        for(int j=i+1;j<n;j++){
          if(p[i]>p[j]){
               int temp = p[i];
               p[i] = p[j];
               p[j] = temp;

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

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

void turnaround_time(int bt[],int p[], int pid[], int n,int tat[]){
     tat[0] = bt[0];
     for(int i=1;i<n;i++){
        tat[i] =  tat[i-1] + bt[i];
     }
}
```

```
void waiting_time(int n, int wt[], int bt[], int tat[]){
     for(int i=0;i<n;i++){
         wt[i] = tat[i] - bt[i];
     }
}

void calculate_avg(int tat[],int wt[], int n){
    int total_turnaround = 0, total_waiting = 0, avg_turnaround= 0, avg_waiting = 0;
    for(int i=0;i<n;i++){
        total_turnaround += tat[i];
        total_waiting += wt[i];
    }
    avg_turnaround = total_turnaround/n;
    avg_waiting = total_waiting/n;
    printf("The Total Turnaround Time of the given processes: %dms"
          "\nThe Total Waiting Time of the given processes: %dms"
          "\nThe Average TurnAround Time of the given processes: %dms"
          "\nThe Average Waiting Time of the given processes:
%dms\n",total_turnaround,total_waiting,avg_turnaround,avg_waiting);
}
int display(int pid[],int bt[], int p[],int tat[],int wt[],int n){
    printf("Pid    Burst Time   Priority   TurnAround Time   Waiting Time\n");
    printf("----------------------------------------------------------\n");
    for(int i=0;i<n;i++){
       printf("%d       %d         %d          %d              %d    \n",pid[i],bt[i],p[i],tat[i],wt[i]);
    }

}

int main(){
    int n;
    printf("Enter the number of Processes:");
    scanf("%d",&n);
    int bt[n],p[n],pid[n];
    for(int i=0;i<n;i++){
        printf("Enter the burst time and priority of P%d:",i+1);
        scanf("%d %d",&bt[i],&p[i]);
        pid[i] = i+1;
    }
    int tat[n],wt[n];
    sort(bt,p,pid,n);
    turnaround_time(bt,p,pid,n,tat);
    waiting_time(n,wt,bt,tat);

    display(pid,bt,p,tat,wt,n);
    calculate_avg(tat,wt,n);


}
```

Console:



```
C:\Users\kambm\OneDrive\Desktop\Madhumitha\sem IV\OS Assignment\Final version>gcc priority_FINAL.c -o priority.exe

C:\Users\kambm\OneDrive\Desktop\Madhumitha\sem IV\OS Assignment\Final version>priority.exe
Enter the number of Processes:4
Enter the burst time and priority of P1:6 3
Enter the burst time and priority of P2:2 2
Enter the burst time and priority of P3:14 1
Enter the burst time and priority of P4:6 4
Pid    Burst Time   Priority   TurnAround Time   Waiting Time
-------------------------------------------------------------
3       14          1           14                0
2       2          2           16                14
1       6          3           22                16
4       6          4           28                22
The Total Turnaround Time of the given processes: 80ms
The Total Waiting Time of the given processes: 52ms
The Average TurnAround Time of the given processes: 20ms
The Average Waiting Time of the given processes: 13ms

C:\Users\kambm\OneDrive\Desktop\Madhumitha\sem IV\OS Assignment\Final version>
```

Result: Thus, the program was executed successfully.