جامعة المستقبل
AL MUSTAQBAL UNIVERSITY

كلية العلوم

قسم الأنظمة الطبية الذكية
Intelligent Medical Systems Department

Lab (5)

CPU scheduling algorithms

Subject: OPERATING SYSTEMS

Class: Second

Lecturer: Dr.Ahmad almhanna

Eng .Jumana altahier

Programar :Ahyab hashim

Study Year: 2023-2024

Write a C prgram simulate the following CPU scheduling algorithms:

d) Priority

HARDWARE REQUIREMENTS: Intel based Desktop Pc RAM of 512 MB

SOFTWARE REQUIREMENTS: Turbo C/ Borland C.

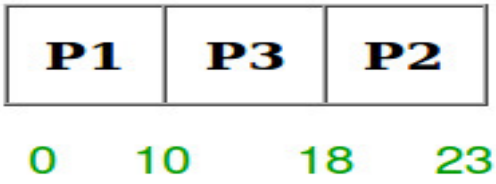
THEORY:

In Priority Scheduling, each process is given a priority, and higher priority methods are executed first, while equal priorities are executed First Come First Served or Round Robin. There are several ways that priorities can be assigned:

- Internal priorities are assigned by technical quantities such as memory usage, and file/IO operations.

- External priorities are assigned by politics, commerce, or user preference, such as importance and amount being paid for process access (the latter usually being for mainframes).

Process	Burst Time	Priority
P1	10	2
P2	5	0
P3	8	1



ALGORITHM

1. Start
2. Declare the array size
3. Read the number of processes to be inserted
4. Read the Priorities of processes
5. sort the priorities and Burst times in ascending order
6. calculate the waiting time of each process wt[i+1]=bt[i]+wt[i]
7. calculate the turnaround time of each process tt[i+1]=tt[i]+bt[i+1]
8. Calculate the average waiting time and average turnaround time.
9. Display the values
10. Stop

```
PROGRAM:
#include<stdio.h>
void main()
{
 int i,j,pno[10],prior[10],bt[10],n,wt[10],tt[10],w1=0,t1=0,s;
 float aw,at;
 printf("enter the number of processes:");
 scanf("%d",&n);
 for(i=0;i<n;i++)
 {
 printf("The process %d:\n",i+1);
 printf("Enter the burst time of processes:");
 scanf("%d",&bt[i]);
 printf("Enter the priority of processes %d:",i+1);
 scanf("%d",&prior[i]);
 pno[i]=i+1;
 }
 for(i=0;i<n;i++)
 {
 for(j=0;j<n;j++)
 {
 if(prior[i]<prior[j])
 {
s=prior[i];
prior[i]=prior[j];
prior[j]=s;
s=bt[i];
bt[i]=bt[j];
bt[j]=s;
s=pno[i];
 pno[i]=pno[j];
 pno[j]=s;
}
}
 }
 for(i=0;i<n;i++)
 {
 wt[0]=0;
 tt[0]=bt[0];
wt[i+1]=bt[i]+wt[i];
tt[i+1]=tt[i]+bt[i+1];
w1=w1+wt[i];
t1=t1+tt[i];
aw=w1/n;
   at=t1/n;
}
printf(" \n job \t bt \t wt \t tat \t prior\n");
 for(i=0;i<n;i++)
 printf("%d \t %d \t %d\t %d\t %d\n",pno[i],bt[i],wt[i],tt[i],prior[i]);
 printf("aw=%f \t at=%f \n",aw,at);
 }
```

```
Input:
Enter no of jobs
4
Enter bursttime
10
2
4
7
Enter priority values
4
2
1
3
Output:
Bt priority wt tt
4 1 0 4
2 2 4 6
7 3 6 13
10 4 13 23
aw=5.750000
at=12.500000
```


THANK YOU!