**CPU SCHEDULING-PRIORITY**

SANJANA S NAIR

CSE C

33

#include<stdio.h>

#include<string.h>

struct process

{

int at,bt,ct,tt,wt,pt;

char name[50];

}p[20],temp;

int main()

{

int n,k=0,g\_time=0,time\_taken=0;

float sum\_tt=0.0,sum\_wt=0.0;

printf("\n PRIORITY - Non Preemptive\n");

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

scanf("%d", &n);

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

printf("\nEnter the name of the process %d: ", (i+1));

fgets(p[i].name, 20, stdin);

printf(" Enter the arrival time of the process : ");

scanf("%d", &p[i].at);

printf(" Enter the burst time of the process : ");

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

printf(" Enter the priority of the process : ");

scanf("%d", &p[i].pt);

p[i].name[strcspn(p[i].name, "\n")] = 0;

}

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

{

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

{

if(p[i].at>p[j].at)

{

temp = p[i];

p[i] = p[j];

p[j] = temp;

}

}

}

int i = 0,j = 0;

printf("Process Table ");

printf("\nProcess Name \t| Process AT \t| Process BT \t| Process CT \t| Process TT \t| Process WT \t|

Priority\n");

while(i<n)

{

if(time\_taken >= p[i].at)

{

j = i+1;

while(j<n)

{

if(p[j].pt < p[i].pt && p[j].at<=time\_taken)

{

temp = p[j];

p[j] = p[i];

p[i] = temp;

}

j++;

}

time\_taken += p[i].bt;

p[i].ct = time\_taken;

p[i].tt = p[i].ct - p[i].at;

p[i].wt = p[i].bt - p[i].tt;

sum\_tt += p[i].tt;

sum\_wt += p[i].wt;

printf("\n%s \t\t %d \t\t %d \t\t %d \t\t %d \t\t %d \t\t %d",p[i].name, p[i].at, p[i].bt, p[i].ct,

p[i].tt, p[i].wt, p[i].pt);

i++;

}

else

{

time\_taken = p[i].at;

}

}

printf("updated process table\n");

printf("pid\tat\tbt\tct\ttt\twt\n");

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

{

printf("%s\t%d\t%d\t%d\t%d\t%d\n",p[i].name,p[i].at,p[i].bt,p[i].ct,p[i].tt,p[i].wt);

}

while(k<n)

{

if(gtime>=p[k].at)

{

printf("| %s\t",p[k].name);

gtime=gtime+p[k].bt;

k++;

}

else

{

printf("| idle\t");

gtime = p[k].at;

}

}

printf("|\n|0\t|");

k=0,gtime=0;

while(k<n)

{

if(gtime>=p[k].at)

{

printf("%d\t",p[k].ct);

gtime=gtime+p[k].bt;

k++;

}

else

{

gtime=p[k].at;

printf("%d\t",gtime);

}

}

**}**

**SAMPLE OUTPUT**

Enter the no of process3

Enter the process namep1

Enter the arrival time0

Enter the burst time2

Enter the priority1

Enter the process namep2

Enter the arrival time2

Enter the burst time3

Enter the priority4

Enter the process namep3

Enter the arrival time2

Enter the burst time1

Enter the priority2

process PR AT BT CT TAT WT

p1 1 0 2 2 2 0

p2 4 2 3 6 4 1

p3 2 2 1 3 1 0

Gantt chart

| p1 | p3 | p2 |

0 2 3 6

Average turnaround time=2.33

Average waiting time=0.33