11 YASHRAJ DEEPAK DEVRAT

#include<iostream>

using namespace std;

int process[]={1,2,3,4,5};

int AT[]={3,0,9,6,8};

int BT[]={3,5,10,7,2};

int rem\_BT[]={3,5,10,7,2};

int WT[]={0,0,0,0,0};

int TAT[]={0,0,0,0,0};

int priority[]={2,1,3,5,4};

class cpu\_scheduling

{

public:void FCFS()

{

int i,j,temp;

for(i=0;i<5;i++)

{

for(j=i+1;j<5;j++)

{

if(AT[i]>AT[j])

{

temp=AT[i];

AT[i]=AT[j];

AT[j]=temp;

temp=priority[i];

priority[i]=priority[j];

priority[j]=temp;

temp=process[i];

process[i]=process[j];

process[j]=temp;

temp=BT[i];

BT[i]=BT[j];

BT[j]=temp;

}

}

}

cout<<"\nSorted on FCFS"<<endl;

for(i=0;i<5;i++)

{

cout<<process[i],BT[i],AT[i];

}

operating\_system();

}

void Priority()

{

int i,j,temp;

for(i=0;i<5;i++)

{

for(j=i+1;j<5;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=BT[i];

BT[i]=BT[j];

BT[j]=temp;

}

}

}

cout<<"\nSorted on priority"<<endl;

for(i=0;i<5;i++)

{

cout<<process[i],BT[i],priority[i];

}

operating\_system();

}

void SJF()

{

float avg\_WT=0;

float avg\_TAT=0;

int temp;

int i,j;

for(i=0;i<5;i++)

{

for(j=i+1;j<5;j++)

{

if(BT[i]>BT[j])

{

temp=BT[i];

BT[i]=BT[j];

BT[j]=temp;

temp=process[i];

process[i]=process[j];

process[j]=temp;

}

}

}

cout<<"\nSorted on BT"<<endl;

for(i=0;i<5;i++)

{

cout<<process[i],BT[i];

}

operating\_system();

}

void round\_robin()

{

int temp;

for(int i=0;i<5;i++)

{

for(int j=i+1;j<5;j++)

{

if(AT[i]>AT[j])

{

temp=AT[i];

AT[i]=AT[j];

AT[j]=temp;

temp=priority[i];

priority[i]=priority[j];

priority[j]=temp;

temp=process[i];

process[i]=process[j];

process[j]=temp;

temp=BT[i];

BT[i]=BT[j];

BT[j]=temp;

}

}

}

cout<<"\nSorted on FCFS"<<endl;

for(int i=0;i<5;i++)

{

cout<<process[i],BT[i],AT[i];

}

int quantum;

float avg\_TAT, avg\_WT;

cout<<"\nEnter the time quantum:- "<<endl;

cin>>quantum;

int t = 0;

while (1)

{

bool done = true;

for (int i = 0 ; i < 5; i++)

{

if (rem\_BT[i] > 0)

{

done = false;

if (rem\_BT[i] > quantum)

{

t += quantum;

rem\_BT[i] -= quantum;

}

else

{

t = t + rem\_BT[i];

WT[i] = t - BT[i];

rem\_BT[i] = 0;

}

}

}

if (done == true)

break;

}

for (int i=0;i<5;i++)

{

TAT[i] = BT[i] +WT[i];

}

for (int i=0; i<5; i++)

{

avg\_WT= avg\_WT + WT[i];

avg\_TAT=avg\_TAT + TAT[i];

}

avg\_WT=avg\_WT/5;

avg\_TAT=avg\_TAT/5;

cout<<"\n------------------------------------------------------------------------------------------"<<endl;

cout<<"\nProcess\t Arrival-Time\t Burst-time\t Waiting-time\t Turnaround-time\t Priority"<<endl;

cout<<"\n-------------------------------------------------------------------------------------------"<<endl;

for(int i=0;i<5;i++)

{

cout<<"\n"<<process[i]<< "\t \t"<<AT[i]<<"\t \t"<<BT[i]<<"\t \t"<<WT[i]<<"\t \t"<<TAT[i]<<"\t\t"<<priority[i];

}

cout<<"\n-------------------------------------------------------------------------------------------"<<endl;

cout<<"\n Average Waiting Time = "<<avg\_WT;

cout<<"\n Average Turn-around Time = "<<avg\_TAT;

cout<<"\n \n"<<endl;

cout<<"The gannt chart of Round Robin can't be printed as the time quantum is same"<<endl;

}

void operating\_system()

{

float avg\_WT=0;

float avg\_TAT=0;

int i,j;

WT[0]=0;

TAT[0]=BT[0]+WT[0];

avg\_TAT=TAT[0];

for(i=1;i<5;i++)

{

WT[i]=WT[i-1]+BT[i-1];

TAT[i]=BT[i]+WT[i];

avg\_WT=avg\_WT+WT[i];

avg\_TAT=avg\_TAT+TAT[i];

}

avg\_WT=avg\_WT/5;

avg\_TAT=avg\_TAT/5;

cout<<"\n------------------------------------------------------------------------------------------"<<endl;

cout<<"\nProcess\t Arrival-Time\t Burst-time\t Waitingtime\t Turnaround-time\t Priority"<<endl;

cout<<"\n-------------------------------------------------------------------------------------------"<<endl;

for(i=0;i<5;i++)

{

cout<<"\n"<<process[i]<< "\t \t"<<AT[i]<<"\t \t"<<BT[i]<<"\t \t"<<WT[i]<<"\t \t"<<TAT[i]<<"\t\t"<<priority[i];

}

cout<<"\n-------------------------------------------------------------------------------------------"<<endl;

cout<<"\n Average Waiting Time = "<<avg\_WT;

cout<<"\n Average Turn-around Time = "<<avg\_TAT;

cout<<"\n \n"<<endl;

for(i=0;i<5;i++)

{

cout<<"|";

for (j=0;j<BT[i];j++)

{

cout<<"-";

}

}

cout<<"|";

cout<<"\n";

cout<<"\n";

}

};

int main()

{

cpu\_scheduling sb;

int ch;

while(1)

{

cout<<"\n-----SCHEDULING ALOGRITHM-----"<<endl;

cout<<"\n 1.FCFS"<<endl;

cout<<"\n 2.Shortest Job First"<<endl;

cout<<"\n 3.Priority"<<endl;

cout<<"\n 4.Round Robin"<<endl;

cout<<"\n 5. Exit"<<endl;

cout<<"\nEnter your choice: "<<endl;

cin>>ch;

switch(ch)

{

case 1:

sb.FCFS();

break;

case 2:

sb.SJF();

break;

case 3:

sb.Priority();

break;

case 4:

sb.round\_robin();

break;

case 5:

exit(1);

default:

cout<<"\n Invalid Choice!"<<endl;

}

}

return 0;

}







