EXPERIMENT NO   : C-10

TITLE   : Write a Java program (using OOP features) to implement following scheduling algorithms: FCFS , SJF (Preemptive), Priority (Non-Preemptive) and Round Robin (Preemptive)

NAME   :

CLASS   :
ROLL NO   : 29

DATE   :   **BATCH** : T2

```
*****************************************************************************
```

FCFS:

```java
import java.util.Scanner;
class Fcfs
{
public static void main(String args[]){
int bst[],process[],wt[],tat[],i,j,n,total=0,pos,temp;
float wait_avg, TAT_avg;

Scanner s = new Scanner(System.in);
System.out.print("Enter number of process: ");
n = s.nextInt();

   process = new int[n];
   bst = new int[n];
   wt = new int[n];
   tat = new int[n];

   System.out.println("\nEnter CPU time:");
   for(i=0;i<n;i++)
   {
   System.out.print("\nProcess["+(i+1)+"]: ");
   bst[i] = s.nextInt();;
   process[i]=i+1; //Process Number
   }


   System.out.println("\t\t\t**********FCFS Scheduling*********");
   //First process has 0 waiting time
  wt[0]=0;
   //calculate waiting time
   for(i=1;i<n;i++)
   {
   wt[i]=0;
   for(j=0;j<i;j++)
   wt[i]+=bst[j];
   total+=wt[i];
```

```java
    }
    //Calculating Average waiting time
    wait_avg=(float)total/n;
    total=0;

System.out.println("-------------------------------------------------------------------------");
System.out.println("\nProcess\t\t| Burst Time \t\t|Waiting Time\t\t|Turn Time");
System.out.println("-------------------------------------------------------------------------");
    for(i=0;i<n;i++)
    {
    tat[i]=bst[i]+wt[i];
    total+=tat[i];//Calculating TurnaroundTimetotal+=tat[i];
    System.out.println("\np"+process[i]+"\t\t\t"+bst[i]+"\t\t\t"+wt[i]+"\t\t\t"+tat[i]);
System.out.println("-------------------------------------------------------------------------");
 }//Calculation of Average Turnaround Time

TAT_avg=(float)total/n;
System.out.println("\n\nAverage Waiting Time: "+wait_avg);
System.out.println("\nAverage Turnaround Time: "+TAT_avg);
}
}
```

******************************Output******************************

C:\Users\ABI>cd Desktop

C:\Users\ABI\Desktop>javac Fcfs.java

C:\Users\ABI\Desktop>java Fcfs
Enter number of process: 3

Enter CPU time:

Process[1]: 20

Process[2]: 3

Process[3]: 4

              **********FCFS Scheduling*********

Process | Burst Time |Waiting Time |Turn Time

```
    ----------------------------------------------------------------------
    p1              |   20           |   0              |   20
    ----------------------------------------------------------------------
    p2              |   3            |   20             |   23
    ----------------------------------------------------------------------
    p3              |   4            |   23             |   27
    ----------------------------------------------------------------------
```

Average Waiting Time: 14.333333

Average Turnaround Time: 23.333334

<u>SJF (Preemptive):</u>

```java
import java.util.Scanner;
class SJF
{
        public static void main(String args[])
  {
        int burst_time[],process[],waiting_time[],tat[],i,j,n,total=0,pp,temp;
        float wait_avg,TAT_avg;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter number of process: ");
        n = s.nextInt();
        process = new int[n];
        burst_time = new int[n];
        waiting_time = new int[n];
        tat = new int[n];

        System.out.println("\nEnter Burst time:");

        for(i=0;i<n;i++)
        {
        System.out.print("\nProcess["+(i+1)+"]: ");
        burst_time[i] = s.nextInt();;
        process[i]=i+1; //Process Number
        }

        System.out.println("\n \t \t************** Shortest Job First Scheduling*********");
        //Sorting
        for(i=0;i<n;i++)
        {
        pp=i;



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

```java
        {
        if(burst_time[j]<burst_time[pp])
        pp=j;
        }
        temp=burst_time[i];
        burst_time[i]=burst_time[pp];
        burst_time[pp]=temp;
        temp=process[i];
        process[i]=process[pp];
        process[pp]=temp;
        }

        //First process has 0 waiting time

        waiting_time[0]=0;

        //calculate waiting time

        for(i=1;i<n;i++)
        {
        waiting_time[i]=0;
        for(j=0;j<i;j++)
        waiting_time[i]+=burst_time[j];
        total+=waiting_time[i];
        }

        //Calculating Average waiting time

        wait_avg=(float)total/n;
        total=0;
        System.out.println("--------------------------------------------------------------");
        System.out.println("\nProcess\t| Burst Time \t|Waiting Time\t|Turnaround Time |");
    System.out.println("--------------------------------------------------------------");
        for(i=0;i<n;i++)
        {
        tat[i]=burst_time[i]+waiting_time[i];

        //Calculating Turnaround Time

        total+=tat[i];
      System.out.println("\n p"+process[i]+"\t|\t "+burst_time[i]+"\t|\t "+waiting_time[i]+"\t|\t
"+tat[i]+"\t|\t ");
        System.out.println("--------------------------------------------------------------");
        }

        //Calculation of Average Turnaround Time

        TAT_avg=(float)total/n;
        System.out.println("\n\nAverage Waiting Time: "+wait_avg);
        System.out.println("\nAverage Turnaround Time: "+TAT_avg);
  }
 }
```

pvgcoen-4@pvgcoen4-ThinkCentre-M700:~$ cd Desktop
pvgcoen-4@pvgcoen4-ThinkCentre-M700:~/Desktop$ javac SJF.java
pvgcoen-4@pvgcoen4-ThinkCentre-M700:~/Desktop$ java SJF
Enter number of process: 3

Enter Burst time:

Process[1]: 20

Process[2]: 3

Process[3]: 4

    ************** Shortest Job First Scheduling**************

Process	Burst Time	Waiting Time	Turnaround Time
p2	3	0	3
p3	4	3	7
p1	20	7	27

Average Waiting Time: 3.3333333

Average Turnaround Time: 12.333333

Round Robin (Preemptive)

```java
import java.util.Scanner;

public class RR {

    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);

        int wtime[],btime[],rtime[],num,quantum,total;

        wtime = new int[10];
        btime = new int[10];
        rtime = new int[10];
```

```java
System.out.print("Enter number of processes(MAX 10): ");
num = s.nextInt();
System.out.print("Enter burst time");
for(int i=0;i<num;i++) { System.out.print("\nP["+(i+1)+"]: "); btime[i] = s.nextInt(); rtime[i] =
btime[i]; wtime[i]=0; } System.out.print("\n\nEnter quantum: "); quantum = s.nextInt(); int rp =
num; int i=0; int time=0; System.out.print("0"); wtime[0]=0; while(rp!=0) { if(rtime[i]>quantum)
 {
  rtime[i]=rtime[i]-quantum;
  System.out.print(" | P["+(i+1)+"] | ");
  time+=quantum;
  System.out.print(time);
  }
 else if(rtime[i]<=quantum && rtime[i]>0)
 {time+=rtime[i];
 rtime[i]=rtime[i]-rtime[i];
 System.out.print(" | P["+(i+1)+"] | ");
 rp--;
System.out.print(time);
 }

i++;
if(i==num)
{
i=0;
}

}

    }
}
```
*****************************OUTPUT**
pvgcoen-4@pvgcoen4-ThinkCentre-M700:~$ javac RR.java
pvgcoen-4@pvgcoen4-ThinkCentre-M700:~$ java RR
Enter number of processes(MAX 10): 3
Enter burst time
P[1]: 20

P[2]: 3

P[3]: 4

Enter quantum: 5
0 | P[1] | 5 | P[2] | 8 | P[3] | 12 | P[1] | 17 | P[1] | 22 | P[1] | 27*/

Priority (Non-Preemptive)

```java
import java.util.Scanner;
public class priority {

public static void main(String args[]) {
Scanner s = new Scanner(System.in);
int x,n,p[],pp[],bt[],w[],t[],awt,atat,i;
p = new int[10];
pp = new int[10];
bt = new int[10];
w = new int[10];
t = new int[10];
//n is number of process
//p is process
//pp is process priority
//bt is process burst time
//w is wait time
// t is turnaround time
//awt is average waiting time
//atat is average turnaround time
System.out.print("Enter the number of process : ");
n = s.nextInt();
System.out.print("\n\t Enter CPU time---priority \n");
for(i=0;i<n;i++)
{
System.out.print("\nProcess["+(i+1)+"]:");
bt[i] = s.nextInt();
pp[i] = s.nextInt();
p[i]=i+1;
}
//sorting on the basis of priority
for(i=0;i<n-1;i++)
{
for(int j=i+1;j<n;j++)
{
if(pp[i]<pp[j])
{
x=pp[i];
pp[i]=pp[j];
pp[j]=x;
x=bt[i];
bt[i]=bt[j];
bt[j]=x;
x=p[i];
p[i]=p[j];
p[j]=x;
}
}
}
w[0]=0;
awt=0;
t[0]=bt[0];
atat=t[0];
```

```
for(i=1;i<n;i++)
 {
 w[i]=t[i-1];
 awt+=w[i];
 t[i]=w[i]+bt[i];
 atat+=t[i];
 }
//Displaying the process
System.out.println("----------------------------------------------------------------------");
 System.out.print("\n\nProcess \t\t |Burst Time \t\t |Wait Time \t\t |Turn Time  \n");
System.out.println("----------------------------------------------------------------------");
for(i=0;i<n;i++)
 System.out.print("\n"+p[i]+"\t\t| "+bt[i]+"\t\t| "+w[i]+"\t\t|"+t[i]+"\t\t| "+pp[i]+"\n");
System.out.println("----------------------------------------------------------------------");
awt/=n;
atat/=n;
 System.out.print("\n Average Wait Time : "+awt);
 System.out.print("\n Average Turn Around Time : "+atat);
 }
}
```

********************************OUTPUT**

C:\Users\ABI\Desktop>javac priority.java
C:\Users\ABI\Desktop>java priority
Enter the number of process : 5

     Enter CPU time---priority

Process[1]:3
4

Process[2]:2
1

Process[3]:1
2

Process[4]:3
4

Process[5]:2
1

Process |Burst Time |Wait Time |TurnTime
----

1 | 3 | 0 |3 | 4

4 | 3 | 3 |6 | 4

3		1		6		7		2
2		2		7		9		1
5		2		9		11		1

 Average Wait Time : 5

 Average Turn Around Time : 7