

Ripunjay Narula 19BCE0470
Java Lab Digital Assignment

Multithreading:

```
1.import java.util.Scanner;
class EvenThread extends Thread
{
    int n;
    EvenThread(int n)
    {
        this.n=n;
    }
    public void run()
    {
        System.out.println("Even Nubers are: ");
        for(int i=1;i<=n;i++)
        {
            if(i%2==0)
            {
                System.out.println(i);
            }
        }
    }
}
class OddThread extends Thread
{
    int n;
    OddThread(int n)
    {
        this.n=n;
    }
    public void run()
    {
        System.out.println("Odd Numbers are: ");
        for(int i=1;i<=n;i++)
        {
            if(i%2!=0)
            {
                System.out.println(i);
            }
        }
    }
}
public class mt_oddeve
{
    public static void main(String arg[])
    {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        EvenThread et = new EvenThread(n);
        OddThread ot = new OddThread(n);
        et.start();
        ot.start();
    }
}
```

The image shows a screenshot of an IDE (likely IntelliJ IDEA) with a Java source file named `mt_oddeve.java` and its execution output.

Java Source Code:

```
1  public void run()
2  {
3      System.out.println("Even Nubers are: ");
4      for(int i=1;i<=n;i++)
5      {
6          if(i%2!=0)
7          {
8              System.out.println(i);
9          }
10     }
11 }
12
13 class OddThread extends Thread
14 {
15     int n;
16     OddThread(int n)
17     {
18         this.n=n;
19     }
20     public void run()
21     {
22         System.out.println("Odd Numbers are: ");
23         for(int i=1;i<=n;i++)
24         {
25             if(i%2!=0)
26             {
27                 System.out.println(i);
28             }
29         }
30     }
31 }
32
33 public class mt_oddeve
34 {
35     public static void main(String arg[])
36     {
37         Scanner input=new Scanner(System.in);
38         System.out.println("Enter Range: ");
39         int n=input.nextInt();
40         EvenThread e= new EvenThread(n);
41         OddThread o= new OddThread(n);
42         e.start();
43         o.start();
44     }
45 }
```

Execution Output:

```
C:\Windows\System32\cmd.exe
D:\VTOP\java-19bce0470>javac mt_oddeve.java
D:\VTOP\java-19bce0470>java mt_oddeve
Enter Range:
6
Even Nubers are:
2
4
6
Odd Numbers are:
1
3
5
D:\VTOP\java-19bce0470>
```

```
}
```

```
public void run()
```

```
{
```

```
    for(int i=0;i<6;i++)
```

```
    {
```

```
        try
```

```
        {
```

```
            Thread.sleep(1000);
```

```
        }
```

```
        catch(InterruptedException e)
```

```
        {
```

```
            System.out.println("Hello");
```

```
        }
```

```
System.out.println("Hello");
```

```
    }
```

```
}
```

```
}
```

```
class Two implements Runnable
```

```
{
```

```
    Two(){
```

```
        new Thread(this,"two").start();
```

```
}
```

```

public void run()
{
    for(int i=0;i<6;i++)
    {
        try
        {
            Thread.sleep(3000);

        }
        catch(InterruptedException e)
        {
            System.out.println("Welcome to VIT");

        }
        System.out.println("Welcome to VIT");
    }
}
}

public class multithreading1
{
    public static void main(String args[])
    {
        One o1 = new One();

        Two t1 = new Two();
    }
}

```

```
}
```

The screenshot shows a Notepad++ window with a Java file named 'multithreading1.java'. The code defines a 'run()' method with a loop that sleeps for 1000ms and prints 'Hello' and 'Welcome to VIT'. It also defines a 'Two' class that implements 'Runnable' and starts a new thread. A command prompt window shows the output of the program, displaying 'Hello' and 'Welcome to VIT' multiple times.

```

21 public void run()
22 {
23     for(int i=0;i<5;i++)
24     {
25         try
26         {
27             Thread.sleep(1000);
28         }
29         catch (InterruptedException e)
30         {
31             System.out.println("Hello");
32             System.out.println("Welcome to VIT");
33         }
34     }
35 }
36
37 class Two implements Runnable
38 {
39     Two() {
40         new Thread(this, "Two").start();
41     }
42 }
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64

```

```

C:\Windows\System32\cmd.exe
D:\VTOP\java-19bce0470>javac multithreading1.java
D:\VTOP\java-19bce0470>java multithreading1
D:\VTOP\java-19bce0470>javac multithreading1.java
D:\VTOP\java-19bce0470>java multithreading1
Hello
Welcome to VIT
Hello
Welcome to VIT
Hello
Welcome to VIT
Hello
Welcome to VIT
Hello
Welcome to VIT
Hello
Welcome to VIT
D:\VTOP\java-19bce0470>

```

```

3.import java.util.Random;
class Vote {
    int[] arr;
    int A_vote;
    int B_vote;
    int C_vote;
    public void GenerateVotes() {
        Random r = new Random();
        this.arr = new int[240];
        for (int i = 0; i < 240; i++) {
            int a = r.nextInt(3);
            this.arr[i] = a + 1;
        }
        this.A_vote = 0;
        this.B_vote = 0;
        this.C_vote = 0;
    }
    synchronized public void VoteCount(int start, int fin) {
        int A = 0, B = 0, C = 0;
        for (int i = start; i < fin; i++) {
            if (this.arr[i] == 1) {
                A++;
            } else if (this.arr[i] == 2) {
                B++;
            } else {
                C++;
            }
        }
    }
}

```

```

        this.A_vote += A;
        this.B_vote += B;
        this.C_vote += C;
    }
}
class ThreadA extends Thread {
    Vote v;
    ThreadA(Vote v) {
        this.v = v;
    }
    public void run() {
        this.v.VoteCount(0, 60);
    }
}
class ThreadB extends Thread {
    Vote v;
    ThreadB(Vote v) {
        this.v = v;
    }
    public void run() {
        this.v.VoteCount(60, 120);
    }
}
class ThreadC extends Thread {
    Vote v;
    ThreadC(Vote v) {
        this.v = v;
    }
    public void run() {
        this.v.VoteCount(120, 180);
    }
}
class ThreadD extends Thread {
    Vote v;
    ThreadD(Vote v) {
        this.v = v;
    }
    public void run() {
        this.v.VoteCount(180, 240);
    }
}
}
public class election_mt {
    public static void main(String args[]) {
        Vote v = new Vote();
        v.GenerateVotes();
        ThreadA a = new ThreadA(v);
        ThreadB b = new ThreadB(v);
        ThreadC c = new ThreadC(v);
        ThreadD d = new ThreadD(v);
        a.start();
    }
}

```

```

        b.start();
        c.start();
        d.start();
        try {
            a.join();
            b.join();
            c.join();
            d.join();
        } catch (Exception e) {
            System.out.println("Exception has " + e);
        }
        if (v.A_vote >= v.B_vote && v.A_vote >= v.C_vote) {
            System.out.println("A is the winner with " + v.A_vote + " votes");
        } else if (v.B_vote >= v.A_vote && v.B_vote >= v.C_vote) {
            System.out.println("B is the winner with " + v.B_vote + " votes");
        } else {
            System.out.println("C is the winner with " + v.C_vote + " votes");
        }
        System.out.println("A: " + v.A_vote + " B: " + v.B_vote + " C: " +
v.C_vote);
    }
}

```

```

1  import java.util.Random;
2  class Vote {
3      int[] arr;
4      int A_vote;
5      int B_vote;
6      int C_vote;
7      public void GenerateVotes() {
8          Random r = new Random();
9          this.arr = new int[240];
10         for (int i = 0; i < 240; i++) {
11             int a = r.nextInt();
12             this.arr[i] = a + 1;
13         }
14         this.A_vote = 0;
15         this.B_vote = 0;
16         this.C_vote = 0;
17     }
18     synchronized public void VoteCount(int start) {
19         int A = 0, B = 0, C = 0;
20         for (int i = start; i < fin; i++) {
21             if (this.arr[i] == 1) {
22                 A++;
23             } else if (this.arr[i] == 2) {
24                 B++;
25             } else {
26                 C++;
27             }
28             this.A_vote += A;
29             this.B_vote += B;
30             this.C_vote += C;
31         }
32     }
33 }
34 class ThreadA extends Thread {
35     Vote v;
36     ThreadA(Vote v) {
37         this.v = v;
38     }
39     public void run() {
40         this.v.VoteCount(0, 60);
41     }
42 }
43 class ThreadB extends Thread {
44     Vote v;

```

```

C:\Windows\System32\cmd.exe
D:\VTOP\java-19bce0470>java election_mt
C is the winner with 84 votes
A:77 B: 79 C: 84
D:\VTOP\java-19bce0470>javac election_mt.java
D:\VTOP\java-19bce0470>java election_mt
C is the winner with 85 votes
A:72 B: 83 C: 85
D:\VTOP\java-19bce0470>java election_mt

```

