

Q.1 INPUT:-

```
import java.util.Scanner;

public class UnsynchronizedCounterTest {

    static class Counter {

        int count;

        void inc() {

            count = count+1;

        }

        int getCount() {

            return count;

        }

    }

    static Counter counter;

    static int numberOfIncrements;

    static class IncrementerThread extends Thread {

        public void run() {

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

                counter.inc();

            }

        }

    }

    public static void main(String[] args) {

        Scanner in = new Scanner(System.in); // For reading the user's inputs.

        while (true) {

            System.out.println();

            System.out.print("How many threads do you want to run (Enter 0 to end)? ");

            int numberOfThreads = in.nextInt();

            if (numberOfThreads <= 0)

                break;

        }

    }

}
```

```

do {

    System.out.println();

    System.out.println("How many times should each thread increment the counter? ");

    numberOfIncrements = in.nextInt();

    if (numberOfIncrements < 1) {

        System.out.println("Number of increments must be positive.");

        numberOfIncrements = 1;

    }

} while (numberOfIncrements <= 0);

System.out.println();

System.out.println("Using " + numberOfThreads + " threads.");

System.out.println("Each thread increments the counter "

    + numberOfIncrements + " times.");

System.out.println();

System.out.println("Working...");

System.out.println();

IncrementerThread[] workers = new IncrementerThread[numberOfThreads];

counter = new Counter();

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

    workers[i] = new IncrementerThread();

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

    workers[i].start();

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

    try {

        workers[i].join();

    }

    catch (InterruptedException e) {

    }

}

```

```

        System.out.println("The final value of the counter should be "
            + (numberOfIncrements*numberOfThreads));

        System.out.println("Actual final value of counter is: " + counter.getCount());

        System.out.println();

        System.out.println();

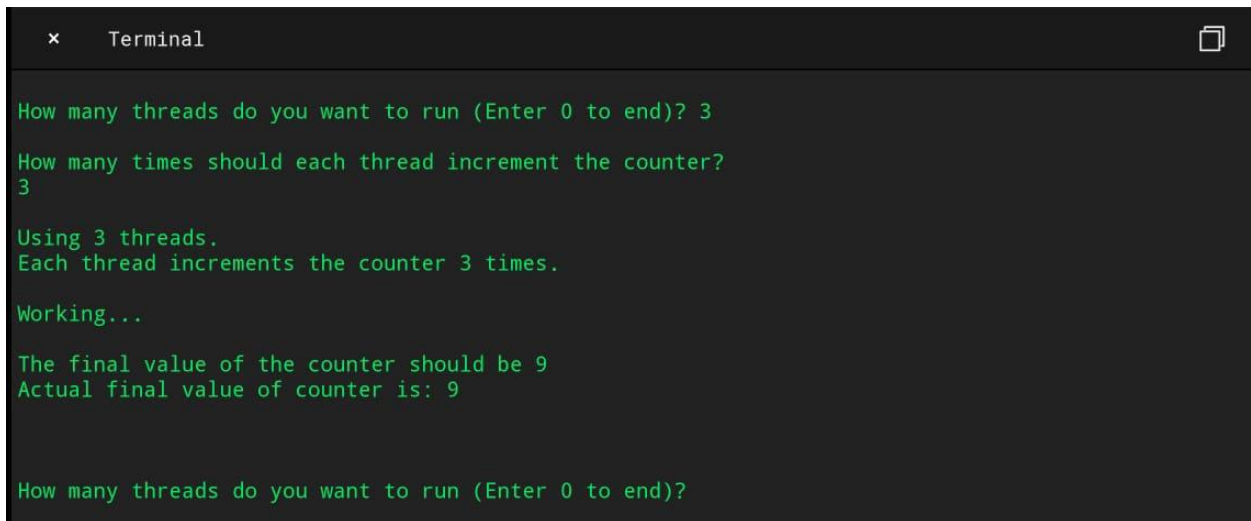
    }

}

}

```

OUTPUT:-



```

x Terminal

How many threads do you want to run (Enter 0 to end)? 3
How many times should each thread increment the counter?
3
Using 3 threads.
Each thread increments the counter 3 times.
Working...
The final value of the counter should be 9
Actual final value of counter is: 9

How many threads do you want to run (Enter 0 to end)?

```

Q.2 INPUT:-

```

import java.io.*;

import java.util.*;

class ThreadA extends Thread{

    public void run( ) {

        Random random = new Random();

        while(true){

            if(q2.size<20){

                int x = random.nextInt(101);

                q2.stack.push(x);
            }
        }
    }
}

```

```

        q2.size=q2.size+1;

        System.out.println("Pushed "+x+". Size = "+q2.size);
    }else{

        System.out.println("q2.stack Full");
    }

    try{

        Thread.sleep(700);
    }

    catch(Exception e){

        System.out.println(e);
    }

}

}

class ThreadB extends Thread {

    public void run( ) {

        Random random = new Random();

        while(true){

            if(q2.size>0){

                q2.size=q2.size-1;

                System.out.println("Popped "+q2.stack.pop()+". Size = "+q2.size);
            }else{

                System.out.println("q2.stack Empty");
            }

            try{

                Thread.sleep(1400);
            }

            catch(Exception e){

                System.out.println(e);
            }
        }
    }
}

```

```

        }
    }

}

public class q2{

    public static int size = 0;

    public static Stack<Integer> stack = new Stack<Integer>();

    public static void main (String[] args){

        ThreadA a = new ThreadA();

        ThreadB b = new ThreadB();

        a.start();

        b.start();

    }

}

```

OUTPUT:-

Q.3 INPUT:-

```

import java.util.*;

import java.io.*;

public class Sums {

    public static void sum(BufferedReader in){

        int s, nextInt;

        s = 0;
    }
}

```

```

try{

System.out.println("Please input the sequence of integers to sum, terminated by a 0");

nextInt = Integer.parseInt(in.readLine());

while (nextInt!=0) {

s = s + nextInt;

nextInt = Integer.parseInt(in.readLine());

}

System.out.println("The sum is " + s);

}catch(Exception e){

    System.out.println("Invalid Number. Please Reenter");

}

}

public static void main(String[] arg) {

BufferedReader in = new BufferedReader(new InputStreamReader(System.in));

String c;

try{

System.out.println("Do you wish to calculate a sum? (y/n)");

c = in.readLine();

while (!c.equals("y") && !c.equals("n")) {

System.out.println("Please answer y or n");

c = in.readLine();

}

while (c.equals("y")) {

sum(in);

System.out.println("Do you wish to calculate another sum? (y/n)");

c = in.readLine();

while (!c.equals("y") && !c.equals("n")) {

System.out.println("Please answer y or n");

c = in.readLine();

```

```

    }

    }

    System.out.println("Goodbye");

    }catch(Exception e){

        System.out.println("Please Input Y or N");

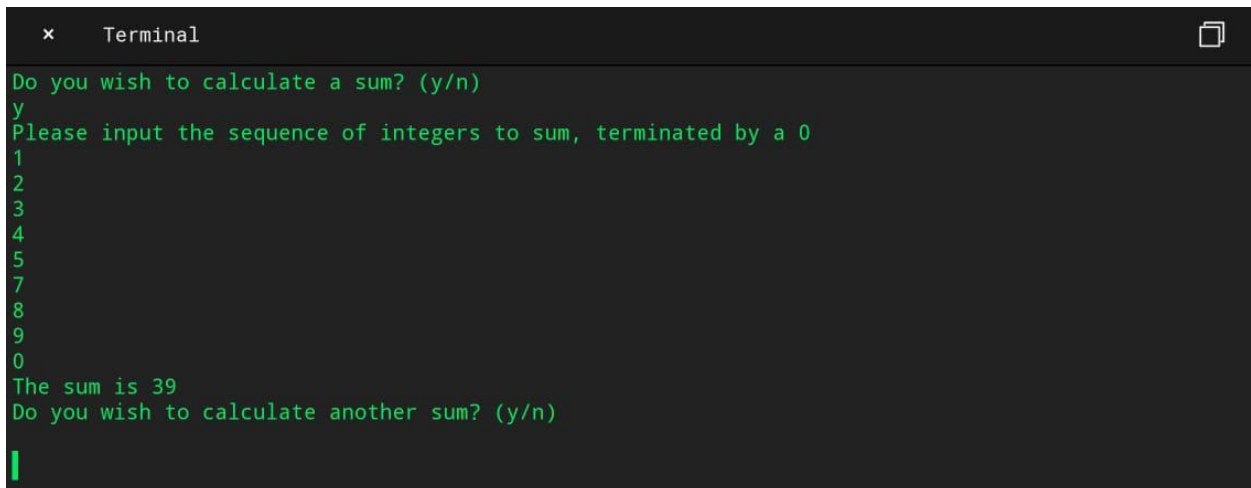
    }

    }

    }

```

OUTPUT:-



```

x Terminal
Do you wish to calculate a sum? (y/n)
y
Please input the sequence of integers to sum, terminated by a 0
1
2
3
4
5
7
8
9
0
The sum is 39
Do you wish to calculate another sum? (y/n)

```

Q.4 INPUT:-

```

import java.util.Scanner;

class MyCalculator {

    /*

    * Create the method long power(int, int) here.

    */

    public long power(int n, int p) throws Exception

    {

        if(n == 0 && p == 0)

            throw new Exception("n and p should not be zero.");
    }
}

```

```

        else if(n < 0 || p < 0)

            throw new Exception("n or p should not be negative.");

        else

            return (long)(Math.pow(n,p));

    }

}

public class Solution {

    public static final MyCalculator my_calculator = new MyCalculator();

    public static final Scanner in = new Scanner(System.in);

    public static void main(String[] args) {

        while (in .hasNextInt()) {

            int n = in .nextInt();

            int p = in .nextInt();

            try {

                System.out.println(my_calculator.power(n, p));

            } catch (Exception e) {

                System.out.println(e);

            }

        }

    }

}

```

OUTPUT:-