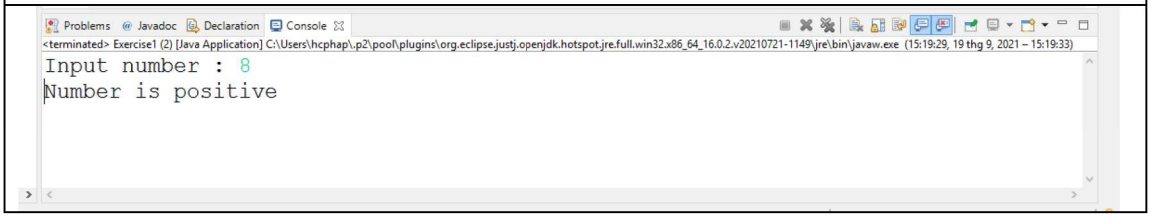
LAB 3
Chapter 3. Java fundamentals and control structures

1. Write an OOP program to get a integer from the keyboard and check if it positive, negative or zero?

```
import java.util.Scanner;
public class Exercise1 {
  public static void main(String[] args) {
    Scanner k=new Scanner(System.in);
    System.out.print("Input number : ");
    int n=k.nextInt();
    if(n==0)System.out.print("Zero");
    else
      if(n>0) System.out.print("Number is positive");
      else Systcm.out.print("Numbcr is ncgativc");
  }

}
```



2. Write an OOP program to solve a quadratic equation (ax2 + bx + c =0)

```
import java.util.Scanner;

public class QuadractiEquation {
  public static void main(String[] args) {

  float a,b,c,x1,x2,del;

  Scanner keyboard = new Scanner(System.in);

  System.out.print("a = ");
  a = keyboard.nextFloat();

  System.out.print("b = ");
  b = keyboard.nextFloat();

  System.out.print("c = ");
  c = keyboard.nextFloat();
```

```
        del = b*b - 4*a*c;
        if (del < 0) System.out.println("Roots are complex
                                          and different");
        else
         if (del==0)
          {
          x1=x2=-b/(2*a);
          System.out.println("Roots are real and same: "+x1);
          }
         else //means del > 0
           {
           x1=(float)(-b+Math.sqrt(del))/(2*a);
           x2=(float)(-b-Math.sqrt(del))/(2*a);
           System.out.println("Roots are real and different:
                               \n x1="+x1+"\n x2="+x2);
           }
        }
}
```

```
a = 3
b = 7
c = 2
oots are real and different:
 x1=-0.33333334
 x2=-2.0
```

Requirement: Students must complete the code above for complex roots

3. Write an OOP program allowing users to input three numbers (a, b, c) from keyboard. Check if a, b, c numbers are 3 edges of a triangle? If yes, print out the type of triangle?

```
import java.util.Scanner;

public class TriangleCheck {

 public static void main(String[] args) {

  float a,b,c;

  //nhập dữ liệu từ bàn phím
  Scanner keyboard = new Scanner(System.in);
  System.out.print("a = ");
  a = keyboard.nextFloat();

  System.out.print("b = ");
  b = keyboard.nextFloat();
```

```
  System.out.print("c = ");
  c = keyboard.nextFloat();

/*Check if 3 triangle edges are valid*/

  if ((a+b>c) && (a+c>b) && (b+c>a)&&(a>0) && (b>0)&&(c> 0))
   {
   System.out.print("a, b, c are valid \n");

/*Get type of the triangle*/
     if ((a==b)&&(b==c)) System.out.println("Equilateral
triangle");
      else
     if((a==b)||(b==c)||(a==c))
System.out.println("Isosceles triangle");
      else
       if
(((a*a+b*b==c*c)&&(a==b))||((a*a+c*c==b*b)&&(a==c))
               || ((c*c+b*b==a*a)&&(c==b)))
                          System.out.println("Isosceles right
triangle");
       else
       if ((a*a==b*b+c*c)||(b*b==a*a+c*c)||(c*c==a*a+b*b))
         System.out.println("Right triangle");
      else
         System.out.println("Triangle");
     }
   else //belong to the first if
       System.out.println("\"a, b, c are NOT valid \"");
  }
}
```

4. Write an OOP program to sum all even numbers from 2 to N.

```
import java.util.Scanner;
public class SumEven
{
 public static void main(String[] args)
    {
      Scanner keyboard = new Scanner(System.in);
      System.out.print("N = ");
      int N = keyboard.nextInt();

     int sum =0;
     for (int i=1;i<=N;i+=1)
         if (i%2==0) sum+=i;
```

```
        System.out.println("Sum = "+sum);
        }
}
```

5. Write an OOP program to check whether N (N>0), which got from the keyboard, is a prime number or not?

```
   import java.util.Scanner;
   public class PrimeNumber {
    public static void main(String[] args)
    {
     int N;
     int i;

     Scanner keyboard = new Scanner(System.in);
     System.out.print("N = ");
     N = keyboard.nextInt();

/*Kiểm tra số nguyên tố*/

     for (i=2;i<=Math.round(Math.sqrt(N));i++)
        if (N%i==0) break;/*Nếu chia hết cho một số i thì N
                           không phải số nguyên tố*/

/*Nếu i nhỏ hơn hoặc bằng căn bậc 2 N có nghĩa vòng lặp bị kết
thúc bằng câu lệnh break*/
     if (i <= Math.round(Math.sqrt(N)))
        System.out.print("N is not a prime number ");
     else System.out.print("N is a prime number ");
    }

}
```

```
Problems  Javadoc  Declaration  Console
<terminated> PrimeNumber [Java Application] C:\Users\hcphap\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_16.0.2.v20210721-1149\jre\bin\javaw.exe  (15:53:30, 19 thg 9, 2021 – 15:53:34)
N = 7
N is a prime number
```

DO IT YOURSELF

1. Write an application that prints all of integer the numbers between 1 (included) and 100 (included) using the "while" loop.
2. Write an application that prints all of the integer numbers between 1 (included) and 100 (included) using the "do..while" loop.

3. Write an application that prints all of the integer numbers between 1 (included) and 100 (included) using the "for" loop.
4. Write an application that prints all of the even numbers between 1 (included) and 100 (included) using the "while" loop.
5. Write an application that prints all of the even numbers between 1 (included) and 100 (included) using the "do..while" loop.
6. Write an application that prints all of the even numbers between 1 (included) and 100 (included) using the "for" loop.
7. Write an application that calculates the sum of all the integer numbers between 1 (included) and 100 (included).
8. Write an application that calculates and prints the average of all the integer numbers between 1 (included) and 100 (included).
9. Write an application that calculates and prints the biggest number between 1 (included) and 100 (included) that divides in 7 without a residual.
10. Write an application that calculates the sum of all the numbers between 1 (included) and 100 (included) that divide in 7 without a residual.
11. Write an application that prints all of the integer numbers between 1 (included) and 1000 (included) and near each one of them (each number will be printed in a new line) it prints EVEN or UNEVEN.
12. Write an application that prints a rectangle of stars as follows:

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

13. Write an application that prints a shape of stars as follows:

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

14. Write an application that prints to the screen the following:

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

15. Write an application that prints to the screen the following:

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

16. Write an application that prints to the screen the following series:

    0, 3, 8, 15, 24, 35, 48, . . .

17. Write an application that prints to the screen the following series:

    1, 3, 7, 15, 31, 63, . . .

18. Write an application that prints to the screen the following series:

    1, 7, 16, 37, 79, 173 ….

    The application should print the first 10 numbers of the series.

19. Write an application that prints to the screen the following series:

    1, 3, 9, 27, 81, 243 ….

    The application should print the first 10 numbers of the series.

20. The following application prints the factorial of 6. The computation of 6! is done using a static method that calculates the factorial of the number that it gets. Complete the missing lines.

```
class FactorialApplication
{
        public static void main(String args[])
        {
                long number, result;
                number = 6;
                result = factorial(number);
                System.out.println("The factorial of 6 is : " + result);
        }
        public static long factorial(long value)
        {
                long result = 1;
```

```
            //add the missing lines here
            return result;
        }
    }
```

21. Develop a stand-alone application that prints each one of the numbers between 0 and FFFF (the numbers should be printed in Hexadecimal base).
22. Develop a stand-alone application that prints each one of the numbers between 0 and 777 (the numbers should be printed in Octal base).