Programming 1

Tutorial 4

# Activity 1

Write a Java program to print out the first 10 natural numbers.

## Expected result

The first 10 natural numbers are:

1

2

3

4

5

6

7

8

9

10

# Activity 2

Write a program in Java to find out if a positive number is prime. A number is called prime if it is only divisible by itself and 1. Based on this definition, we can use loop to check if a given number is prime in our program.

## Expected result

Enter a positive integer: 17

It is a prime number.

Enter a positive integer: 15

It is not a prime number.

# Activity 3

Write a program to calculate your savings account balance after *y* years with an interest rate of *x* % per year. The interest is re-invested after each year.

## Expected result

How many years do you want to deposit your money? 10

How much money? 15300

What's the interest rate (%)? 6.8

After 10 years, you'll receive 29539.55

# Activity 4

Write a Java program to check if a number is a palindrome in Java.

A number is called a palindrome if the number is equal to the reverse of a number, e.g. 121 is a palindrome because the reverse of 121 is 121 itself. On the other hand 321 is not a palindrome because the reverse of 321 is 123, which is not equal to 321.

(\*) Hint:

Method 1: Find number R which is the reverse of N, if R = N then N is palindrome.

* Get the last digit: D = N % 10
* Add it to the beginning of number R: R = R \* 10 + D
* Remove that digit (the last digit) from N: N = N / 10

Method 2: Convert N into String, reverse the String, then compare with the original N.

# Activity 5

Write a program to check if a number entered by user is the power of 2. For example: 1, 2, 4 are power of 2, but 3, 10, 15 are not.

# Activity 6

Calculate the square root of a number without a built-in function (such as Math.sqrt()).

# Activity 7

Write Java program to find all Armstrong’s numbers. An Armstrong’s number is 3-digit number for which sum of cube of its digits are equal to itself. For example, 371 is an Armstrong number because of 3\*3\*3 + 7\*7\*7 + 1\*1\*1 = 371).

# Activity 8

Write a program in Java to reverse a user-entered String without using any built-in method (such as StringBuffer or StringBuilder).

# Activity 9

Write a program in Java to print the Fibonacci series up to a given number (entered by user) with the Java techniques that you’ve learned up to Lecture 4.

Fibonacci series a popular number series, in which starting from the 3rd number, the number is equal to the sum of the previous two numbers. The first two numbers are 0 and 1.

# Activity 10

Write a Java program to calculate Factorial of a user-entered integer using iteration. Factorial is defined as follows:

0! = 1

*n*! = *n* × (*n* – 1)!

Examples:

1! = 1

2! = 1 \* 2

3! = 1 \* 2 \* 3

# Activity 11

Write a program to print the following shape in Java using loop.

\*

\* \* \*

\* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \*

\* \* \*

\*