

Міністерство освіти, науки, молоді та спорту України  
Національний університет «Львівська політехніка»

**Кафедра СШ**

## Лабораторна робота №2

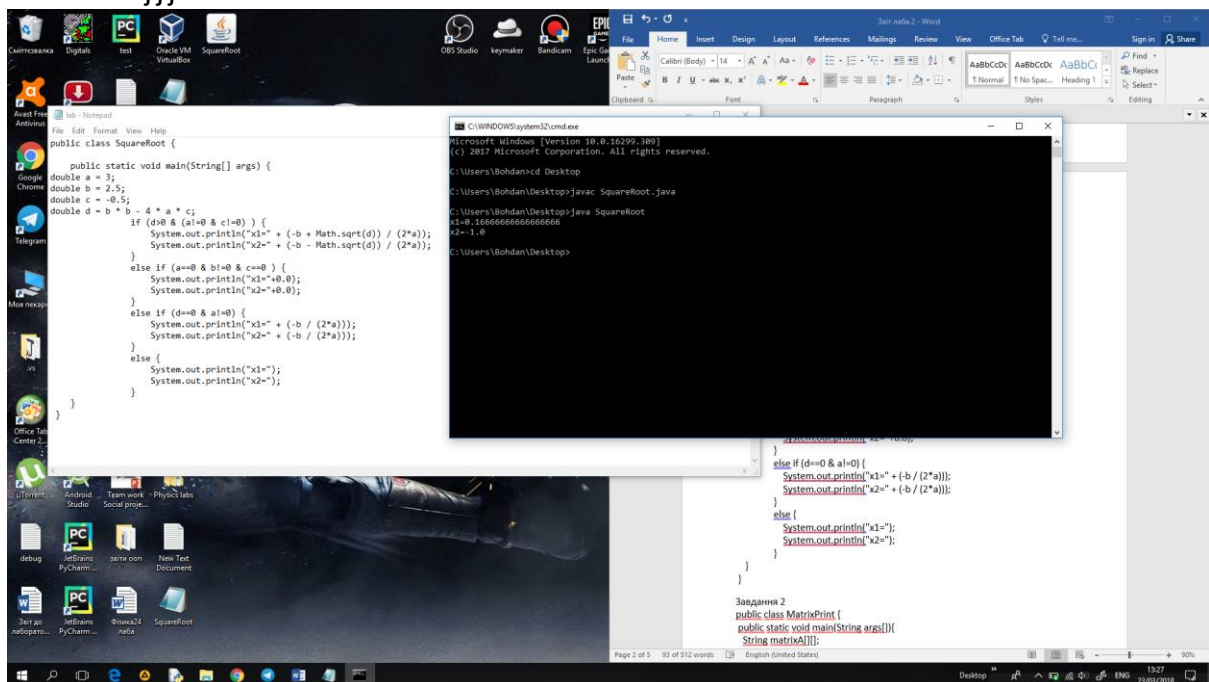
Виконав:  
ст. групи КН-  
107  
Гуменчук Б. Р.  
Прийняв:  
Старший  
викладач  
Гасько Р.Т.

Львів-2017

**Мета:** Пройти третій тиждень курсу вивчення Java на Prometheus, навчитись створювати зв'язний список і використовувати рекурсію в Java

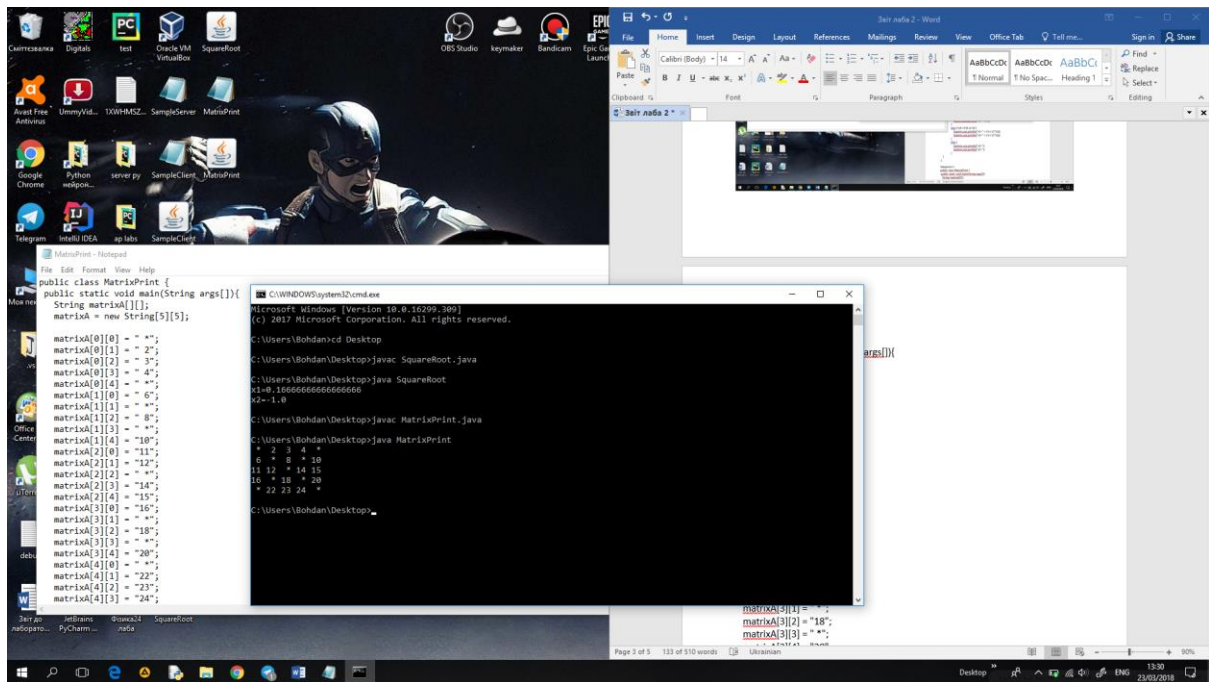
**Завдання 1**

```
public class SquareRoot {  
    public static void main(String[] args) {  
        double a = 3;  
        double b = 2.5;  
        double c = -0.5;  
        double d = b * b - 4 * a * c;  
        if (d>0 & (a!=0 & c!=0) ) {  
            System.out.println("x1=" + (-b + Math.sqrt(d)) / (2*a));  
            System.out.println("x2=" + (-b - Math.sqrt(d)) / (2*a));  
        }  
        else if (a==0 & b!=0 & c==0 ) {  
            System.out.println("x1="+0.0);  
            System.out.println("x2="+0.0);  
        }  
        else if (d==0 & a!=0) {  
            System.out.println("x1=" + (-b / (2*a)));  
            System.out.println("x2=" + (-b / (2*a)));  
        }  
        else {  
            System.out.println("x1=");  
            System.out.println("x2=");  
        }  
    }  
}
```



## Завдання 2

```
public class MatrixPrint {  
    public static void main(String args[]){  
        String matrixA[][];  
        matrixA = new String[5][5];  
  
        matrixA[0][0] = " *";  
        matrixA[0][1] = " 2";  
        matrixA[0][2] = " 3";  
        matrixA[0][3] = " 4";  
        matrixA[0][4] = " *";  
        matrixA[1][0] = " 6";  
        matrixA[1][1] = " *";  
        matrixA[1][2] = " 8";  
        matrixA[1][3] = " *";  
        matrixA[1][4] = "10";  
        matrixA[2][0] = "11";  
        matrixA[2][1] = "12";  
        matrixA[2][2] = " *";  
        matrixA[2][3] = "14";  
        matrixA[2][4] = "15";  
        matrixA[3][0] = "16";  
        matrixA[3][1] = " *";  
        matrixA[3][2] = "18";  
        matrixA[3][3] = " *";  
        matrixA[3][4] = "20";  
        matrixA[4][0] = " *";  
        matrixA[4][1] = "22";  
        matrixA[4][2] = "23";  
        matrixA[4][3] = "24";  
        matrixA[4][4] = " *";  
        for (int i = 0; i < 5; i++) {  
            for (int j = 0; j < 5; j++) {  
                System.out.print(matrixA[i][j] + " ");  
            }  
            System.out.println();  
        }  
    }  
}
```

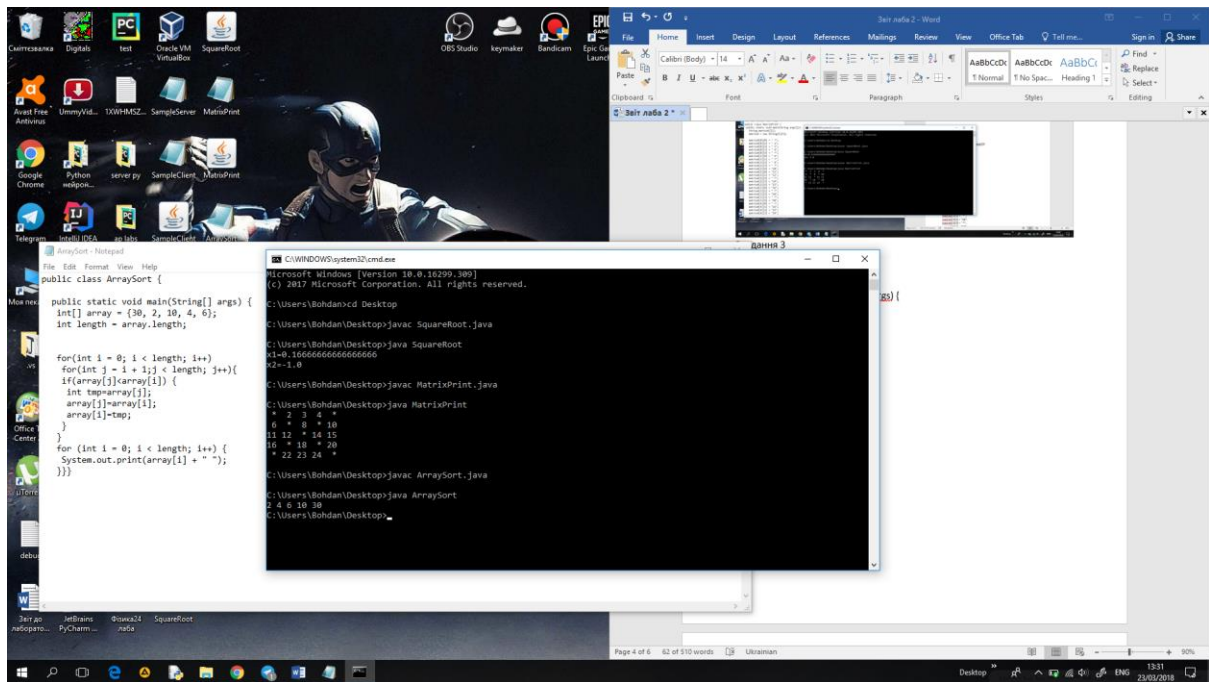


### Завдання 3

```
public class ArraySort {
```

```
    public static void main(String[] args) {
        int[] array = {30, 2, 10, 4, 6};
        int length = array.length;
```

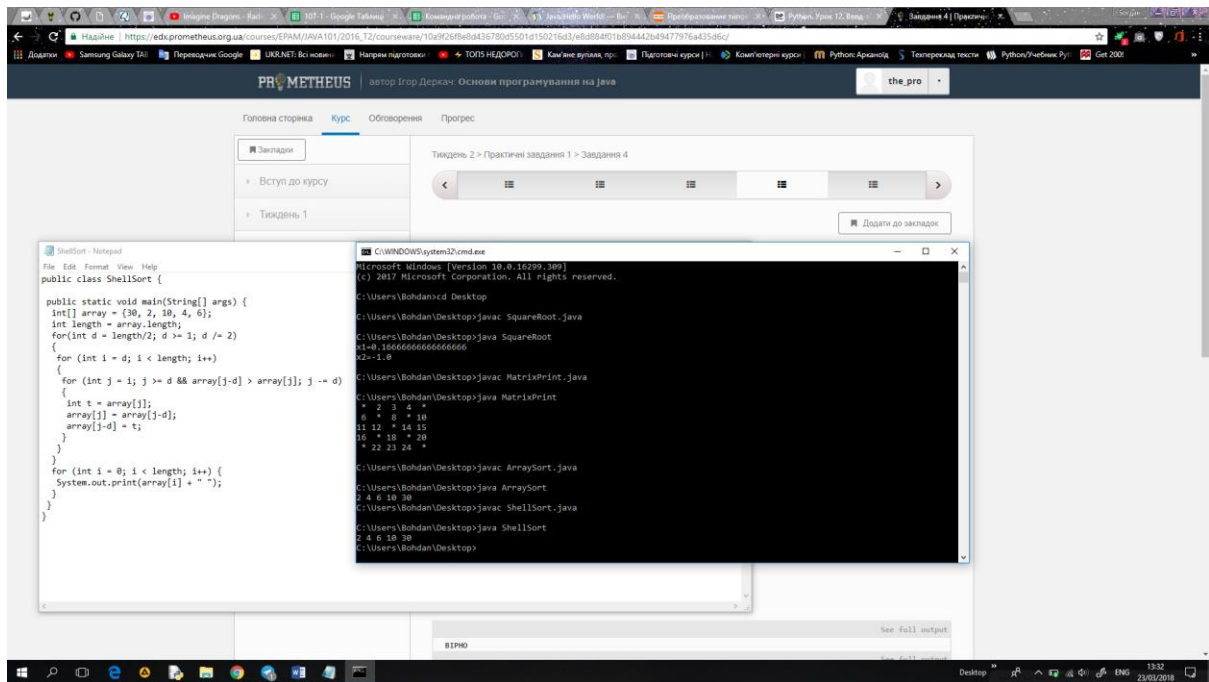
```
        for(int i = 0; i < length; i++)
            for(int j = i + 1; j < length; j++){
                if(array[j]<array[i]) {
                    int tmp=array[j];
                    array[j]=array[i];
                    array[i]=tmp;
                }
            }
        for (int i = 0; i < length; i++) {
            System.out.print(array[i] + " ");
        }
    }
}
```



## Завдання 4

```
public class ShellSort {
```

```
    public static void main(String[] args) {
        int[] array = {30, 2, 10, 4, 6};
        int length = array.length;
        for(int d = length/2; d >= 1; d /= 2)
        {
            for (int i = d; i < length; i++)
            {
                for (int j = i; j >= d && array[j-d] > array[j]; j -= d)
                {
                    int t = array[j];
                    array[j] = array[j-d];
                    array[j-d] = t;
                }
            }
        }
        for (int i = 0; i < length; i++) {
            System.out.print(array[i] + " ");
        }
    }
}
```



## Завдання 5

```
public class BinarySearch {

    public static void main(String[] args) {

        int data[] = { 3, 6, 7, 10, 34, 56, 60 };
        int numberToFind = 10;
        int averageIndex = 0;
        int firstIndex = 0;
        int lastIndex = data.length-1;
        while(firstIndex < lastIndex)
        {
            averageIndex = firstIndex + (lastIndex - firstIndex) / 2;
            if(numberToFind <= data[averageIndex])
            {
                lastIndex = averageIndex;
            }
            else
            {
                firstIndex = averageIndex + 1;
            }
        }
        if(data[lastIndex] == numberToFind)
        {
```

```

System.out.println(lastIndex);
}
else
{
    System.out.println(-1);
}
}
}

```

