



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційні систем та технологій

Лабораторна робота №7
із дисципліни «Основи програмування»
Тема: « Шаблон «Стратегія»»

Виконали:

Студенти групи ІА-24

Сіденко Д.Д.

Філімонов Є.А.

Іскандаров Е. Е. огли

Яблонський Д.Б.

Перевірив:

Колеснік Валерій Миколайович

Хід роботи:

1. Повторити теоретичні відомості
2. Провести рефакторинг свого коду з лабораторної роботи №6
 - використати шаблон «Стратегія»
 - дотримуватись принципів SOLID
3. Додати реалізацію ще одного алгоритму сортування на свій вибір
 - крім Array.sort()

```
package lab7;

interface Sorting {
    4 usages 3 implementations
    long[] sortSomething(long[] array);
    1 usage 3 implementations
    void printResult(long[] array);
}
```

```
package lab7;

import java.util.Arrays;
1 usage
public class SelectionSort implements Sorting{
    4 usages
    @Override
    public long[] sortSomething(long[] array) {
        //Inserting max element in its place for each iteration
        for (int maxIndex = 0; maxIndex < array.length; maxIndex++) {
            long maxVal = array[maxIndex];

            int maxIndexLocal = maxIndex;
            long maxValLocal = maxVal;
            //Selecting element with max value for present iteration
            for (int i = maxIndexLocal + 1; i < array.length; i++) {
                if (array[i] > array[maxIndexLocal]) {
                    maxValLocal = array[i];
                    maxIndexLocal = i;
                }
            }
            array[maxIndex] = maxValLocal;
            array[maxIndexLocal] = maxVal;
        }
        return array;
    }
}
```

```

package lab7;

import java.util.Arrays;
1 usage
public class InsertionSort implements Sorting{
    4 usages
    @Override
    public long[] sortSomething(long[] array) {
        // Selecting ab element for sorting
        for (int i = 1; i < array.length; i++) {
            long temVar = array[i];
            int location = i - 1;
            // Finding a place for the element and inserting
            while (location >= 0 && array[location] < temVar) {
                array[location + 1] = array[location];
                location -= 1;
            }
            array[location + 1] = temVar;
        }
        return array;
    }
    1 usage
    @Override
    public void printResult(long[] array) {
        String result = Arrays.toString(sortSomething(array));
        System.out.println("Sorted array: " + result);
    }
}

```

```

package lab7;

import java.util.Arrays;
1 usage
public class BubbleSort implements Sorting{
    4 usages
    @Override
    public long[] sortSomething(long[] array) {
        // Cycle for complete sorting
        for (int i = 0; i < array.length - 1; i++) {
            // Swapping two adjacent elements
            for (int j = 0; j < array.length - 1; j++) {
                if (array[j] < array[j + 1]) {
                    long temVar = array[j + 1];
                    array[j + 1] = array[j];
                    array[j] = temVar;
                }
            }
        }
        return array;
    }
    1 usage
    @Override
    public void printResult(long[] array) {
        String result = Arrays.toString(sortSomething(array));
        System.out.println("Sorted array: " + result);
    }
}

```

```

package lab7;

public class SortMachine {
    3 usages
    Sorting typeOfSorting;
    2 usages
    long[] array;
    3 usages
    public void setTypeOfSorting(Sorting typeOfSorting) { this.typeOfSorting = typeOfSorting; }
    public void setArray(long[] array) { this.array = array; }
    3 usages
    public void executeSorting() {
        long[] result = typeOfSorting.sortSomething(array);
        typeOfSorting.printResult(result);
    }
}

```

```

package lab7;

public class Test {
    public static void main(String[] args) {
        SortMachine firstMachine = new SortMachine();

        firstMachine.setTypeOfSorting(new SelectionSort());
        firstMachine.setArray(new Long[]{10293729, 42649321, 9238422, 948422, 2107846362, 3267843, 52, 10982347});
        firstMachine.executeSorting();

        firstMachine.setTypeOfSorting(new InsertionSort());
        firstMachine.setArray(new Long[]{10293729, 42649321, 9238422, 948422, 2107846362, 3267843, 2378, 10982347});
        firstMachine.executeSorting();

        firstMachine.setTypeOfSorting(new BubbleSort());
        firstMachine.setArray(new Long[]{10293729, 42649321, 9238422, 948422, 2107846362, 3267843, 182735, 10982347});
        firstMachine.executeSorting();
    }
}

```

Висновок: під час виконання лабораторної роботи ми навчились оформляти наш код відповідно до шаблону «Стратегія» та задовольнили деякі з принципів SOLID. В результаті ми отримали більш читабельну та зрозумілу програму, яку легше та зручніше підтримувати та оновлювати.