



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»»

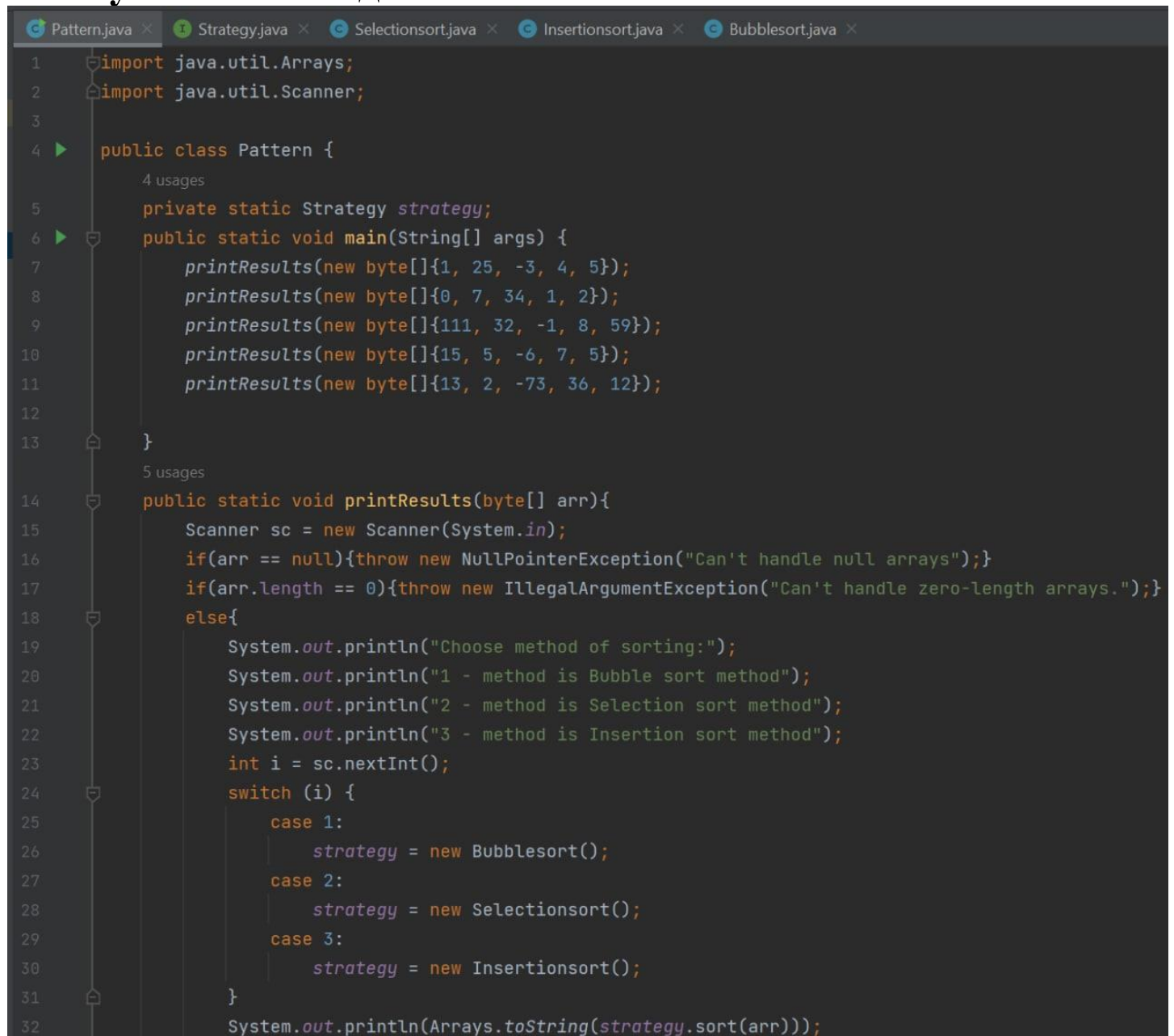
Виконали:
Студенти групи ІА-24
(бригада №1)
Чайка А.П.
Котлярчук М.С.
Коханчук М.М.

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()`
4. Відповісти на контрольні запитання

Результати + код:



```
Pattern.java x Strategy.java x Selectionsort.java x Insertionsort.java x Bubblesort.java x
1  import java.util.Arrays;
2  import java.util.Scanner;
3
4  public class Pattern {
5      4 usages
6      private static Strategy strategy;
7      public static void main(String[] args) {
8          printResults(new byte[]{1, 25, -3, 4, 5});
9          printResults(new byte[]{0, 7, 34, 1, 2});
10         printResults(new byte[]{111, 32, -1, 8, 59});
11         printResults(new byte[]{15, 5, -6, 7, 5});
12         printResults(new byte[]{13, 2, -73, 36, 12});
13     }
14     5 usages
15     public static void printResults(byte[] arr){
16         Scanner sc = new Scanner(System.in);
17         if(arr == null){throw new NullPointerException("Can't handle null arrays");}
18         if(arr.length == 0){throw new IllegalArgumentException("Can't handle zero-length arrays.");}
19         else{
20             System.out.println("Choose method of sorting:");
21             System.out.println("1 - method is Bubble sort method");
22             System.out.println("2 - method is Selection sort method");
23             System.out.println("3 - method is Insertion sort method");
24             int i = sc.nextInt();
25             switch (i) {
26                 case 1:
27                     strategy = new Bubblesort();
28                 case 2:
29                     strategy = new Selectionsort();
30                 case 3:
31                     strategy = new Insertionsort();
32             }
33             System.out.println(Arrays.toString(strategy.sort(arr)));
34         }
35     }
36 }
```

```
Pattern.java x Strategy.java x Selections
4 usages 3 implementations
1 public interface Strategy {
    1 usage 3 implementations
2     byte[] sort(byte[] arr);
3 }
```

1. Selection sort:

```
Pattern.java x Strategy.java x Selectionsort.java x Insertionsort.java x
1 usage
1 public class Selectionsort implements Strategy{
    1 usage
2     public byte[] sort(byte[] arr) {
3         for (int i = 0; i < (arr.length - 1); i++) {
4             int min_idx = i;
5             for (int j = i + 1; j < arr.length; j++) {
6                 if (arr[j] < arr[min_idx]) {
7                     min_idx = j;
8                 }
9             }
10            byte temp = arr[min_idx];
11            arr[min_idx] = arr[i];
12            arr[i] = temp;
13        }
14        return arr;
15    }
16 }
17 }
```

2. Insertion sort:

```
Pattern.java x Strategy.java x Selectionsort.java x Insertionsort.java x
1 usage
1 public class Insertionsort implements Strategy{
    1 usage
2 ① @ public byte[] sort(byte[] arr) {
3     {
4         int n = arr.length;
5         for (int i = 1; i < n; ++i) {
6             byte key = arr[i];
7             int j = i - 1;
8
9             /* Move elements of arr[0..i-1], that are
10              greater than key, to one position ahead
11              of their current position */
12             while (j >= 0 && arr[j] > key) {
13                 arr[j + 1] = arr[j];
14                 j = j - 1;
15             }
16             arr[j + 1] = key;
17         }
18     }
19     return arr;
20 }
21 }
```

3. Bubble sort:

```
Pattern.java x Strategy.java x Selectionsort.java x Insertionsort.java x Bubblesort.java x
1 usage
1 public class Bubblesort implements Strategy{
    1 usage
2 ① @ public byte[] sort(byte[] arr) {
3     byte k;
4     boolean sort = false;
5     if (arr.length == 0) {
6         throw new IllegalArgumentException("Can't handle zero-length arrays.");
7     } else {
8         while (!sort) {
9             sort = true;
10            for (int j = 0; j < arr.length - 1; j++) {
11                if (arr[j] > arr[j + 1]) {
12                    sort = false;
13                    k = arr[j];
14                    arr[j] = arr[j + 1];
15                    arr[j + 1] = k;
16                }
17            }
18        }
19    }
20    return arr;
21 }
22 }
```

```
Choose method of sorting:
1 - method is Bubble sort method
2 - method is Selection sort method
3 - method is Insertion sort method

1
[-3, 1, 4, 5, 25]
Choose method of sorting:
1 - method is Bubble sort method
2 - method is Selection sort method
3 - method is Insertion sort method

2
[0, 1, 2, 7, 34]
Choose method of sorting:
1 - method is Bubble sort method
2 - method is Selection sort method
3 - method is Insertion sort method

3
[-1, 8, 32, 59, 111]
Choose method of sorting:
1 - method is Bubble sort method
2 - method is Selection sort method
3 - method is Insertion sort method
```

Висновки:

У цій роботі ми провели рефакторинг свого коду та реалізували три методи сортування(бульбашкою, вибором та вставкою) за допомогою шаблону «Стратегія» у IDE за допомогою Java.