**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ**

**УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

ЛАБОРАТОРНАЯ РАБОТА №3

По дисциплине: **«Технологии разработки программного обеспечения»**

на тему: **Моделирование и алгоритмизация как средства проектирования программного обеспечения**

Выполнил: студент группы ИТП-21

Шинкарук В.Г.

Проверил:

Малиновский И.Л.

Гомель 2024

**Цель работы**: изучить особенности разработки в объектно-ориентированном языке программирования, используя механизмы моделирования и автоматизации. Развить представление о разработке программ на *Java*.

**Задание:**

1. Описать блок-схему алгоритма согласно варианту (таблица 3).

2. На основе описанной блок-схемы реализовать алгоритм средствами языка *Java*.

3. Протестировать разработанный алгоритм при помощи модульных тестов.

4. Создать консольное приложение для демонстрации работы алгоритма.

5. Составить отчет о проделанной работе.

**Ход выполнения и результаты работы:**

Текстовый вариант задания для варианта 23 представлен на рисунке 1.



Рисунок 1 – Задание варианта 23

На рисунке 2 представлен результат выполнения программы.

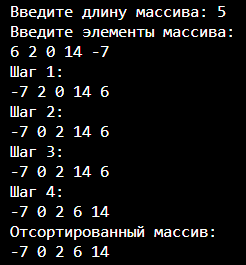


Рисунок 2 – Сортировка выбором

На рисунке 3 представлен результат выполненных тестов.

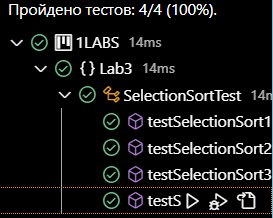


Рисунок 3 – Тесты

Текст класса *SelectionSort.java* представлен в приложении А.

Текст класса *SelectionSortTest.java* представлен в приложении Б.

**Вывод:** в ходе выполнения лабораторной работы изучены особенности разработки в объектно-ориентированном языке программирования, используя механизмы моделирования и автоматизации. В результате выполнения лабораторной работы разработана программа для сортировки выбором.

**ПРИЛОЖЕНИЕ А**

(обязательное)

**Текст *SelectionSort.java***

package Lab3;

import java.util.Scanner;

public class SelectionSort {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Введите длину массива: ");

int length = scanner.nextInt();

int[] array = new int[length];

System.out.println("Введите элементы массива:");

for (int i = 0; i < length; i++) {

array[i] = scanner.nextInt();

}

selectionSort(array);

System.out.println("Отсортированный массив:");

printArray(array);

scanner.close();

}

public static void selectionSort(int[] arr) {

int n = arr.length;

for (int i = 0; i < n-1; i++) {

int minIndex = i;

for (int j = i+1; j < n; j++) {

if (arr[j] < arr[minIndex]) {

minIndex = j;

}

}

int temp = arr[minIndex];

arr[minIndex] = arr[i];

arr[i] = temp;

// Выводим результат после каждого этапа сортировки

System.out.println("Шаг " + (i+1) + ":");

printArray(arr);

}

}

public static void printArray(int[] arr) {

for (int i = 0; i < arr.length; i++) {

System.out.print(arr[i] + " ");

}

System.out.println();

}

}

**ПРИЛОЖЕНИЕ Б**

(обязательное)

**Текст *SelectionSortTest.java***

package Lab3;

import static org.junit.Assert.\*;

import org.junit.Test;

public class SelectionSortTest {

@Test

public void testSelectionSort1() {

int[] arr = {64, 25, 12, 22, 11};

SelectionSort.selectionSort(arr);

assertArrayEquals(new int[]{11, 12, 22, 25, 64}, arr);

}

@Test

public void testSelectionSort2() {

int[] arr = {25, 9, 3, 15, 48, 6};

SelectionSort.selectionSort(arr);

assertArrayEquals(new int[]{3, 6, 9, 15, 25, 48}, arr);

}

@Test

public void testSelectionSort3() {

int[] arr = {-13, 6, -18, 123, 7, 87, 0};

SelectionSort.selectionSort(arr);

assertArrayEquals(new int[] {-18, -13, 0, 6, 7, 87, 123}, arr);

}

@Test

public void testSelectionSort4() {

int[] arr = {5, -5, 15, -25};

SelectionSort.selectionSort(arr);

assertArrayEquals(new int[] {-25, -5, 5, 15}, arr);

}

}