Институт информационных технологий и управления

в технических системах

Кафедра информационных технологий и компьютерных систем

ОТЧЕТ

по лабораторной работе № 8

**«ИССЛЕДОВАНИЕ АЛГОРИТМОВ ОБРАБОТКИ**

**ОДНОМЕРНЫХ МАССИВОВ»**

по дисциплине «Программирование. Базовые процедуры обработки информации»

Выполнил студент группы ИВТ/б-11д

Орлов И.В.

Проверил доцент Петров И.И.

Севастополь

2020

**1. ЦЕЛЬ РАБОТЫ**

Целью данной работы является исследование циклических алгоритмов и программ, осуществляющих типичные операции над одномерными массивами, получение навыков разработки и использования подпрограмм (методов-процедур и методов-функций), а также изучение возможностей отладчика BlueJ.

2. **ПОСТАНОВКА ЗАДАЧИ**

1) Ознакомиться с теоретическими сведениями, приведенными в пункте 2.4 методических указаний: структурирование данных в виде одномерных массивов и их обработка, работа с отладчиком BlueJ, структурирование программ: методы-процедуры и методы функции.

2) Разработать программу, реализующую обработку массива данных по заданию в соответствии с номером варианта. Программа должна удовлетворять нижеперечисленным требованиям.

Основная программа (метод main) должна вызывать четыре вспомогательные подпрограммы (четыре метода).

Первый метод-**функция** предназначен для вычисления значения в соответствии с заданием, указанным в столбце 3 таблицы вариантов (таблица 2.1).

Второй метод-**процедура** предназначен для выполнения сортировки элементов массива, метод и тип которой заданы столбцами 4 и 5 таблицы вариантов.

Третий метод-**процедура** предназначен для обработки массива способом, указанным в столбце 6 таблицы вариантов.

Четвертый метод-**процедура** предназначен для вывода в окно терминала всех элементов массива. Элементы массива должны быть выведены по **n элементов в строке, где n – параметр метода**.

Значения переменных А, В, m (в зависимости от варианта) **должны быть параметрами методов** и задаваться в основной программе.

Основная программа (метод main) должна выполнять следующие действия:

* инициализировать два различных массива (отличаются значения элементов и длины массивов); базовый тип массива (тип элементов) определен вариантом;
* для каждого из массивов вывести:

исходный массив (метод 4);

результат выполнения метода-функции для исходного массива (метод 1);

результат сортировки исходного массива (метод 2, метод 4);

результат выполнения метода-процедуры для исходного (не отсортированного) массива (метод 3, метод 4).

3) Продемонстрировать процесс отладки (пошаговое выполнение) программы с помощью отладчика BlueJ.

Вариант задания V вычислен по формуле

V = (N%14 != 0) ? N%14 : 14; ,

где N – номер студента в списке группы:

1%14=1;

Данные варианта задания приведены в таблице 2.1

Таблица 2.1 – Вариант задания

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Но-мер вари-анта | Базовый тип массива | Функция (метод 1)  должна возвращать  значение: | Метод сортировки  (для  метода 2) | Вид сорти-ровки  (для мето-да 2) | Процедура обработки массива  (метод 3) |
| 1 | int | Cумма положительных элементов массива | Пузырек 1 | По возр. | Увеличить на А значения элементов с четными индексами |

**3. АНАЛИЗ ЗАДАЧИ**

**3.1. Входные данные, выходные данные, функции**

**и структура программы**

Согласно заданию программа должна инициализировать два одномерных массива разной длины и выполнять их обработку с помощью четырех методов. Результаты обработки должны выводиться в окно терминала. Схема алгоритма вычислительного процесса изображена на рисунке 3.1. Как видно из рисунка 3.1 вызывающая программа будет иметь линейную структуру. Все подробности обработки массивов будут скрыты в соответствующих методах. Копия каждого из двух массивов создается для того, чтобы заданный метод обработки массива (метод 3) применялся не к отсортированному массиву, а к массиву с исходными значениями элементов.



Рисунок 3.1 – Схема алгоритма вычислительного процесса

Структура программы изображена на рисунке 3.2.

***Вставьте названия методов, которые будут использоваться в вашей программе***



Рисунок 3.2 – Структура программы

**3.2. Входные данные, выходные данные, выполняемые действия**

**для вызываемых методов**

Первый метод – method1 – предназначен для вычисления значения суммы положительных элементов массива. По заданию он должен быть реализован как функция. Формальный параметр метода: arr – локальная ссылочная переменная (ссылка на одномерный массив элементов типа int). При вызове метода переменной arr будет присвоено значение соответствующего фактического параметра, т.е. адрес массива, созданного в вызывающем методе (main). Таким образом, передача параметра методу осуществляется по ссылке. Метод возвращает значение типа int.

Второй метод – method2 – предназначен для сортировки элементов массива по возрастанию методом пузырька. По заданию метод должен быть реализован как процедура, т.е. он должен просто обрабатывать массив, ссылка на который передана ему в качестве фактического параметра, не возвращая никакого значения. Формальный параметр метода: arr – локальная ссылочная переменная (ссылка на одномерный массив элементов типа int). Передача фактического параметра методу осуществляется по ссылке. Тип возвращаемого значения – void. Метод пузырька, основанный на перестановке соседних элементов в массиве, относится к простым методам сортировки с асимптотической временн*о*й сложностью .

Третий метод – method3 – должен увеличивать на А значения элементов с четными индексами. По заданию метод должен быть реализован как процедура, т.е. он должен просто обрабатывать массив, ссылка на который передана ему в качестве фактического параметра, не возвращая никакого значения. Метод имеет два формальных параметра. Первый формальный параметр метода: arr – локальная ссылочная переменная (ссылка на одномерный массив элементов типа int). Передача первого параметра осуществляется по ссылке. Второй формальный параметр – локальная переменная А типа int. При вызове метода переменной А будет присвоено значение соответствующего фактического параметра, т.е. переменной или int-литерала, указанного в вызове метода. Таким образом, второй параметр передается по значению. Тип возвращаемого значения – void.

Схема алгоритма метода 3 изображена на рисунке 3.3.

***Схема алгоритма должна быть изображена аккуратно и достаточно крупно***

Рисунок 3.3 – Схема алгоритма метода method3

Четвертый метод – method4 – предназначен для вывода в окно терминала всех элементов массива по n элементов в строке. Метод имеет два формальных параметра. Первый формальный параметр метода: arr – локальная ссылочная переменная (ссылка на одномерный массив элементов типа int). Передача первого параметра осуществляется по ссылке. Второй формальный параметр – локальная переменная n типа int. Передача второго параметра осуществляется по значению. Тип возвращаемого значения – void (метод не возвращает значения).

**4. ТЕСТОВЫЕ ПРИМЕРЫ И РЕЗУЛЬТАТЫ ИХ ОБРАБОТКИ ВРУЧНУЮ**

*В данном пункте нужно привести пример одномерного массива и ожидаемые результаты его обработки. Если есть частные случаи, в которых содержимое массива существенно влияет на результат обработки (например, требуется найти номер элемента со значением А, а такого элемента в массиве нет), нужно придумать тестовые примеры для таких частных случаев и привести ожидаемые результаты обработки для каждого тестового примера.*

**6.ТЕКСТ ПРОГРАММЫ**

Согласно пункту 2.8.1. методических указаний к лабораторным работам, текст программы должен быть отформатирован следующим образом:

1. шрифт: arial, 12 пт, полужирный;
2. абзац:

* выравнивание по левому краю;
* уровень – основной текст;
* отступ слева – 0;
* отступ справа – 0;
* интервал перед – 0;
* интервал после – 0;
* первая строка – нет (отступа или выступа),
* междустрочный интервал – множитель, значение – 1,2;

1. комментарии (обязательно должны присутствовать в тексте программы) выделить курсивом и синим цветом.

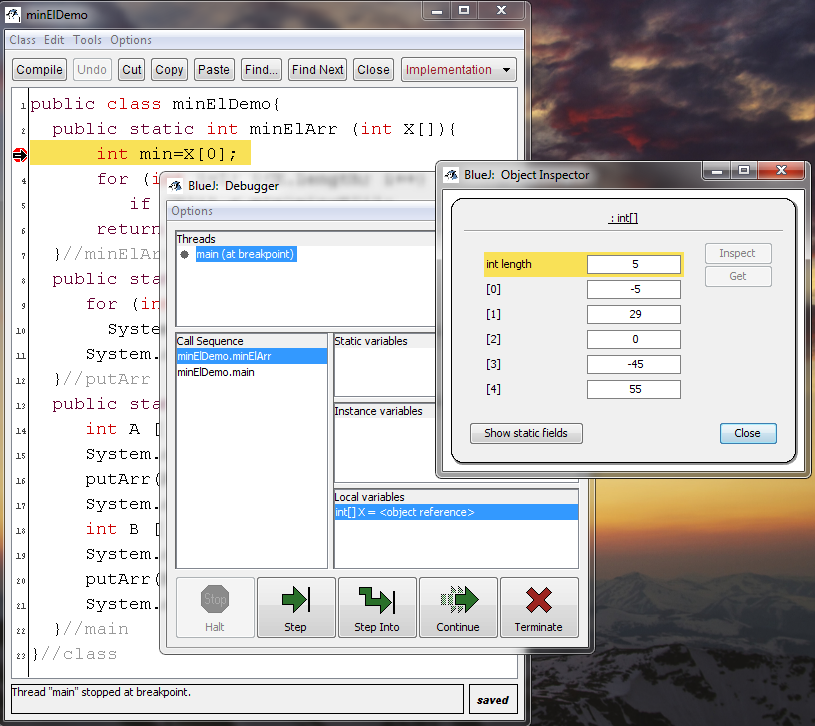
**7. СВЕДЕНИЯ ОБ ОТЛАДКЕ ПРОГРАММЫ И ПРОВЕРКЕ**

**ЕЕ РАБОТОСПОСОБНОСТИ**

Проверка работы программы производилась в пошаговом (с помощью отладчика BlueJ и в непрерывном режиме).

На рисунке 7.1 изображены следующие окна: окноредактора с текстом класса, окно отладчика после запуска метода main() и окно инспектора объектов, в котором визуализированы элементы массива.

Результат первого шага (выполнения команды в точке останова после нажатия на кнопку «Step») представлен на рисунке 7.2.



***Привести результаты для своей программы, точку останова установить на любом операторе присваивания в method1***

Рисунок 7.1 – Работа с отладчиком: просмотр

элементов массива, переданного в метод

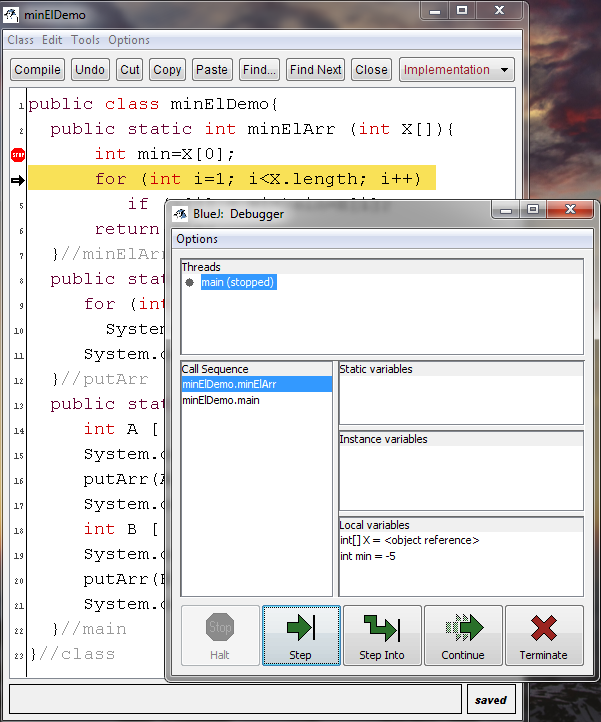


Рисунок 2.7 – Пошаговое выполнение программы

с точки останова – шаг 1

Результаты выполнения программы в непрерывном режиме представлены на рисунке 7.3.

***Скриншот окна терминала (не мельчить!)***

Рисунок 7.3 – Результаты выполнения программы

На всех тестовых примерах программа выдала ожидаемый результат, что позволяет сделать вывод о ее работоспособности.

**ВЫВОД**

В ходе выполнения лабораторной работы были изучены принципы структурирования данных в виде одномерных массивов, принципы обработки массивов с помощью использования операторов цикла, принципы структурирования программ, правила передачи параметров в метод и возвращения методом значения, а также разработана программа, вызывающая различные методы обработки одномерных массивов, в том числе, простой метод сортировки массива (метод пузырька).

***Уважаемые студенты! Обращаю ваше внимание на то, что шаблоны отчетов даются вам для того, чтобы вы научились писать технические отчеты и правильно составлять документацию к программам. Поэтому настоятельно прошу сначала внимательно прочитать текст шаблона, при необходимости творчески переработать его в соответствии с данными своего варианта и только затем вставлять результаты работы своей программы.***