**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**ВЫСШАЯ ШКОЛА ПЕЧАТИ И МЕДИАИНДУСТРИИ**

***Институт Принтмедиа и информационных технологий***

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**ЛАБОРАТОРНАЯ РАБОТА № 4-5**

**Дисциплина:** Основы алгоритмизации и программирования.

**Тема:** Алгоритм сортировки «пузырёк»

**Выполнила:**

**студентка группы 201-723**

Круглова А.М.

19.10.20  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Проверил:** преп. Хуснулина Д.Р.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Москва**

**2020**

**Оглавление**

[Цель: 3](#_Toc54138562)

[Постановка задачи: 3](#_Toc54138563)

[Идея алгоритма: 3](#_Toc54138564)

[Словесное представление алгоритма: 4](#_Toc54138565)

[Блок-схема с использованием элемента модификации: 5](#_Toc54138566)

[Листинг программы с использованием параметрического цикла: 6](#_Toc54138567)

[Результат работы программы: 7](#_Toc54138568)

[Блок-схема без использования элемента модификации: 8](#_Toc54138569)

[Листинг программы с использованием цикла с предусловием: 9](#_Toc54138570)

[Результат работы программы: 10](#_Toc54138571)

Цель:

Получить практические навыки разработке алгоритмов и их программной

реализации.

Постановка задачи:

Необходимо выполнить и оформить описание следующих пунктов:

1. Сформулировать идею алгоритма
2. Выполнить словесное представление алгоритма
3. Выполнить полнить представление алгоритма с помощью блок схем с использованием элемента модификации и без него.
4. Выполнить программную реализацию алгоритмов на языке С с использованием параметрического цикла и цикла с предусловием.

Идея алгоритма:

Алгоритм основан на повторяющихся проходах по сортируемому массиву. За

каждый проход последовательно сравниваются соседние элементы. Если порядок

в паре неверный, то происходит обмен значений элементов. При каждом проходе

алгоритма по внутреннему циклу, очередной наибольший элемент массива

ставится на своё место в конце массива рядом с предыдущим «наибольшим

элементом», а наименьший элемент перемещается на одну позицию к началу

массива («всплывает» до нужной позиции, как пузырёк в воде — отсюда и

название алгоритма).

Проходы по массиву повторяются n-1 (где n – размер массива) раз или до тех пор,

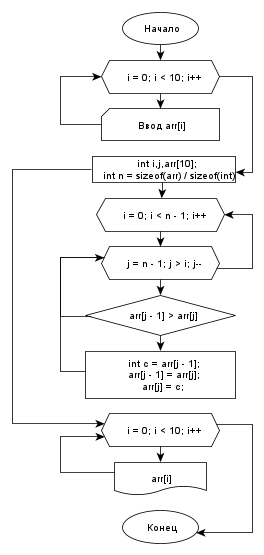
пока не будет перестановок, т.е. когда массив окажется отсортированным.

Словесное представление алгоритма:

arr – массив, n – длина массива

1. Начало алгоритма
2. Параметр i=0
3. Если i<10, то п.4, иначе п.6
4. Ввод с клавиатуры arr[i]
5. i++, п.3
6. Параметр внешнего цикла i = 0
7. Если i < n – 1, то п.8, иначе п.14
8. Параметр внутреннего цикла j = n-1
9. Если j > i, то п.10, иначе п.13
10. Если arr[j-1] > arr[j], то п.11, иначе п.12
11. Обмен значениями arr[j-1] и arr[j]
12. j--, п.4
13. i++, п.2
14. Параметр i=0
15. Если i<10, то п.16, иначе п.18
16. Вывод на экран arr[i]
17. i++, п.15
18. Конец алгоритма

Блок-схема с использованием элемента модификации:



Нет

Да

Рисунок 1 — Блок-схема 1 программы

Листинг программы с использованием параметрического цикла:

Листинг 1 — Исходный код 1 программы

#include <iostream>

#include <stdlib.h>

int main()

{

setlocale(LC\_ALL, "Russian"); //установка русского языка

int i, j;

int arr[10]; // Объявляем массив из 10 элементов

for (i = 0; i < 10; i++) // Вводим значения элементов массива через клавиатуру

{

printf("[%d] = ", i);

scanf\_s("%d", &arr[i]);

}

int n = sizeof(arr) / sizeof(int); //размер массива

for (i = 0; i < n - 1; i++) //начинаем внешний цикл

{

for (j = n - 1; j > i; j--) //внутренний цикл

{

if (arr[j - 1] > arr[j]) // условие, если текущий элемент меньше предыдущего

{

int c = arr[j - 1]; // меняем их местами

arr[j - 1] = arr[j];

arr[j] = c;

}

}

}

// Выводим отсортированные элементы массива

printf("Отсортированный массив:\n");

for (i = 0; i < 10; i++)

printf("%d ", arr[i]);

printf("\n");

system("pause");

return 0;

}

Результат работы программы:

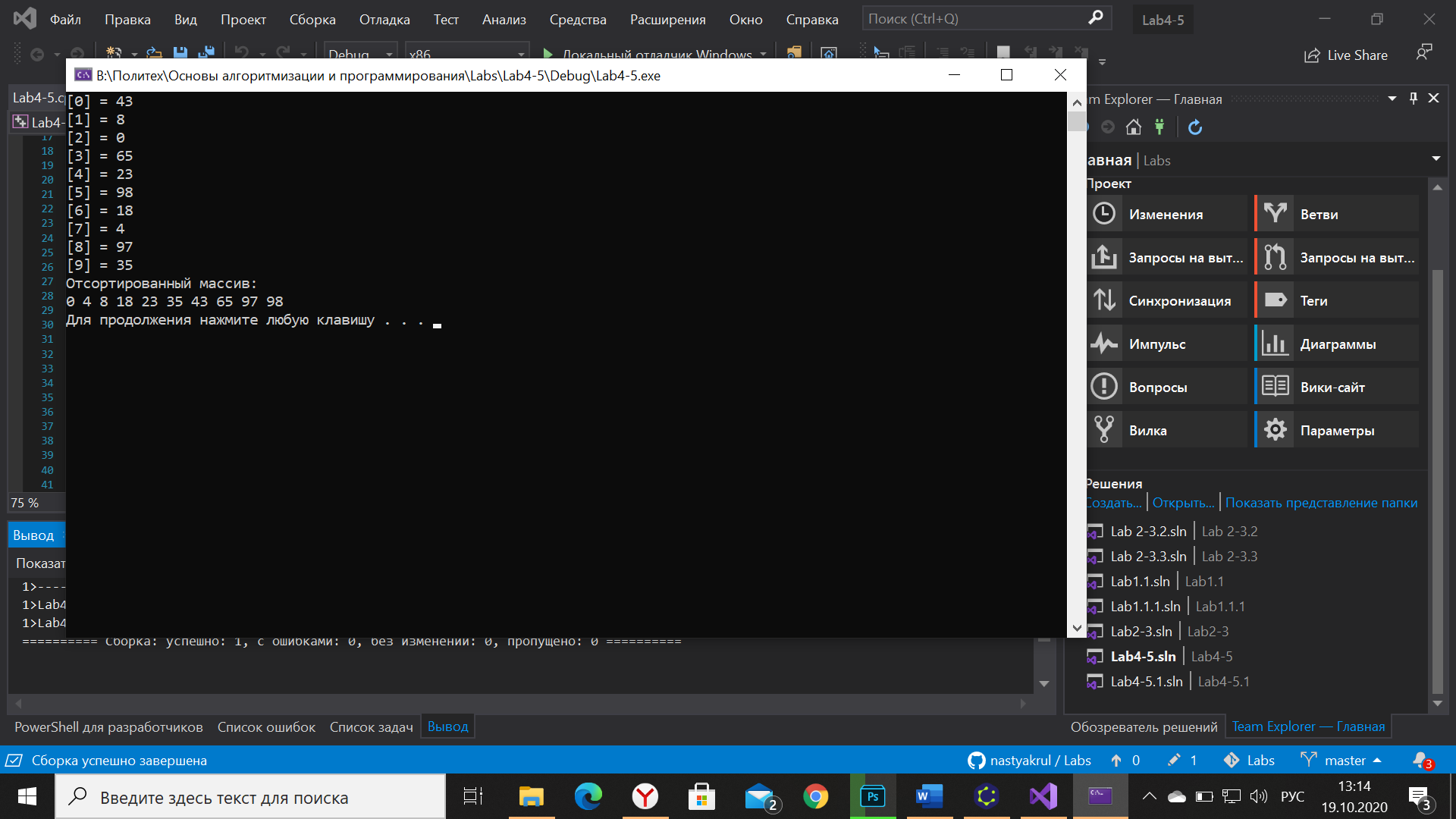
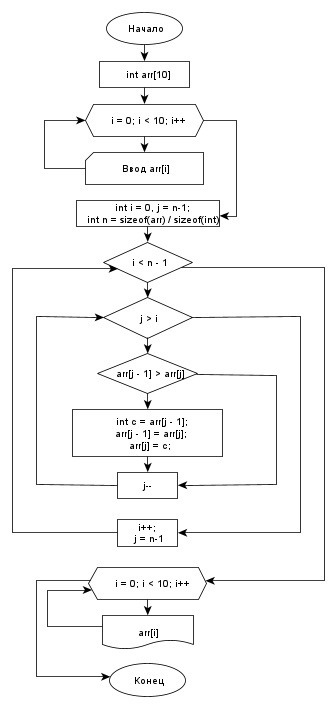


Рисунок 2 — Результат работы 1 программы

Блок-схема без использования элемента модификации:



Нет

Нет

Нет

Да

Да

Да

Рисунок 3 — Блок-схема 2 программы

Листинг программы с использованием цикла с предусловием:

Листинг 2 — Исходный код 2 программы

#include <iostream>

#include <stdlib.h>

int main()

{

setlocale(LC\_ALL, "Russian"); //установка русского языка

int arr[10]; // Объявляем массив из 10 элементов

int n = sizeof(arr) / sizeof(int); //размер массива

for (int i = 0; i < 10; i++) // Вводим значения элементов массива через клавиатуру

{

printf("[%d] = ", i);

scanf\_s("%d", &arr[i]);

}

int i = 0, j = n-1; //начальное значение

while (i < n - 1) //внешний цикл

{

while (j > i) //внутренний цикл

{

if (arr[j - 1] > arr[j]) // условие, если текущий элемент меньше предыдущего

{

int c = arr[j - 1]; // меняем их местами

arr[j - 1] = arr[j];

arr[j] = c;

}

j--;

}

i++;

j = n-1;

}

// Выводим отсортированные элементы массива

printf("Отсортированный массив:\n");

for (i = 0; i < 10; i++)

printf("%d ", arr[i]);

printf("\n");

system("pause");

return 0;

}

Результат работы программы:

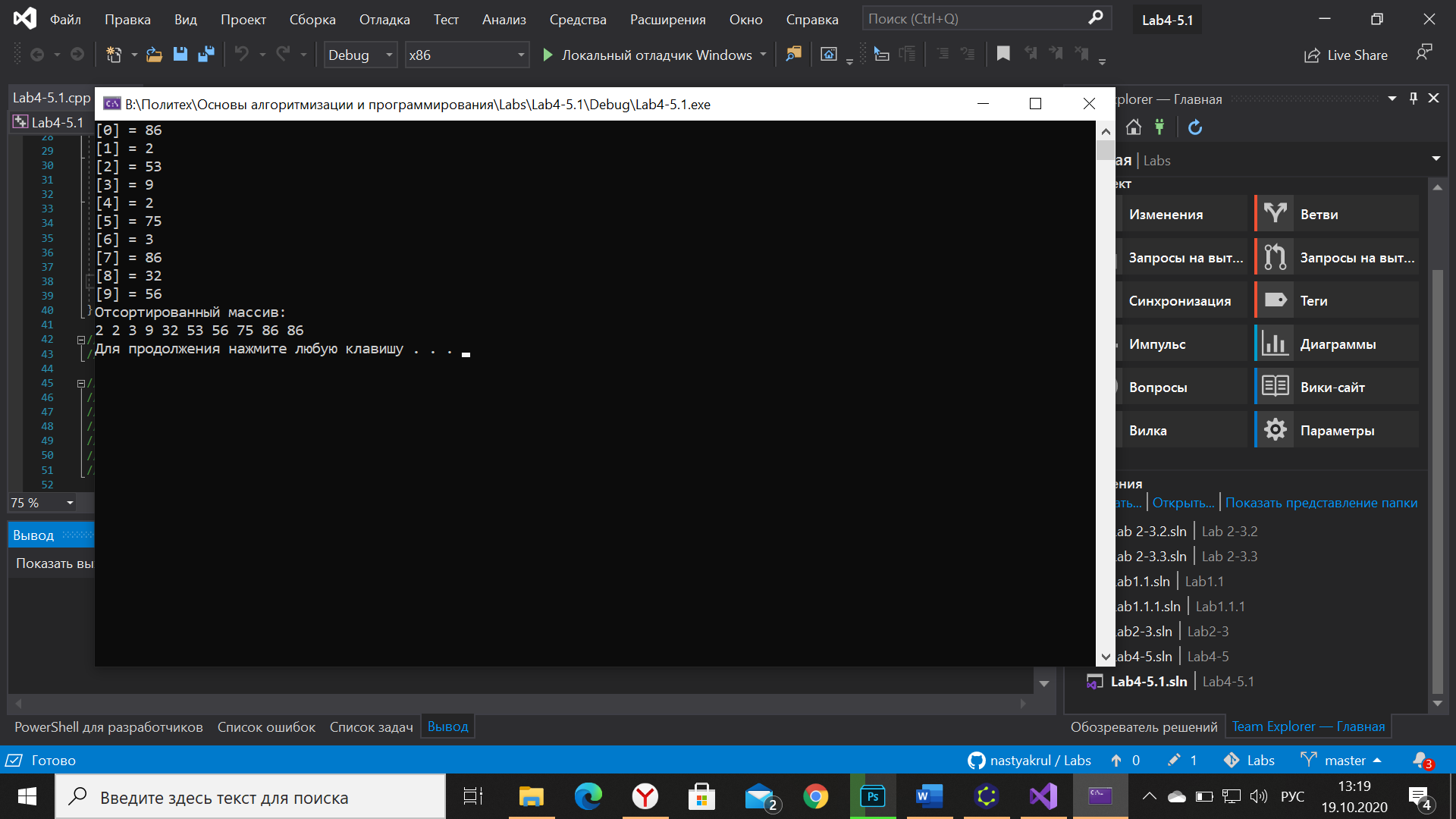


Рисунок 4 — Результат работы 2 программы