**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 № 14-15**

**Дисциплина:** Основы алгоритмизации и программирования.

**Тема:** Алгоритм сортировки «гномья»

**Выполнила:**

**студентка группы 201-723**

Круглова А.М.

06.12.20  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Проверил:** преп. Хуснулина Д.Р.

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

(Дата) (Подпись)

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

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

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

**Москва**

**2020**

**Оглавление**

[Цель 3](#_Toc58178567)

[Постановка задачи 3](#_Toc58178568)

[Идея алгоритма 3](#_Toc58178569)

[Словесное представление алгоритма 4](#_Toc58178570)

[Листинг программы с использованием параметрического цикла 6](#_Toc58178571)

[Результат работы 1 программы 7](#_Toc58178572)

[Блок-схема без использования элемента модификации 8](#_Toc58178573)

[Листинг программы с использованием цикла с предусловием 9](#_Toc58178574)

[Результат работы 2 программы 10](#_Toc58178575)

# **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**

Гномья сортировка (англ. Gnome sort) — алгоритм сортировки, похожий на сортировку

вставками, но в отличие от последней перед вставкой на нужное место происходит серия

обменов, как в сортировке пузырьком. Название происходит от предполагаемого

поведения садовых гномов при сортировке линии садовых горшков.

Алгоритм концептуально простой, не требует вложенных циклов. Время работы O(n²). На

практике алгоритм может работать так же быстро, как и сортировка вставками.

Идея алгоритма очень проста. Пусть имеется массив A размером N, тогда

сортировка выбором сводится к следующему:

* Смотрим на текущий и предыдущий элемент массива:
* если они в правильном порядке, шагаем на один элемент вперед,
* иначе меняем их местами и шагаем на один элемент назад.
* Граничные условия:
* если нет предыдущего элемента, шагаем вперёд;
* если нет следующего элемента, стоп.

Это оптимизированная версия с использованием переменной j, чтобы разрешить прыжок

вперёд туда, где он остановился до движения влево, избегая лишних итераций и

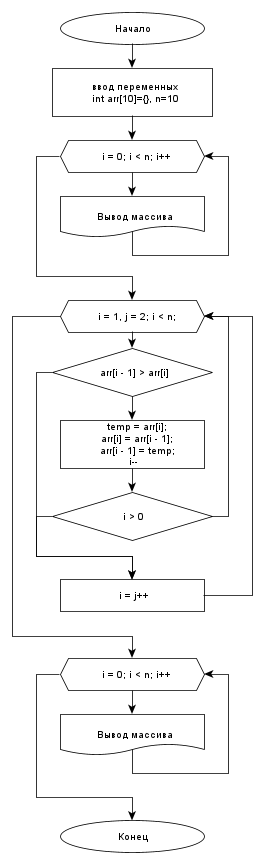
сравнений.

Словесное представление алгоритма

Arr – массив, N – длина массива, i,j – индексы массивов, min – индекс локального минимума

1. Сортировка начинается со второго и третьего элемента i=1, j=2
2. Если i < N, то к пункту 3, иначе к пункту 9
3. Если arr[i-1] > arr[i], то к пункту 4, иначе к пункту 7
4. Меняем местами значения arr[i] и arr[i-1]
5. Шагаем на один элемент назад i—
6. Если i>0, то к пункту 2(используя оператор continue), иначе к пункту 7
7. I=j++
8. К пункту 2
9. Конец алгоритма

Блок-схема с использованием элемента модификации



Нет

Да

Да

Нет

Рисунок 1 — Блок-схема 1 программы

Листинг программы с использованием параметрического цикла

Листинг 1 — Исходный код 1 программы

#include <iostream>

#include <stdlib.h>

#include <locale.h>

int main()

{

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

int i, j, temp, n = 10; //ввод переменных

int arr[10] = { 56, 12, 9, 88, 2, 74, 166, 35, 90, 1 }; // Объявляем массив из 10 элементов

for (i = 0; i < n; i++) //вывод исходного массива на экран

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

printf("\n");

for (i = 1, j = 2; i < n;) //ввод цикла

{

if (arr[i - 1] > arr[i]) //ввод условия, если предыдущий элемент больше взятого

{

temp = arr[i]; //обмен значениями

arr[i] = arr[i - 1];

arr[i - 1] = temp;

i--; //уменьшение переменной

if (i > 0) continue; //если это не конец массива, то продолжаем

}

i = j++; //присвоение ового значения

}

printf("Отсортированный массив:\n");

for (i = 0; i < n; i++) //вывод преобразованного массива на экран

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

printf("\n");

return 0;

}

Результат работы 1 программы

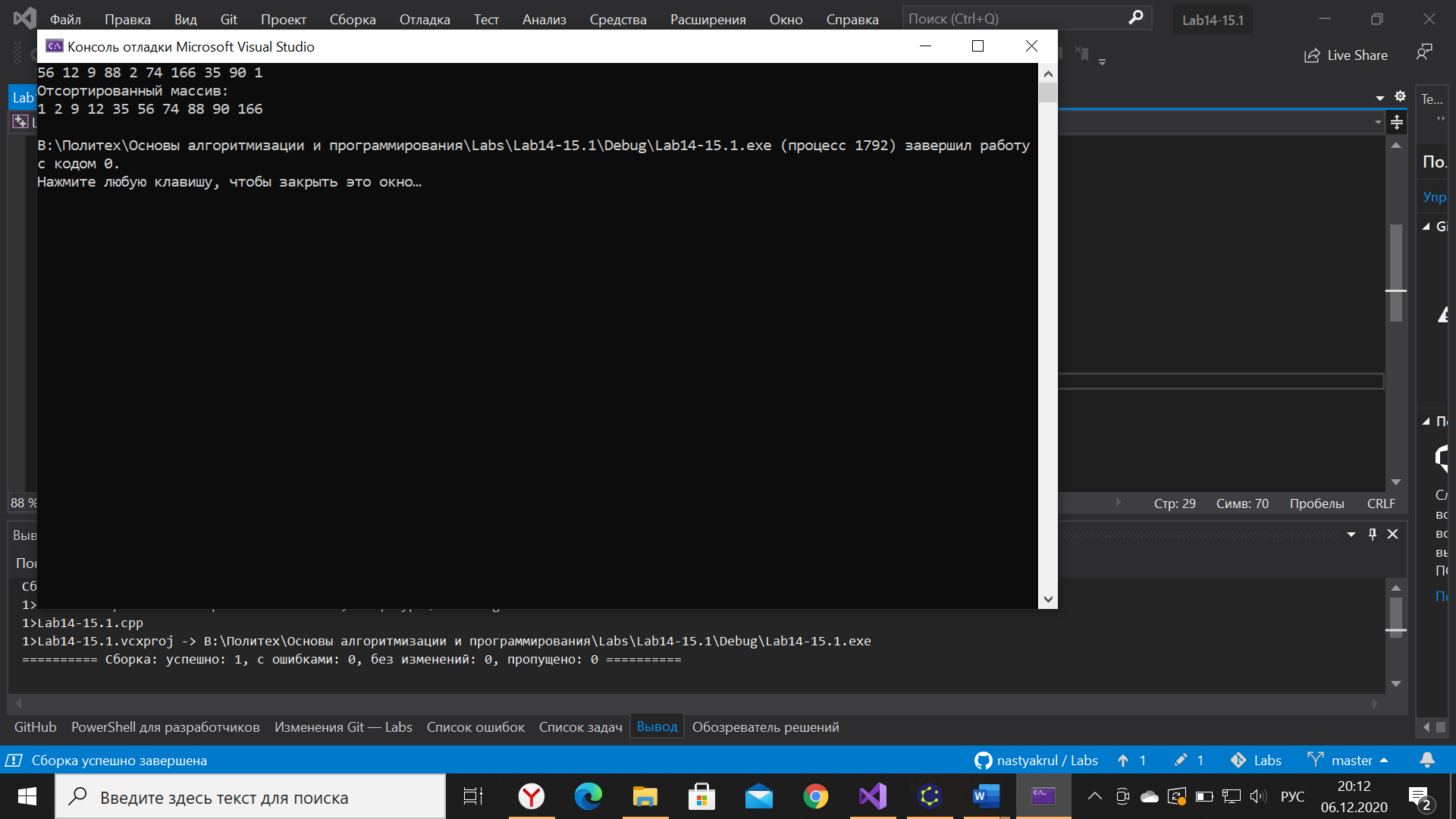
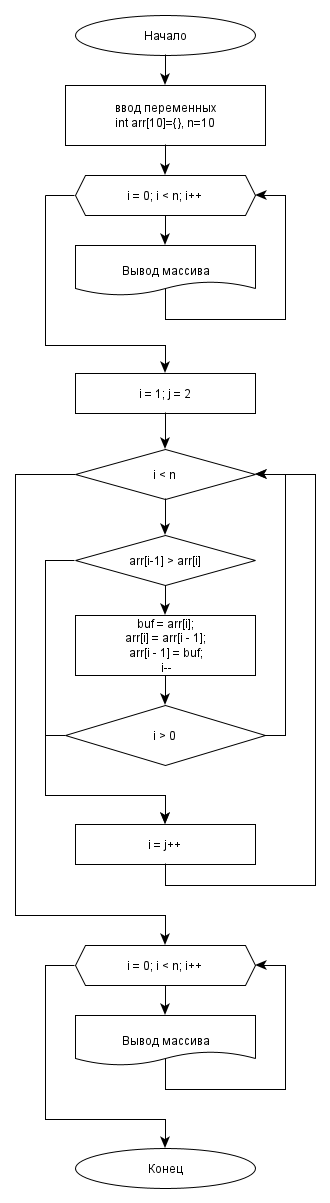


Рисунок 2 — Результат работы 1 программы

Блок-схема без использования элемента модификации



Да

Нет

Нет

Нет

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>

#include <locale.h>

int main()

{

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

int i, j, temp, n = 10; //ввод переменных

int arr[10] = { 56, 12, 9, 88, 2, 74, 166, 35, 90, 1 }; // Объявляем массив из 10 элементов

for (i = 0; i < n; i++) //вывод исходного массива на экран

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

printf("\n");

for (i = 1, j = 2; i < n;) //ввод цикла

{

if (arr[i - 1] > arr[i]) //ввод условия, если предыдущий элемент больше взятого

{

temp = arr[i]; //обмен значениями

arr[i] = arr[i - 1];

arr[i - 1] = temp;

i--; //уменьшение переменной

if (i > 0) continue; //если это не конец массива, то продолжаем

}

i = j++; //присвоение ового значения

}

printf("Отсортированный массив:\n");

for (i = 0; i < n; i++) //вывод преобразованного массива на экран

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

printf("\n");

return 0;

}

Результат работы 2 программы

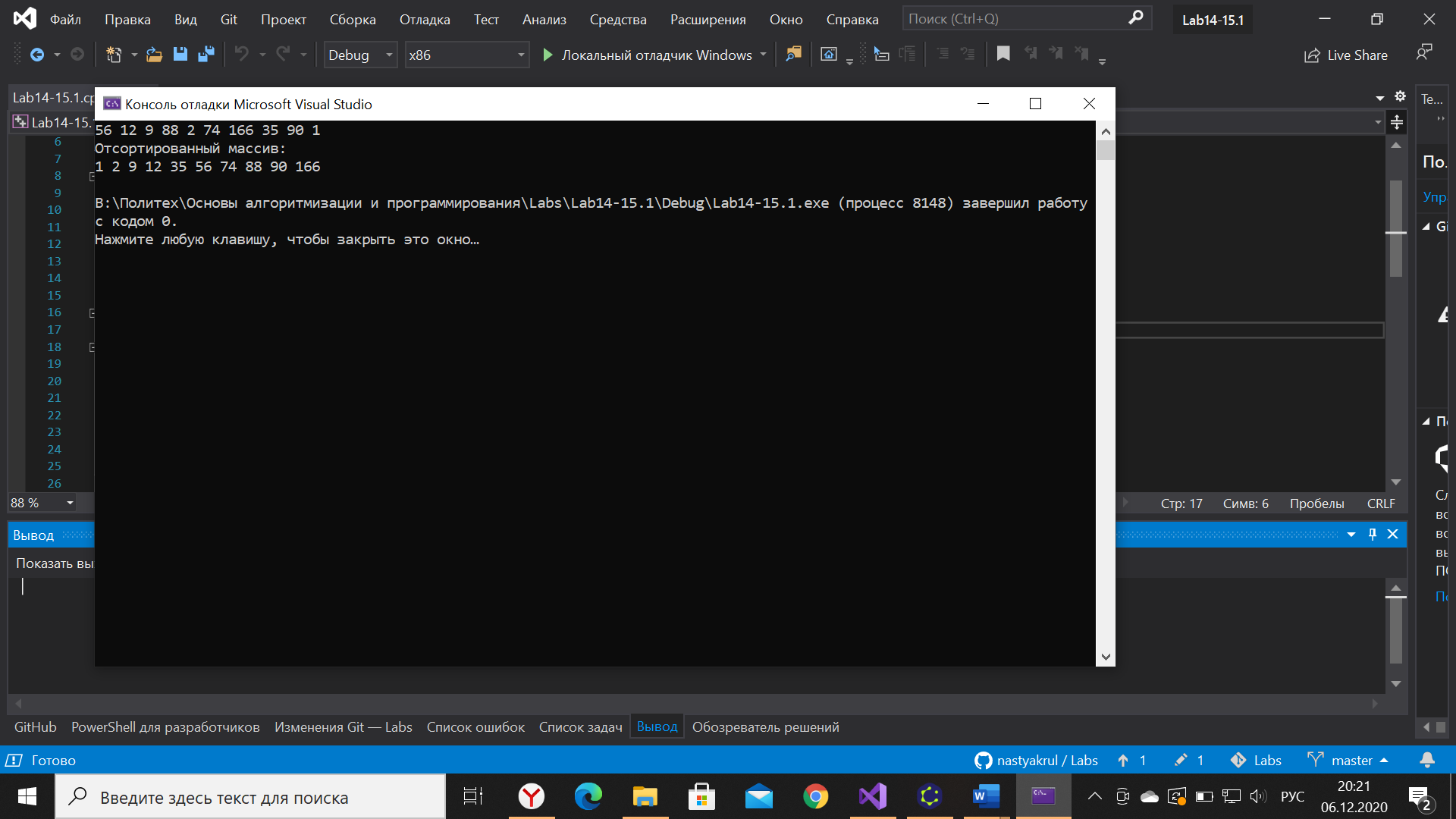


Рисунок 4 — Результат работы 2 программы