

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий
Кафедра вычислительной техники

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ № 10

Сортировка массивов
Вариант № 19

Преподаватель

подпись, дата

Пушкарев К. В.

Студент КИ18-096, 031831293

подпись

18.11.2018

Овсянников В.А.

Красноярск 2018

1 Дополнительное упражнение № 3 (вариант 9)

Разработать, отладить, продемонстрировать и защитить преподавателю графическую схему алгоритма и составить программу, решающую следующую задачу: на основе исходных целочисленных массивов $X[n]$ и $Y[m]$ сформировать массив C по правилам, заданным в таблице №10.3, Правила формирования массива C : Из расстояний от начала координат до точек координаты которых являются элементами массивов X и Y : $(x_1, y_1), \dots, (x_n, y_n) (n \leq 30, m = n)$; Дополнительные указания: Найти и напечатать координаты точки, для которой угол между осью абсцисс и лучом, соединяющим начало координат с точкой, минимальный (считать, что такая точка одна). Полученный массив отсортировать по невозрастанию. Провести анализ разработанной программы согласно критериям качества.

2 Цель работы

Освоение основных приемов обработки массивов, методов доступа к элементам массивов, их реорганизации и модификации.

3 Графическая схема алгоритма

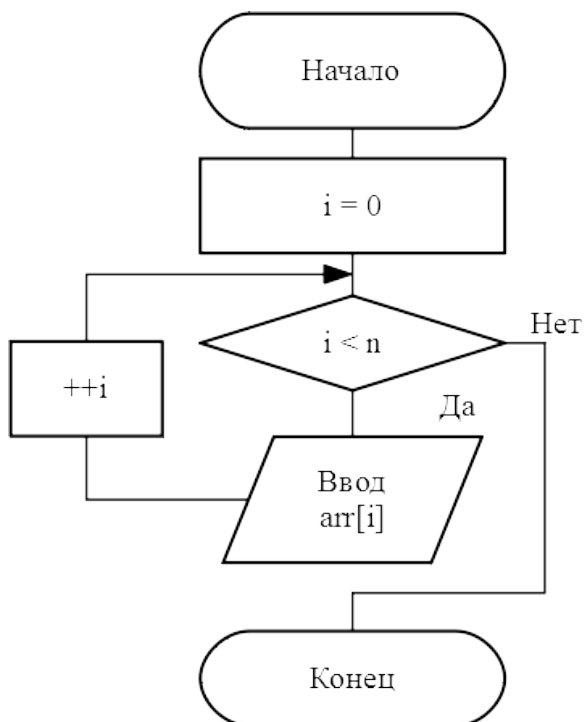


Рисунок 1 – Функция *void input_array(float * arr, int n).*

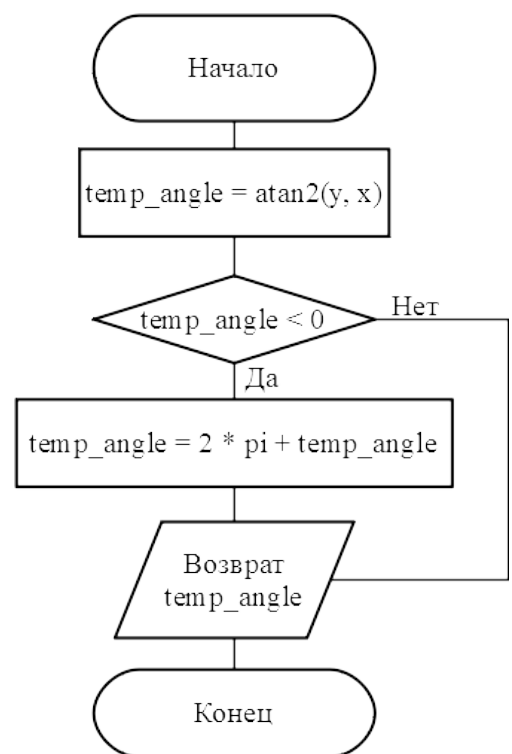


Рисунок 2 – Функция *float angle(float y, float x).*

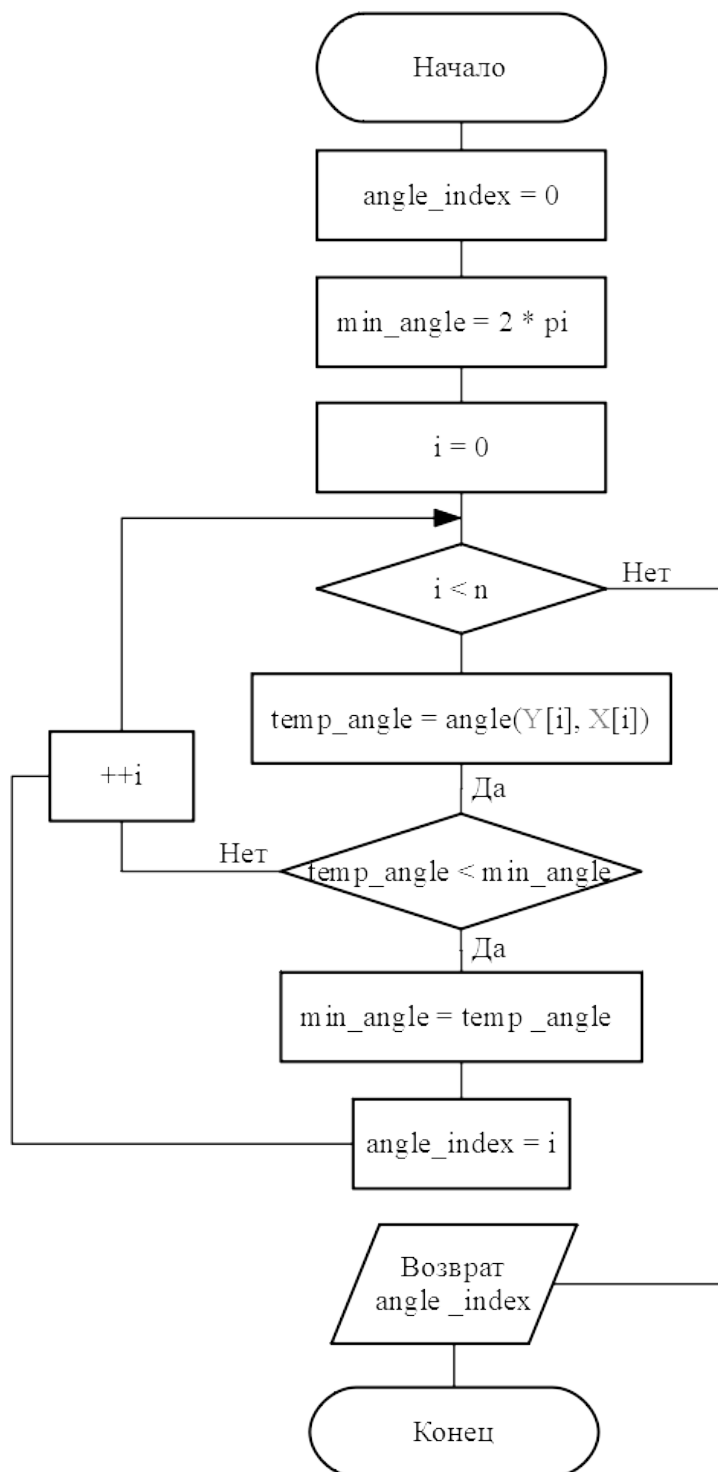


Рисунок 3 – функция
*float min_angle_index(const float * X, const float * Y, int n)*

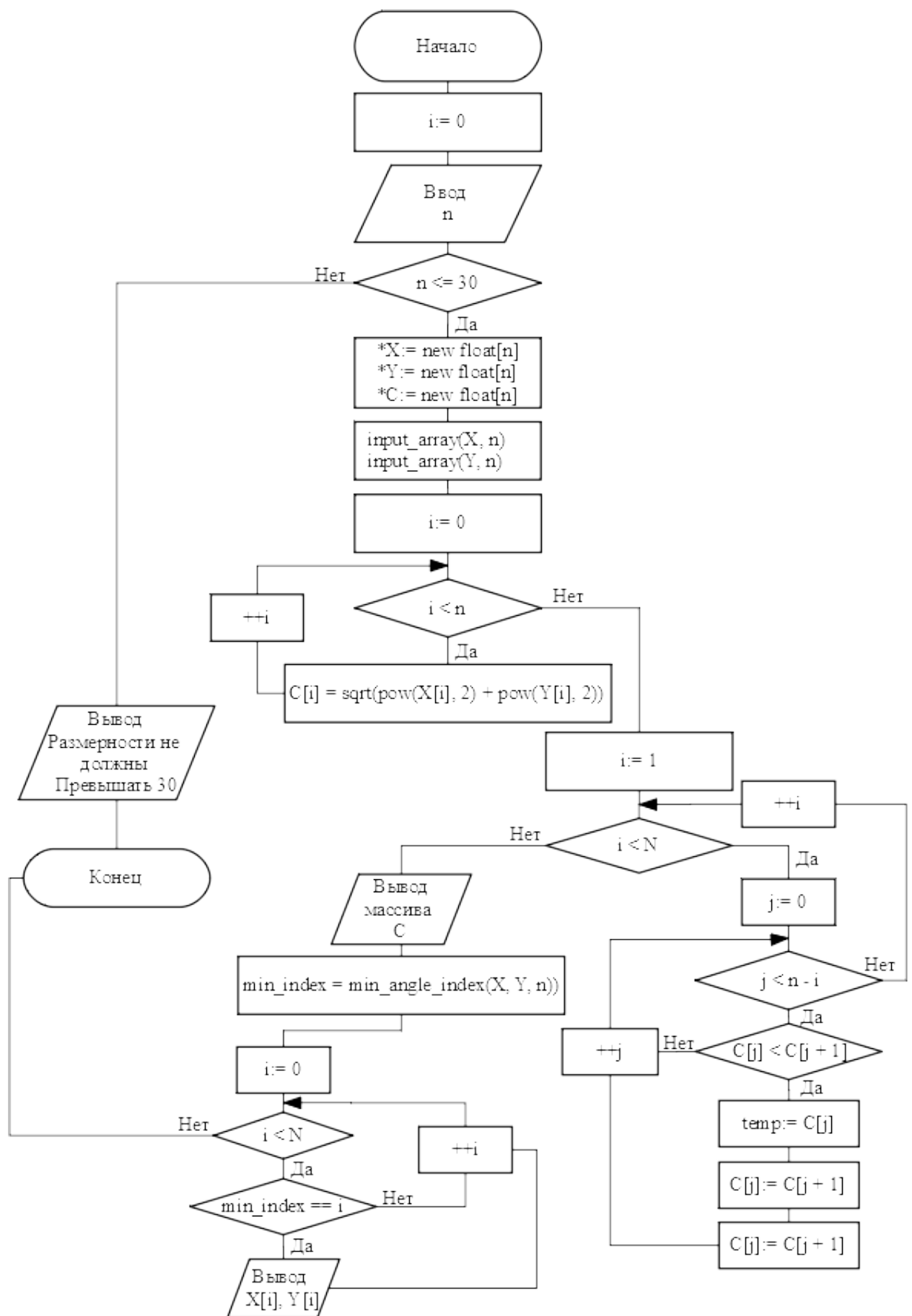


Рисунок 4 – Функция *main*

4 Код программы

```
1 #include "pch.h"
2 #include <iostream>
3 #include <locale>
4 #include <math.h>
5
6 using namespace std;
7
8 void input_array(float *arr, int n);
9 float min_angle_index(const float *X, const float *Y, int n);
10 float angle(float y, float x);
11
12 static const double pi = acos(-1);
13
14 int main(){
15     setlocale(LC_ALL, "rus");
16
17     int n, i = 0;
18
19     cout << "Введите размерность массива n: ";
20     cin >> n;
21
22     if (n <= 30) {
23         float *X = new float[n];
24         float *Y = new float[n];
25         float *C = new float[n];
26
27         cout << "Введите массив X: ";
28         input_array(X, n);
29
30         cout << "Введите массив Y: ";
31         input_array(Y, n);
32
33         for (int i = 0; i < n; i++) {
34             C[i] = sqrt(pow(X[i], 2) + pow(Y[i], 2));
35         }
36
37         for (int i = 1; i < n; i++) {
38             for (int j = 0; j < n - i; j++) {
39                 if (C[j] < C[j + 1]) {
40                     float temp = C[j];
41                     C[j] = C[j + 1];
42                     C[j + 1] = temp;
43                 }
44             }
45         }
46         cout << "\nМассив C = {";
47         for (int i = 0; i < n; i++) {
48             cout << " " << C[i];
49         }
50         cout << " }" << endl;
51
52         float min_index = min_angle_index(X, Y, n);
53
54         for (int i = 0; i < n; i++) {
55             if (min_index == i) {
56                 cout << "Точка для которой угол минимален: (" << X[i] << ", "
57 << Y[i] << ")" << endl;
58             }
59         }
60     }
```

```

61     else {
62         cout << "\nРазмерности не должны превышать 30" << endl;
63     }
64     return 0;
65 }
66 void input_array(float *arr, int n) {
67     for (int i = 0; i < n; i++) {
68         cin >> arr[i];
69     }
70 }
71 float angle(float y, float x) {
72     float temp_angle = atan2(y, x);
73     if (temp_angle < 0) {
74         temp_angle = 2 * pi + temp_angle;
75     }
76     return temp_angle;
77 }
78 float min_angle_index(const float *X, const float *Y, int n) {
79     int angle_index = 0;
80     double min_angle = 2 * pi, temp_angle;
81     for (int i = 0; i < n; i++) {
82         temp_angle = angle(Y[i], X[i]);
83         if (temp_angle < min_angle) {
84             min_angle = temp_angle;
85             angle_index = i;
86         }
87     }
88     return angle_index;
89 }

```

6 Результаты выполнения экспериментальной части работы

Результаты запуска программы с различными входными значениями приведены в таблице 1.

Таблица 1 – Работа программы на тестовых наборах данных

Входные данные	Результат
$n = 5$ $A = \{1 \ -1 \ 1 \ 100 \ 10\}$ $B = \{-1 \ 1 \ 1 \ 10 \ -10\}$	Массив $C = \{100.499 \ 14.1421 \ 1.41421 \ 1.41421 \ 1.41421\}$ Точка для которой угол минимален: (100, 10)
$n = 3$ $A = \{1 \ 3 \ 5 \ 8\}$ $B = \{-12 \ 2 \ -8 \ 0\}$	Массив $C = \{4.24264 \ 2.82843 \ 1.41421\}$ Точка для которой угол минимален: (1, -1)
$n = 4$ $A = \{1 \ 2 \ 3\}$ $B = \{-1 \ -2 \ -3\}$	Массив $C = \{12.0416 \ 9.43398 \ 8 \ 3.60555\}$ Точка для которой угол минимален: (8, 0)