**ЛАБОРАТОРНАЯ РАБОТА №4**

**«ПРОГРАММИРОВАНИЕ АЛГОРИТМОВ ОБРАБОТКИ ОДНОМЕРНЫХ СТАТИЧЕСКИХ МАССИВОВ»**

**1. Цель работы**

Изучить способы представления массивов в памяти ЭВМ, получить практические навыки реализации алгоритмов обработки одномерных массивов.

**2. Вариант задания**

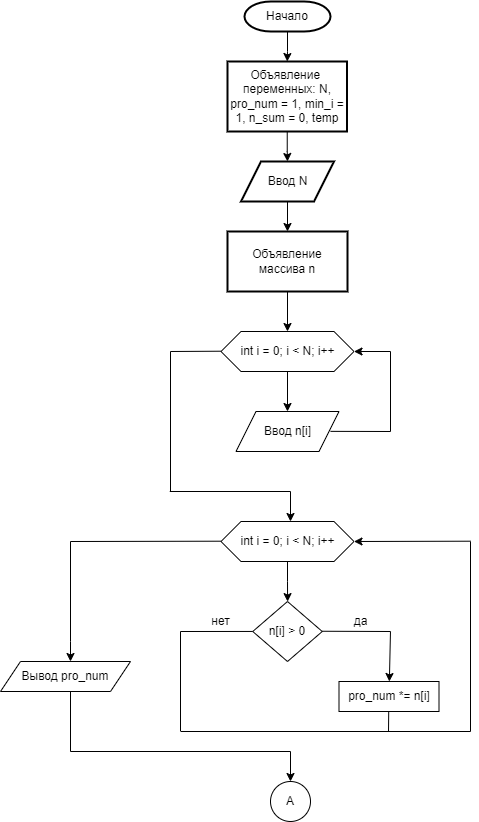
В одномерном массиве, состоящем из N вещественных элементов, вычислить:

1) произведение положительных элементов массива;

2) сумму элементов массива, расположенных до минимального элемента. Упорядочить по возрастанию отдельно элементы, стоящие на четных местах, и элементы, стоящие на нечетных местах, методом вставки.

**3. Ход работы**

На рисунке 1 представлена структурная схема программы на С++.



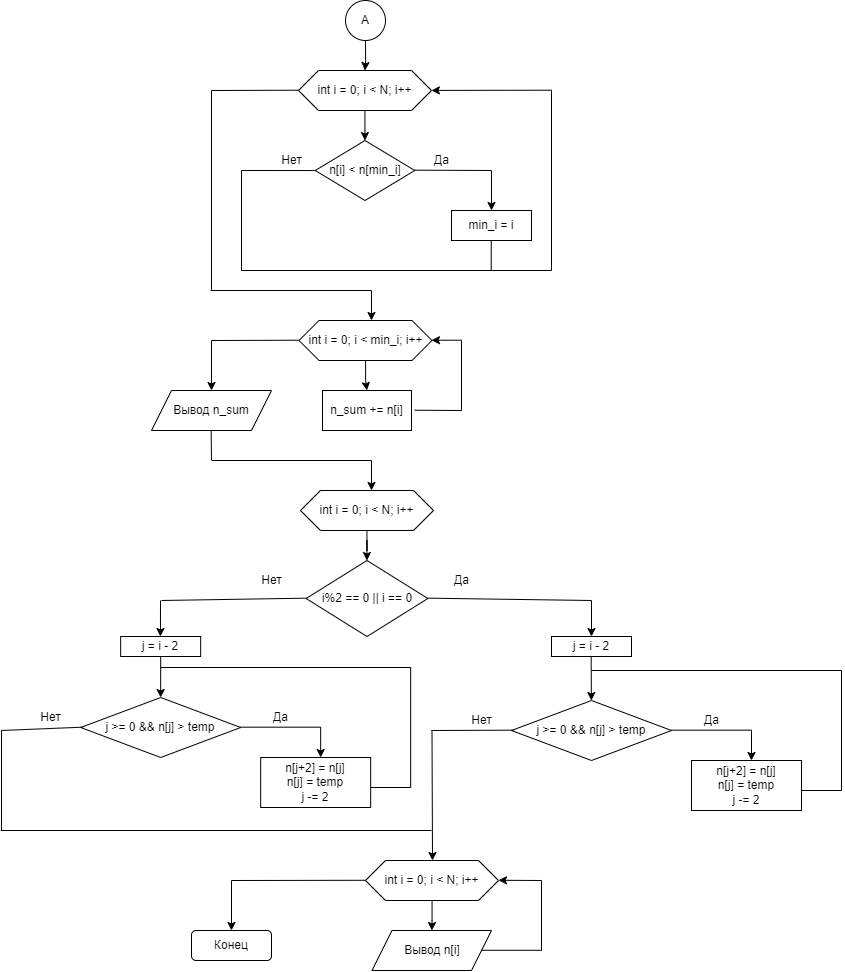


Рисунок 1 – Структурная схема работы программы.

На основе структурной-схемы была написана программа на языке C++.

В листинге 1 представлен код написанной программы.

Листинг 1 – Исходный код программы на C++.

#include <iostream>

#include <windows.h>

using namespace std;

int main() {

SetConsoleOutputCP(65001);

int N;

float pro\_num = 1, n\_sum = 0, temp;

int j, min\_i = 1;

cout << "input len list: "; cin >> N;

float n[N];

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

cin >> n[i];

}

// Произведение всех элементов массива больше нуля

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

if (n[i] > 0) pro\_num \*= n[i];

}

cout << "1)product of numbers: " << pro\_num << endl;

// Сумма всех элементов массива до его минимального элемента

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

if (n[i] < n[min\_i]) min\_i = i;

}

for (int i = 0; i < min\_i; i++) {

n\_sum += n[i];

}

cout << "2)sum of all elements up to 'min': " << n\_sum << endl;

// Сортировка отдельно стоящих элементов на чётных и нечётных позициях методом вставки

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

temp = n[i];

if (i%2 == 0 || i == 0) {

j = i - 2;

while (j >= 0 && n[j] > temp) {

n[j+2] = n[j];

n[j] = temp;

j -= 2;

}

} else {

j = i - 2;

while (j >= 0 && n[j] > temp) {

n[j+2] = n[j];

n[j] = temp;

j -= 2;

}

}

}

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

cout << n[i] << " ";

}

return 0;

}

Далее было проведено тестирование программы. Били взяты три различных набора данных для тестирования программы(Рисунки 2-4).

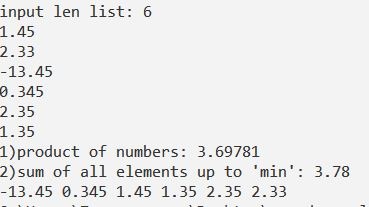


Рисунок 2 – Тестирование программы на C++

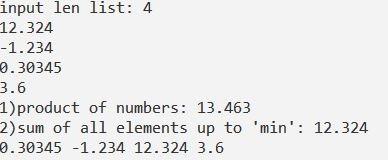


Рисунок 2 – Тестирование программы на C++

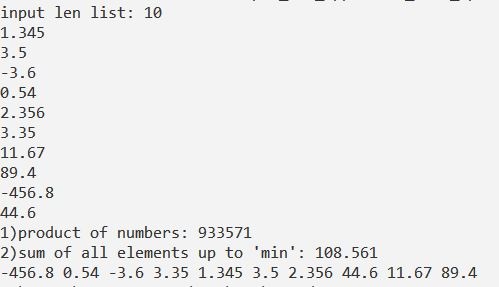


Рисунок 2 – Тестирование программы на C++

**Выводы**

В ходе лабораторной работы были изучены способы представления массивов в памяти ЭВМ, одномерные массивы в языках C\C++ и несколько алгоритмов сортировки.