**Отчет**

**по Лабораторной работе № 5**

**«Работа с функциями»**

Специальность 09.02.03«Программирование в компьютерных системах»

ПМ.01 «Разработка программных модулей и программного обеспечения для компьютерных систем»

МДК.01.02 «Прикладное программирование»

**Министерство образования и науки Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,**

**МЕХАНИКИ И ОПТИКИ”**

**ФАКУЛЬТЕТ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ**

Выполнил:

студент группы 2337

Коряков С.А.

Санкт-Петербург

2018/2019

Преподаватель:

ФИО Сорокин Д. Ю.

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_201\_\_г.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Цель работы

Продемонстрировать работу с функциями и способами передачи параметров между ними.

Практическое задание

Выполнить вторую лабораторную работу, разделив программу на функции:

1. Первая функция не возвращает значений (void), выполняет динамическое выделение памяти для элементов массива и инициализируются их с данными, введёнными с клавиатуры;

2. Вторая функция выполняет задание по варианту;

3. Третья функция печатает результат. Освобождение памяти происходит в главной функции.

Индивидуальное задание

8. Даны вещественные массивы A[15], W[15]. Для каждого массива определить количество элементов, предшествующих наибольшему элементу массива и их среднее геометрическое.

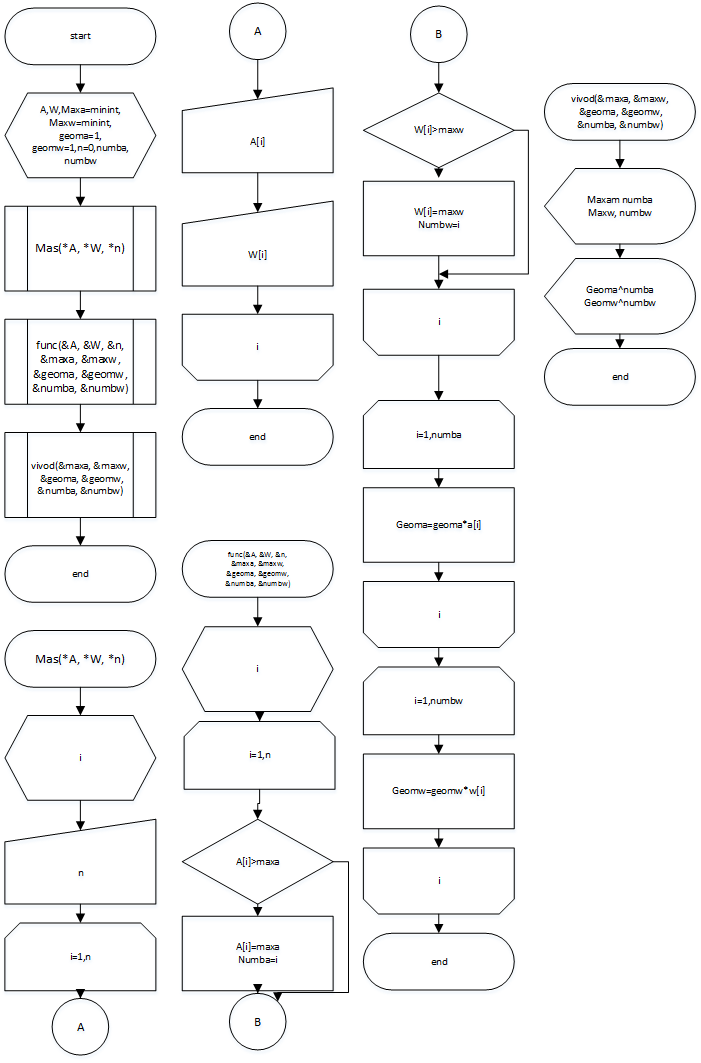
 Блок-схема алгоритма работы программы

Рисунок 1, алгоритм выполнения программы

Листинг

#include <iostream> // подключение основной библиотеки для работы основных функций

#include <math.h> // подключение библиотеки для математических вычислений

#include <limits> // подключение библиотеки для задания минимальных значений

using namespace std; // использование постоянного пространства имен для удобного пользования

void mas(float \*& A, float \*& W, int \*n);//прототип функции инициализации массива

void func(float \*maxa, float \*maxw, float \*geoma, float \*geomw, int \*n, float \*& A, float \*& W, int \*numba, int \*numbw);//прототип функции выполнения задания по варианту

void vivod(float \*maxa, float \*maxw, float \*geoma, float \*geomw, int \*numba, int \*numbw);//прототип функции вывода

void main()

{

float \*A, \*W, maxa = numeric\_limits<float>::min(), maxw = numeric\_limits<float>::min(), geoma = 1, geomw = 1; //создание двух указателей для массивов, переменных для нахождения максимального значения и двух переменных для среднего геометрического

int n = 0, numba, numbw; // создание переменной, задающая длину массивов, и переменных хранящих позицию максимального элемента массивов

mas(\*&A, \*&W, &n); // вызов функции инициализации массива

func(&maxa, &maxw, &geoma, &geomw, &n, \*&A, \*&W, &numba, &numbw); //вызов функции выполнения задания по варианту

vivod(&maxa, &maxw, &geoma, &geomw, &numba, &numbw); //вызов функции вывода данных

delete[] A; //очищение памяти от массива А

delete[] W; //очищение памяти от массива W

}

void mas(float \*& A, float \*& W, int \*n)

{

int i = 0;

cout << "vvedite dlinu oboih massivov:";

cin >> \*n;

A = new float[\*n]; // создание массива А

W = new float[\*n]; // создание массива W

for (int i = 0; i < \*n; i++) // Цикл ввода масива А

{cout << "Please enter A[" << i << "] number "; // Сообщение о вводе i-того элемента

cin >> A[i]; // ввотого i-того элемента

}

for (int i = 0; i < \*n; i++) // Цикл ввода масива W

{cout << "Please enter W[" << i << "] number "; // Сообщение о вводе i-того элемента

cin >> W[i]; // ввотого i-того элемента

}

}

void func(float \*maxa, float \*maxw, float \*geoma, float \*geomw, int \*n, float \*& A, float \*& W, int \*numba, int \*numbw)

{for (int i = 0; i < \*n; i++) // цикл по поиску максимальных элементов с их адресами

{if (A[i] > \*maxa) //анализ переменной, по сравнению с предыдущей на то является ли она она больше

{\*maxa = A[i]; //запись наибольшее значение массива А на данный момент итерации

\*numba = i; // запись номера наибольшего значения массива А на данный момент итерации

}

if (W[i] > \*maxw) //анализ переменной, по сравнению с предыдущей на то является ли она она больше

{\*maxw = W[i]; //запись наибольшее значение массива W на данный момент итерации

\*numbw = i; // запись номера наибольшего значения массива W на данный момент итерации } }

for (int i = 0; i < \*numba; i++) // цикл для перемножение элементов до максимального в массиве А

{\*geoma = A[i] \* \*geoma; //умножение элемента под номером i с общим произведением элементов

}for (int i = 0; i < \*numbw; i++) // цикл для перемножение элементов до максимального в массиве W

{\*geomw = W[i] \* \*geomw; //умножение элемента под номером i с общим произведением элементов

}}

void vivod(float \*maxa, float \*maxw, float \*geoma, float \*geomw, int \*numba, int \*numbw)

{ cout << "Kol-vo el. v massive A do max. elementa :" << \*numba << endl; // вывод количества элементов до максимального в массива А

cout << "Kol-vo el. v massive W do max. elementa :" << \*numbw << endl; // вывод количества элементов до максимального в массива W

cout << "srednee geometricheskoe pervih " << \*numba << "elementov v Massive A = " << pow(\*geoma, float(1) / \*numba) << endl; //финальное вычисление и вывод среднего геометрического в массиве А

cout << "srednee geometricheskoe pervih " << \*numbw << "elementov v Massive W = " << pow(\*geomw, float(1) / \*numbw) << endl; //финальное вычисление и вывод среднего геометрического в массиве W

}}

Протокол выполнения



Рисунок 2, протокол выполнения программы