**Отчет**

**по Лабораторной работе № 6**

**«Работа с шаблонами функций»**

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 шаблонами функций.

Практическое задание

Выполнить третью лабораторную работу, разделив программу на шаблоны функций:

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[5][6], B[6]. Определить значения вещественного массива C[6] по правилу:

C[j] = сумме j-го столбца массива A при B[j]>0

C[j]=0 при B[j]=0

C[j]=A[0][j]+A[4][j] при B[j]<0

Значение переменной i изменяется от 0 до 5.

 Блок-схема алгоритма работы программы

Рисунок 1, алгоритм выполнения программы

Листинг

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

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

template <typename T, typename K >

void create(T \*\*&A, T \*&B, T \*&C, K &n, K &m);

template <typename T, typename K >

void vivod(T \*&C, K &m);

template <typename T, typename K >

void prikoli(T \*\*&A, T \*&B, T \*&C, K &n, K &m, T &sum);

void main() // главная область действия программы

{float \*\*A, \*B, \*C, sum = 0; //Создание указателей на один двумерного и на два одномерных массивов, а так же создание вспомогательной переменной

int n, m; //содание переменных, хранящих длинны массивов

create(A, B, C, n, m);

prikoli(A, B, C, n, m, sum);

vivod(C, m);

for (int i = 0; i < n; i++)//очищение из памяти строк массива А

{delete[] A[i];}

delete[] A;//очищение из памяти массива указателей А

delete[] B;//очищение из памяти массива В

delete[] C;//очищение из памяти массива С

}

template <typename T, typename K >

void create(T \*\*&A, T \*&B, T \*&C, K &n, K &m)

{int i, j;

cout << "vvedite kol-vo strok v Massive A ";

cin >> n; // ввод кол-ва строк в массиве А

cout << "vvedite kol-vo stolbcov v Massive A i lol-vo elementov v massive B i C ";

cin >> m; // ввод кол-ва столбцов в массиве А и ввод длины массивов B и C

A = new float \*[n]; //Создание массива укзателей на строки А

B = new float[m]; //Создание массива В

C = new float[m]; //Создание массива С

for (i = 0; i < n; i++) //Создание строк

{A[i] = new float[m]; //выделение памяти для каждой iтой строки

}

for (i = 0; i < n; i++) //два цикла на ввод Элементов массива А (цикл с итерацией по строкам)

{for (j = 0; j < m; j++) //цикл с итерацией по столбцам

{cout << "vvedite element " << i << " stolbca " << j << " stroki ";

cin >> A[i][j]; //ввод элемента массива

}}

for (i = 0; i < n; i++) //цикл с удобным выводом массива A

{for (j = 0; j < m; j++)

{cout << A[i][j] << " ";}

cout << endl;}

for (j = 0; j < m; j++) //Цикл для ввода массива В

{cout << "vvedite " << j << " elemnt massiva B ";

cin >> B[j];}}

template <typename T, typename K >

void prikoli(T \*\*&A, T \*&B, T \*&C, K &n, K &m, T &sum)

{ int i, j;

for (i = 0; i < m; i++) //создание цикла для ввода массива С с определнными условиями

{if (B[i] > 0) //первое условие

{for (j = 0; j < n; j++) //реализация первого условия согласно заданию (суммирование i-того столбца массива А и запись суммы в i-тый элемент массива С)

{sum = sum + A[j][i];}

C[i] = sum;

sum = 0;//обнуление переменной суммирования для последующих сумм

}

else if (B[i] == 0)//Второе условие (если iтый элемент массива В=0 то и iтый элемент массива С=0)

{C[i] = 0;}

else if (B[i] < 0)//Третье условие (если iтый элемент массива В<0 то и iтый элемент массива С высчитывается по формуле в коде)

{C[i] = A[0][i] + A[4][i];}} }

template <typename T, typename K >

void vivod(T \*&C, K &m)

{int j;

for (j = 0; j < m; j++) //вывод массива С

{cout << "C[" << j << "]=" << C[j] << endl}}

Протокол выполнения

Рисунок 2, протокол выполнения программы