

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего
профессионального образования
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО
ПРИБОРОСТРОЕНИЯ»

КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ПРОГРАММНОЙ ИНЖЕНЕРИИ

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
РУКОВОДИТЕЛЬ

Старший преподаватель
должность, уч. степень, звание

подпись, дата

Е.О.Шумова
инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ

№5

**«Стандартная библиотека C++. Последовательные и ассоциативные
контейнеры. Обобщенные алгоритмы »»**

по дисциплине: Объектно-ориентированное программирование

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. Z7431

20.01.2020

подпись, дата

М.Д.Семочкин

инициалы, фамилия

Студ. билет 2014/1054

Санкт-Петербург

2020

Цель работы

Изучить принципы построения консольных приложений, применив на практике знания базовых синтаксических конструкций языка C++ и объектно-ориентированного программирования.

Закрепить знания по теме: Классы, конструкторы и деструкторы, права доступа. Операторные функции в пространстве имен и как члены класса. Стандартная библиотека C++. Библиотека ввода-вывода.

Задание

Приложение должно осуществлять ввод и вывод информации о реализованном классе. Заполнить массив данных (vector) случайными числами в диапазоне $m1 - m2$. Выполнить набор действий при помощи обобщенных алгоритмов, объектов-функций и предикатов. После выполнения каждого действия выводить на экран результат.

Вариант 7:

- $m1=0, m2=100$
- подсчитать количество элементов со значениями больше 10
- найти корень квадратный из всех элементов
- вычислить сумму всех элементов

Используемые инструменты разработки

IDE – Microsoft Visual C++ 2010 Express

Текст программы

```
// Приложение должно осуществлять ввод и вывод информации о реализованном классе.
// Заполнить массив данных (vector) случайными числами в диапазоне m1 - m2 .
// Выполнить набор действий при помощи обобщенных алгоритмов, объектов-функций и
// предикатов.
// После выполнения каждого действия выводить на экран результат.
//
// Вариант 7
//
// - m1=0, m2=100
// - подсчитать количество элементов со значениями больше 10
// - найти корень квадратный из всех элементов
// - вычислить сумму всех элементов
//

#include <iostream>
#include <stdlib.h>
#include <vector>
#include <random>
#include <math.h>
using namespace std;

int const ARR_LENGTH = 10;
int const M1 = 0;
int const M2 = 100;

class MyArray {
    private:
        vector<double> content;

    public:
        void print();
        void fillWithRandomNumbers();
        int countNumberOfGT10();
        void sqrtAllElements();
        double getSumOfAllElements();
};

void MyArray::print() {
    for(int i = 0; i < content.size(); i++) {
        cout << content[i] << " ";
    }
    cout << endl << endl;
}
```

```

void MyArray::fillWithRandomNumbers() {
    random_device rd;        // only used once to initialise (seed) engine
    mt19937 rng(rd());        // random-number engine used (Mersenne-Twister in this case)
    uniform_int_distribution<int> uni(M1, M2); // guaranteed unbiased

    for(int i = 0; i < ARR_LENGTH; i++) {
        auto random_integer = uni(rng);
        content.push_back(random_integer);
    }
}

int MyArray::countNumberOfGT10() {
    int GT10 = 0;

    for(int i = 0; i < content.size(); i++) {
        if (content[i] > 10) {
            GT10++;
        }
    }

    return GT10;
}

void MyArray::sqrtAllElements() {
    for(int i = 0; i < content.size(); i++) {
        content[i] = sqrt(content[i]);
    }
}

double MyArray::getSumOfAllElements() {
    double sum = 0;

    for(int i = 0; i < content.size(); i++) {
        sum += content[i];
    }

    return sum;
}

int main(int argc, char **argv) {
    setlocale(LC_ALL, "russian");

    MyArray arr0;

    cout << "Создаем vector и заполняем случайными числами: " << endl;
    arr0.fillWithRandomNumbers();
    arr0.print();

    cout << "Количество элементов со значением больше 10: " <<
        arr0.countNumberOfGT10() << endl << endl;

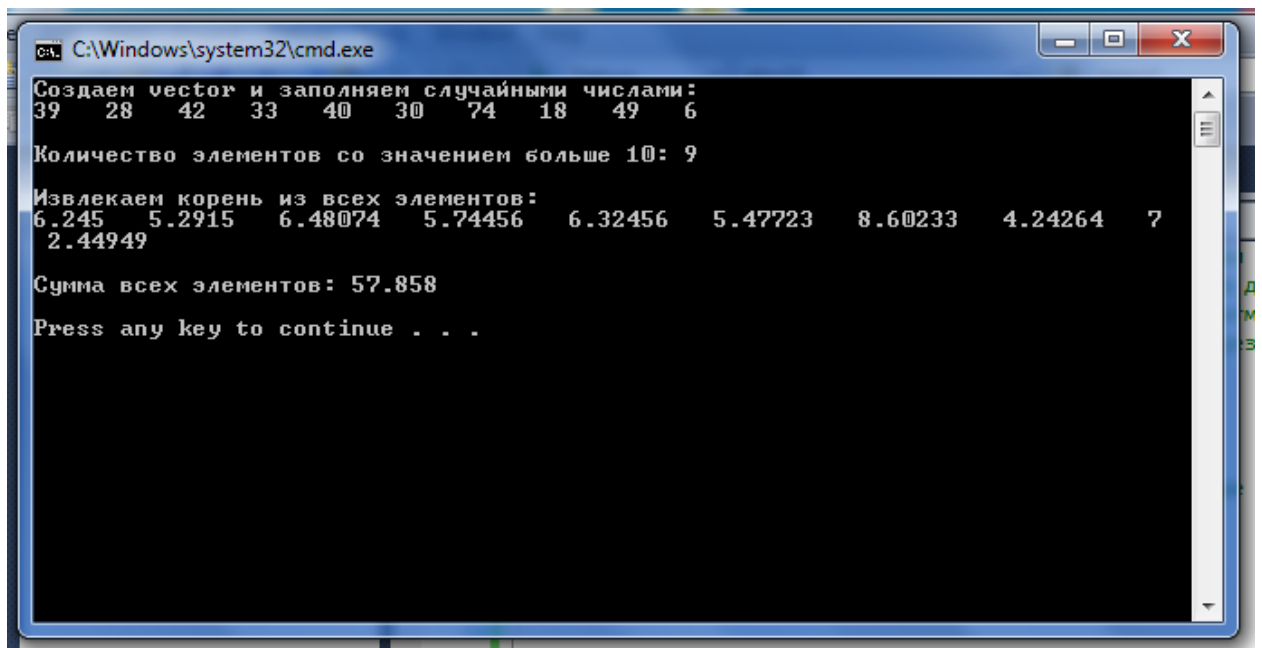
    cout << "Извлекаем корень из всех элементов: " << endl;
    arr0.sqrtAllElements();
    arr0.print();
}

```

```
    cout << "Сумма всех элементов: " <<
        arr0.getSumOfAllElements() << endl << endl;

    return 0;
}
```

Результат работы программы



```
cmd.exe
Создаем vector и заполняем случайными числами:
39 28 42 33 40 30 74 18 49 6
Количество элементов со значением больше 10: 9
Извлекаем корень из всех элементов:
6.245 5.2915 6.48074 5.74456 6.32456 5.47723 8.60233 4.24264 7
2.44949
Сумма всех элементов: 57.858
Press any key to continue . . .
```

Вывод

Во время выполнения лабораторной работы были изучены принципы построения консольных приложений, применены на практике знания базовых синтаксических конструкций языка C++ и объектно-ориентированного программирования.