

**1. 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 С++. Библиотека ввода-вывода.

**Вариант 8**

* m1=-10, m2=10
* переставить все элементы массива в обратном порядке
* для каждого элемента вычислить выражение x = (x<0) ? 0 : x\*2
* подсчитать количество элементов со значениями не равными 0

**2. Листинг программы**

#include <algorithm>

#include <ctime>

#include <functional>

#include <iostream>

#include <vector>

using namespace std;

const int LENGTH = 10;

vector<int> v;

// x = (x<0) ? 0 : x \* 2

int f(int x) { return (x < 0) ? 0 : x \* 2; }

// x != 0

bool c(int x) { return x != 0; }

void menu() {

int inp;

cout << endl << "Select menu item:" << endl;

cout << "1: Print" << endl;

cout << "2: Reverse" << endl;

cout << "3: Transform" << endl;

cout << "4: Count" << endl;

cout << "0: Exit" << endl;

cin >> inp;

switch (inp) {

case 1: {

cout << endl;

for (auto i = v.begin(); i != v.end(); ++i)

cout << \*i << ' ';

menu();

break;

};

case 2: {

reverse(v.begin(), v.end());

menu();

break;

};

case 3: {

transform(v.begin(), v.end(), v.begin(), f);

menu();

break;

};

case 4: {

cout << "x != 0: " << count\_if(v.begin(), v.end(), c) << endl;

menu();

break;

};

case 0: {

break;

};

default:

cout << "Invalid Input";

menu();

break;

}

}

int main() {

srand(time(nullptr));

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

int val = rand();

v.push\_back(val % 20 - 10);

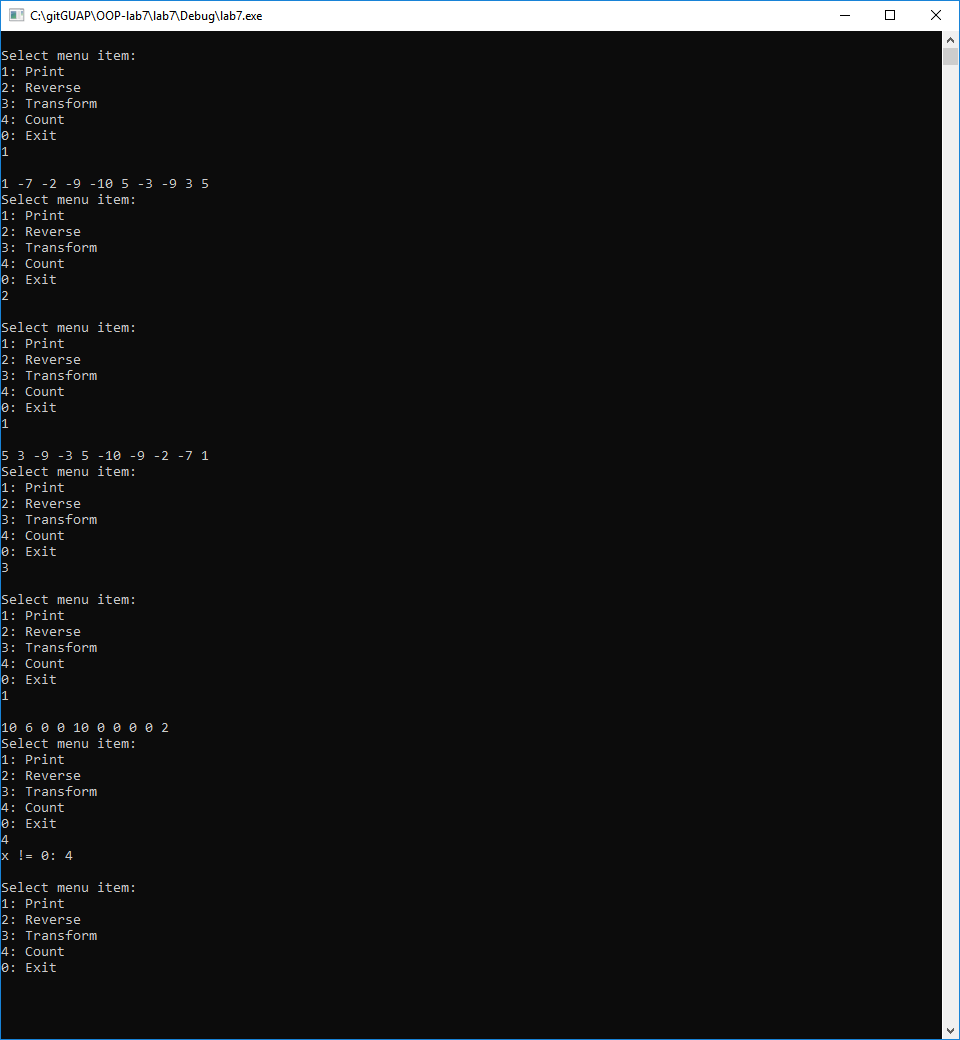
}

menu();

return 0;

}

**3. Пример работы**



**4. Вывод**

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