

КАФЕДРА №

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

должность, уч. степень, звание

подпись, дата

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №7

Стандартная библиотека C++. Последовательные и ассоциативные контейнеры.
Обобщенные алгоритмы

по курсу: Объектно-ориентированное программирование

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №

подпись, дата

инициалы, фамилия

Санкт-Петербург 2022

Условие

6 вариант

Вариант 6.

- $m1=-50, m2=50$
- поделить все элементы массива на 2
- обменять зеркально 2 первых элемента массива с 2 последними
- заменить все элементы, абсолютное значение которых меньше 10 на 0

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

Реализовать класс, содержащий:

- массив данных (вектор), заполненный случайными числами в диапазоне $m1$ - $m2$;
- методы, обеспечивающие выполнение действий (согласно варианта) с использованием обобщенных алгоритмов, объектов-функций и предикатов (использование обобщенных алгоритмов обязательно;
- обеспечить вывод результатов после выполнения каждого действия

$m1=-50, m2=50$

- поделить все элементы массива на 2
- обменять зеркально 2 первых элемента массива с 2 последними
- заменить все элементы, абсолютное значение которых меньше 10 на 0

Листинг программы

main.cpp

```
#include <iostream>
using namespace std;

#include <cmath>

#include "arr_vec.h"

int main() {
    draw_line(60);
    // смена кодировки
    system("chcp 65001");

    // srand(time(NULL));

    Arr_vec arr_vec(10);
    cout << "Сгенерированный вектор." << endl;
    arr_vec.show();

    arr_vec.divide_all();
```

```

    cout << "Поделили все элементы массива на 2" << endl;
    arr_vec.show();

    if (arr_vec.swap_e1s(2)) {
        cout << "Зеркально обменяли 2 первых и 2 последних элемента."
<< endl;
        arr_vec.show();
    } else {
        cout << "Невозможно обменять 2 первых и 2 последних элемента
вектора, т.к. он мал." << endl;
    }

    arr_vec.chenage_on_zero();
    cout << "Заменяли элементы массива (всё что меньше 10 заменить
на 0)." << endl;
    arr_vec.show();

    return 0;
}

```

arr_vec.h

```

#include <iostream>
using namespace std;

#include <algorithm>

#include <iomanip>
#include <vector>

#define rand_min -50 // m1
#define rand_max 50 // m2
// #define separator_show " " //

// рисует линию в терминале
void draw_line(int size = 20) {
    for (int i = 0; i < size; i++)
        cout << '-';
    cout << endl;
}

// генерирует случайное число в диапазоне от А до В
int random_int(int a, int b) {
    return a + (rand() % (b - a + 1));
}

// класс реализующий работу с вектором
class Arr_vec {
public:
    Arr_vec(int);
    ~Arr_vec();

    void generator(int);
    void show(const char* separator);
    void chenage_on_zero();
    void devide_all();
    bool swap_e1s(int);
}

```

```

private:
    vector<int> my_vector;
};

// конструктор
Arr_vec::Arr_vec(int Size) {
    // my_vector.reserve(Size);
    generator(Size);
}

// деструктор
Arr_vec::~Arr_vec() {

}

// метод заполняющий вектор случайными числами
void Arr_vec::generator(int Size) {
    for (int i = 0; i < Size; i++) {
        my_vector.push_back(random_int(rand_min, rand_max));
    }
}

// вывод массива на экран
void Arr_vec::show(const char* separator = " ") {
    for (int i = 0; i < my_vector.size(); i++) {
        cout << setw(4) << my_vector.at(i) << separator;
    }
    cout << endl;
}

// замена элементов массива (всё что меньше 10 заменить на 0)
void Arr_vec::change_on_zero() {
    // for (int i = 0; i < my_vector.size(); i++) {
    //     if (my_vector.at(i) < 10)
    //         my_vector.at(i) = 0;
    // }
    replace_if(my_vector.begin(), my_vector.end(), [](int i) {return
i < 10;}, 0);
}

// поделить все элементы массива на 2
int xform(int i) {
    return i / 2;
}

void Arr_vec::divide_all() {
    transform(my_vector.begin(), my_vector.end(), my_vec-
tor.begin(), xform);
}

// зеркально обменять 2 первых элемента массива с 2 последними
bool Arr_vec::swap_els(int count = 2) {
    if (count * 2 > my_vector.size())
        return false;

    for (int i = 0; i < count; i++) {
        swap(my_vector.at(i), my_vector.at(my_vector.size() - i - 1));
    }

    return true;
}

```

}

Скриншоты

Вывод

Мы изучили работу с векторами и обобщёнными методами.