

Министерство Образования Республики Беларусь
УО Брестский Государственный Технический Университет
Кафедра интеллектуальных информационных технологий

Лабораторная работа №7.
"Контейнерные классы"

Выполнил:
Студент 2-го курса
Группы АС-53
Вожейко М.В.
Проверил:
Давидюк Ю. И.

Брест, 2020

1. **Цель.** Получить практические навыки работы с контейнерными классами STL.

2. Постановка задачи

Задание 1

Новая последовательность сначала должна содержать все элементы исходной, значение которых по модулю меньше 1, затем - все остальные (с сохранением исходного относительного порядка).

Задание 2.

6. Сумму элементов последовательности от начала до последнего положительного элемента (включительно).

1. Код программы

```
#include <iostream>
#include <math.h>
#include <vector>

int main() {
    setlocale(0, "RUS");
    std::vector<double> values; //создаем вектор, генерим туда случайные числа
    std::vector<double> results;
    double summ = 0;
    rand(); //чтобы при каждой компиляции генерировалось новое число
    std::vector<double>::const_iterator iter, lastPos;
    for (int i = 0; i < 10; i++) //заполняем случайными числами
        if (i % 5 == 0)
            values.push_back(0); //здесь добавляем 1 ноль чтобы гарантированно было нулевое число
        else
            values.push_back(-3 + rand() % 6 + rand() % 100 / 100.0);
    for (double element : values) // Добавляем в вектор числа, модуль которых меньше 1
        if (fabs(element) <= 1)
            results.push_back(element);
    for (double element : values) // Добавляем в вектор числа, модуль которых больше 1
        if (fabs(element) > 1)
            results.push_back(element);

    std::cout << "Исходная последовательность:" << std::endl; //печатаем сначала те которые были
    //сгенерены случайно
    for (int i = 0; i < values.size(); i++)
        std::cout << values[i] << " | ";
    std::cout << std::endl;

    std::cout << "Задание #1:" << std::endl; //печатаем а теперь те что в нужной
    //последовательности
    for (int i = 0; i < results.size(); i++)
        std::cout << results[i] << " | ";

    lastPos = values.begin();
    for (iter = values.begin(); iter != values.end(); iter++) { // Ищем первое и второе
    //отрицательное значение
        if (*iter >= 0) {
            lastPos = iter;
        }
    }
    for (iter = values.begin(); iter <= lastPos; iter++)
    {
        summ += *iter;
    }
    std::cout << "\nПоследнее положительное: " << *lastPos << std::endl;
```

```
std::cout << "\nЗадание #2(Используется исходная последовательность). Сумма: " << summ <<  
std::endl;  
return 0;  
}
```

2. Вывод:

Получил практические навыки работы с контейнерными классами C++.