

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

1. В новой последовательности сначала должны идти все отрицательные элементы исходной, затем все нулевые элементы, затем все положительные (с сохранением исходного относительного порядка).
2. Сумму элементов последовательности, расположенных между первым и вторым отрицательными элементами.

3. 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;
    int countNeg = 0;
    rand(); //чтобы при каждой компиляции генерировалось новое число
    std::vector<double>::iterator iter, firstNeg, secondNeg; //создаем итератор по вектору, 2
    итератора, которые будут указывать на 1 и 2 отрицательных числа
    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(-10 + rand() % 20);
    for (double element : values) // Добавляем в вектор отрицательные элементы
        if (element < 0)
            results.push_back(element);
    for (double element : values) // Добавляем в вектор нулевые элементы
        if (element == 0)
            results.push_back(element);
    for (double element : values) // Добавляем в вектор положительные элементы
        if (element > 0)
            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] << " | ";

    firstNeg = values.begin();
    secondNeg = values.begin();
    for (iter = values.begin(); iter != values.end(); iter++) { // Ищем первое и второе
    отрицательное значение
        if (*iter < 0) {
            switch (countNeg)
            {
```

```

        case 0://если мы словили первое, то просто указываем итератором на него, в счетчик
негативных чисел добавляем еще одно число
            firstNeg = iter;
            countNeg++;
            break;
        case 1://если уже одно было, то теперь указываем итератором secondNeg на новое
отрицательное число, увеличиваем счетчик отриц. чисел на 1
            secondNeg = iter; countNeg++; break;
        default:
            break;
    }
    if (countNeg >= 2) break;
}
}
std::cout << "\nПервое отр.: " << *firstNeg << "; Второе отр.: " << *secondNeg << std::endl;
for (iter = ++firstNeg; iter < secondNeg; iter++) {
    summ += *iter;
}

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

```

4. Результат выполнения программы

```

Исходная последовательность:
0 | -3 | 4 | -10 | -1 | 0 | -6 | 8 | 8 | -8 |
Задание #1:
-3 | -10 | -1 | -6 | -8 | 0 | 0 | 4 | 8 | 8 |
Первое отр.: -3; Второе отр.: -10

```

```

Задание #2(Используется исходная последовательность). Сумма: 4

```

5. Вывод:

Получил практические навыки работы с контейнерными классами C++.