Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №7.

"Контейнерные классы”

Выполнил:

Ст. 2 курса гр. АС-53

Альциванович Н.В.

Проверила:

Давидюк Ю. И.

Брест, 2020

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

STL.

1. **Постановка задачи (Вариант 1)**

**1.**Дана последовательность действительных чисел. Необходимо

сформировать новую последовательность по некоторому правилу:

Новая последовательность должна содержать все ненулевые элементы исходной (с сохранением исходного относительного порядка).

**2.**Дана последовательность действительных чисел. Необходимо вычислить:

Произведение элементов последовательности, расположенных между

максимальным и минимальным элементами.

1. **Код программы**

**#include** **<**iostream**>**

**#include** **<**QVector**>**

**#include** **<**math.h**>**

**void** task1();

**void** task2();

**int** main(){

std::cout << "Task 1:\n";

task1();

std::cout << "\nTask 2:\n";

task2();

**return** 0;

}

**void** task1(){

QVector<**int**> vector1,

                 vector2;

**int** value = 0;

std::cout<<"Enter the value:\n";

**do**{

std::cin >> value;

vector1.append(value);

**if** (value != 0) {

            vector2.append(value);

        }

} **while** (std::cin.get() != '\n');

**for** (**int** i = 0; i < vector2.length(); i++){

std::cout << vector2[i] << ' ';

}

}

**void** task2(){

QVector<**int**> vector1;

**int** value = 0,

min = 044,

max = 0,

minPos = 0,

maxPos = 0,

mult=1;

{ *//First digit*

     std::cin >> value;

vector1.append(value);

min = value;

max = value;

}

**do** {

std::cin >> value;

vector1.append(value);

**if** (value < min){

min = value;

}

**if** (value > max){

max = value;

}

} **while** (std::cin.get() != '\n');

std::cout<< "Min= "<<min <<"\nMax= "<< max;

maxPos = vector1.indexOf(max);

minPos = vector1.indexOf(min);

**if**(maxPos<minPos){

QMutableVectorIterator<**int**> i(vector1);

**while**(i.hasNext()){

**if**(i.peekNext() == max){

i.next();

**while**(i.peekNext() != min){

mult\*=i.next();

}

**break**;

}**else**{

i.next();

}

}

}**else**{

QMutableVectorIterator<**int**> i(vector1);

**while**(i.hasNext()){

**if**(i.peekNext() == min){

i.next();

**while**(i.peekNext() != max){

mult\*=i.next();

}

**break**;

}**else**{

i.next();

}

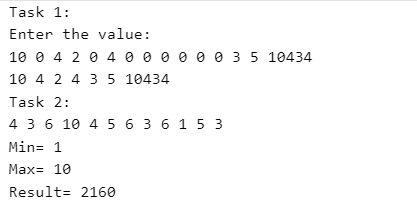
}

}

std::cout <<"\nResult= "<< mult;

}

* 1. **Результат выполнения программы**



**5. Вывод:**

Получил практические навыки работы с контейнерными классами С++.