Специальность: 09.02.07 Информационные системы и программирование

Выполнил:

Студент группы Y2433

Тертень Д.В.

ОТЧЕТ

о лабораторной работе № 1

по теме: Разветвляющиеся программы. Циклы.

по дисциплине: Разработка программных модулей

Санкт-Петербург 2020

Проверил:

Антонов. М.Б.

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2020 г.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

УНИВЕРСИТЕТ ИТМО

Факультет среднего профессионального образования

1. Цель и задачи

Цель: Научиться работать с шаблонами функций и одномерными массивами.

Задачи:

* реализовать несколько функций, использующих шаблон-функции;
* реализовать динамическую типизацию внутри функций.

1. Задание

Дан массив из 20 элементов, заранее предопределённого типа (к примеру int)

Необходимо написать 4 функции возвращающие один результат:

* минимальное значение массива min\_seq
* максимальное значение массива max\_seq
* среднее значение массива mid\_seq
* сумму всех элементов массива sum\_seq

Все функции должны быть реализованы в виде шаблонов. Способ проверки: при изменении типа изначального массива в коде программы - вся остальная программа должна остаться неизменной.

1. КОд программы

Файл main.cpp:

/\*\*

\* @file main.cpp

\* @author Terten Daniil

\* @brief Лабораторная работа 1

\*/

#include <iostream>

using namespace std;

/\*\*

\* @brief Возвращает сумму последовательности

\* @param sequence Последовательность

\* @param sizeArray Размер последовательности

\* @return Сумма последовательности

\*/

template <typename A>

A sum\_seq (A \*sequence, int sizeArray)

{

A sum = 0;

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

sum += sequence[i];

}

return sum;

}

/\*\*

\* @brief Возвращает среднее значение последовательности

\* @param sequence Последовательность

\* @param sizeArray Размер последовательности

\* @return Среднее значение

\*/

template <typename A>

A mid\_seq (A \*sequence, int sizeArray)

{

return sum\_seq(sequence, sizeArray) / sizeArray;

}

/\*\*

\* @brief Возвращает максимальное значение последовательности

\* @param sequence Последовательность

\* @param sizeArray Размер последовательности

\* @return Максимальное значение

\*/

template <typename A>

A max\_seq (A \*sequence, int sizeArray)

{

A maxEl = sequence[0];

for(int i = 1; i < sizeArray; i++){

if(sequence[i] > maxEl){

maxEl = sequence[i];

}

}

return maxEl;

}

/\*\*

\* @brief Возвращает минимальное значение последовательности

\* @param sequence Последовательность

\* @param sizeArray Размер последовательности

\* @return Минимальное значение

\*/

template <typename A>

A min\_seq (A \*sequence, int sizeArray)

{

A minEl = sequence[0];

for(int i = 1; i < sizeArray; i++){

if(sequence[i] < minEl){

minEl = sequence[i];

}

}

return minEl;

}

/\*\*

\* @brief Печатает все вычисляемые значения для последовательности

\* @param sequence Последовательность

\* @param sizeArray Размер последовательности

\*/

template <typename A>

void printResult(A \*sequence, int sizeArray)

{

cout << "Average: " << mid\_seq(sequence, sizeArray) << endl;

cout << "Max: " << max\_seq(sequence, sizeArray) << endl;

cout << "Min: " << min\_seq(sequence, sizeArray) << endl;

cout << "Sum: " << sum\_seq(sequence, sizeArray) << endl;

}

/\*\*

\* @brief Функция выбора типа массива

\* @return Возвращает выбранный тип

\*/

int selectType()

{

int type;

cout << "Select type: 1-int 2-float 3-double" << endl;

cin >> type;

return type;

}

/\*\*

\* @brief Выбор массива для вычислений в зависимости от выбора пользователя

\* @param sequence1 Массив int

\* @param sequence2 Массив float

\* @param sequence3 Массив double

\* @param sizeArray Размер массивов

\*/

template <typename A, typename B, typename C>

void exec(A \*sequence1, B \*sequence2, C \*sequence3, int sizeArray)

{

int type = selectType();

if (type == 1) {

printResult(sequence1, sizeArray);

} else if (type == 2) {

printResult(sequence2, sizeArray);

} else {

printResult(sequence3, sizeArray);

}

}

/\*\*

\* @brief Точка входа в программу, инициализация последовательностей разных типов

\* @return 0

\*/

int main()

{

int sequenceInt[20] = {80, 34, 4, 20, 84, 37, 46, 82, 17, 7, 80, 71, 59, 30, 77, 34, 69, 8, 86, 85};

float sequenceFloat[20] = {80.3, 34, 4, 20, 84, 37, 46, 82, 17, 7, 80, 71, 59, 30, 77, 34, 69, 8, 86, 85};

double sequenceDouble[20] = {80.3, 34, 4, 20, 84, 37, 46, 82, 17, 7, 80, 71, 59, 30, 77, 34, 69, 8, 86, 85};

int sizeArray = 20;

exec(sequenceInt, sequenceFloat, sequenceDouble, sizeArray);

return 0;

}

В качестве результата работы программы представлены скриншоты результатов выполнения программы на рисунках 1-2.

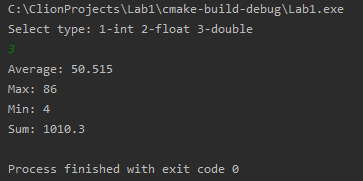


Рисунок 1 – выбор последовательности Double и вывод всех расчётов

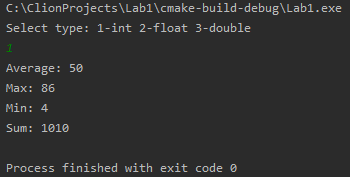


Рисунок 2 – выбор последовательности Int и вывод всех расчётов

# ВЫВОД

В процессе выполнения лабораторной работы были получены навыки работы с шаблонами, одномерными массивами и передачей массивов в функции.