

Лабораторная работа №22

Изучение процесса разработки шаблонов функций и классов на C++

1 Цель работы

- 1.1 Изучить процесс разработки шаблонов функций и классов на C++;
- 1.2 Изучить реализацию параметрического полиморфизма на языке C++;
- 1.3 Закрепить навык разработки программ методами структурного, процедурного и модульного программирования.

2 Литература

2.1 Ашарина, И. В. Объектно-ориентированное программирование в C++ : лекции и упражнения. Учебное пособие для вузов. / И. В. Ашарина. – 2-е изд., перераб. и доп. – Москва : Горячая Линия–Телеком, 2017. – 336 с. – URL: <https://ibooks.ru/reading.php?productid=359752>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный. – гл.25.

3 Подготовка к работе

- 3.1 Повторить теоретический материал (см. п.2).
- 3.2 Изучить описание лабораторной работы.

4 Основное оборудование

- 4.1 Персональный компьютер.

5 Задание

5.1 Разработать шаблон функции `Maximum` для поиска и возврата максимального значения из двух переданных в параметрах.

Протестировать функцию, вызвав ее со значениями различных типов данных. Значения задать программно в основной программе.

5.2 Разработать шаблон функции `IndexOfMin` для поиска и возврата индекса элемента массива с максимальным значением. В функцию передается массив и его размерность:

```
template <typename T>
int IndexOfMin(T *array, const int size)
{
    // тело функции
}
```

Протестировать функцию, вызвав ее с значениями различных типов данных. Значения задать программно в основной программе, например:

```
int a[5] = {10, 34, 167, 12, -5};
```

5.3 Создать шаблон класса `List` (список, в котором по умолчанию 10 элементов, но можно передать другую размерность), в котором объявить закрытые поля `array` (массив) и `count` (количество элементов в списке):

```
template <typename T, int size=10>
class List
{
    T array[size];
    int count;
};
```

Добавить в шаблон класса следующие открытые элементы:

- конструктор по умолчанию, в котором присвоить count значение 0;
- метод int Count(), возвращающий количество элементов в списке.

Протестировать шаблон класса, создав объекты класса и вызвав разработанные функции.

5.4 Добавить в шаблон класса List метод void Add(T value), дописывающий в конец списка переданное в параметрах значение и увеличивающий на 1 значение поля count. При реализации учесть, что нельзя добавлять элементы, если size и count совпадают.

Протестировать разработанный метод, добавив в один из объектов типа список не менее 5 элементов.

5.5 Добавить в шаблон класса List следующий метод (по вариантам) и протестировать его:

№ варианта	Задание
1	IndexOf(T value), возвращающий индекс первого вхождения указанного элемента в список. Если совпадение не найдено, возвращать -1.
2	CountOf(T value), возвращающий количество вхождений указанного элемента в список.

6 Порядок выполнения работы

6.1 Используя Microsoft Visual Studio, создать проект C++ и выполнить задания из п.5.

6.2 Ответить на контрольные вопросы.

7 Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

8 Контрольные вопросы

8.1 Для чего применяются шаблоны в C++?

8.2 Какова общая форма шаблона функции?

8.3 Какова общая форма шаблона класса?

8.4 Как передать в шаблон несколько обобщенных типов?

8.5 Как задаются значения параметров шаблона при создании объекта?