

на компьютер нельзя быть уверенным в порядке изменения битов (слева направо или справа налево). Любой код, использующий битовые поля, зависит от компьютера.

УДК 621.762.4

Веренич М.С.

ИСПОЛЬЗОВАНИЕ ШАБЛОНОВ В C/C++

БНТУ, Минск

Научный руководитель: Дробыш А.А.

Шаблоны – средство языка C++, предназначенное для кодирования обобщённых алгоритмов, без привязки к некоторым параметрам (например, типам данных, размерам буферов, значениям по умолчанию).

В C++ возможно создание шаблонов функций и классов.

Шаблоны позволяют создавать параметризованные классы и функции. Параметром может быть любой тип или значение одного из допустимых типов (целое число, указатель на любой объект с глобально доступным именем, ссылка и т.д.). Например, нам нужен какой-то класс:

```
class SomeClass{
    int SomeValue;
    int SomeArray[20];
    ...
};
```

Для одной конкретной цели мы можем использовать этот класс. Но, вдруг, цель немного изменилась, и нужен еще один класс. Теперь нужно 30 элементов массива `SomeArray` и вещественный тип `SomeValue` элементов `SomeArray`. Тогда мы можем абстрагироваться от конкретных типов и использовать шаблоны с параметрами.

Синтаксис: в начале перед объявлением класса напишем слово `template` и укажем параметры в угловых скобках. В нашем примере:

```
template < int ArrayLength, typename SomeValueType >
class SomeClass{
    SomeValueType SomeValue;
    SomeValueType SomeArray[ ArrayLength ];
    ...
};
```

Тогда для первой модели пишем:

```
SomeClass < 20, int > SomeVariable;
```

для второй:

```
SomeClass < 30, double > SomeVariable2;
```

Хотя шаблоны предоставляют краткую форму записи участка кода, на самом деле их использование не сокращает исполняемый код, так как для каждого набора параметров компилятор создаёт отдельный экземпляр функции или класса. Как следствие, исчезает возможность совместного использования скомпилированного кода в рамках разделяемых библиотек.

УДК 371

Войткевич И.С.

КАЧЕСТВО ПРОГРАММНОГО КОДА

БНТУ, Минск

Научный руководитель: Дробыш А.А.

При создании любой программы разработчики беспокоятся прежде всего о ее работоспособности, ведь если приложение не удовлетворяет требованиям заказчика, то вас вряд ли спасут рассказы об изящной внутренней архитектуре, эффективных алгоритмах и других достоинствах вашего детища.