

Фредерик Брукс-младший придумал высказывание, которое звучит так: «Выдавать глобальные идеи – это удовольствие; искать сволочные маленькие ошибки – вот настоящая работа». Это высказывание заставляет задуматься о том, что глобальная идея может прийти в голову спонтанно, а нахождение ошибок в коде – это кропотливый труд.

Итак, проанализировав разные источники, может прийти к выводу, что законы программирования существуют и будут использоваться на протяжении многих лет.

УДК 621

Романенко А. С.

## **БИТОВЫЕ ПОЛЯ И ОПЕРАЦИИ НАД НИМИ**

*БНТУ, Минск*

*Научный руководитель Дробыш А. А*

Битовое поле – некоторое количество бит, расположенных последовательно в памяти, значение которых процессор не способен прочитать из-за особенностей аппаратной реализации. Битовые поля могут оказаться полезными во многих ситуациях:

- если память ограничена, в одном байте можно хранить несколько логических переменных, принимающих значение true и false;
- когда в одном байте нужно хранить информацию о состоянии некоторых устройств, закодированную несколькими битами;
- если шифровальным процедурам требуется доступ к отдельным битам.

Порядок размещения битовых полей в памяти в значительной степени зависит от компьютера и аппаратного обеспечения. Поля могут описываться только в качестве элементов

структурного шаблона аналогично объявлению целой знаковой или без знаковой переменной, после имени которой через двоеточие записывается целая константа, определяющая размер битового поля. В описании допускаются неименованные битовые поля (для них имя опускается, а указывается только двоеточие и размер), которые используются как заполнители пространства битового поля. Битовые поля размещаются в направлении от младших к старшим битам в слове.

Минимальная длина битового поля, естественно, равняется 1 (одному биту), максимальная длина зависит от реализации. Битовое поле длинной в восемь бит – не байт. Байт – это минимальная адресуемая область памяти ЭВМ, битовое поле – языковая конструкция. Среди форм Бэкуса-Наура, посвященных объявлению класса, напомним соответствующую БНФ:

**Описатель\_Члена\_Класса::=[Идентификатор]: Константное Выражение**

Вот такой описатель члена класса и задает битовое поле. Битовое поле может существовать исключительно как элемент класса. Идентификатор (необязательный!) задаёт имя поля, константное выражение – размеры этого поля в битах. Согласно ранее приведённым БНФ, подобному описателю должны предшествовать спецификаторы объявления. Как известно, они специфицируют тип объявляемого члена класса.

Битовые операции – тестирование, установка или сдвиг битов в байте или слове, который соответствует стандартным типам языка C `char` и `int`. Битовые операции не могут использоваться с `float`, `double`, `long double`, `void` и другими сложными типами.

**Назначение битовых операций:**

– разработчики Windows всегда старались сделать так, чтобы старшие поколения Windows поддерживали, по возможности, младшие версии, то есть, приложения, работающие в 16-разрядах Windows, должны поддерживать 32-разрядные;

- к битовым операциям необходимо прибегать при работе с некоторыми системными функциями;
- битовые операции часто приходится использовать при работе с графикой (имеется в виду программ с графикой).

В С имеются следующие битовые операции:

& – битовое и;  
| – битовое или;  
^ – битовое исключающая или;  
~ – инвертируют каждый разряд;  
<< – поразрядный сдвиг влево;  
>> – поразрядный сдвиг вправо.

Таким образом, можно сказать, что битовое поле – это своеобразная структура, которая позволяет работать с отдельными битами. Не смотря на то что, при решении задач можно использовать побитовые операторы, битовые поля позволяют создавать более простые и эффективные программы.

УДК 621

Рудакова В. О.

## ОСНОВНЫЕ ПРАВИЛА ПРОГРАММИРОВАНИЯ

БНТУ, Минск

Научный руководитель Дробыш А. А.

Правила здесь довольно общего характера по своей природе, они совсем не затрагивают техники программирования на Си или Си++, а скорее рассматривают более общий процесс проектирования и разработки программы. В известном смысле, большинство правил предназначены для управленицев; программисты их часто знают, но у них нет свободы, необходимой, чтобы воспользоваться своими знаниями.

Не решайте проблем, которых не существует. Решайте конкретную проблему, а не общий случай. Интерфейс пользователя