

C++ Starter

Логические операторы

C++ Starter

Автор курса



Владимир Виноградов



MCID: 9210561

C++ Starter

После урока обязательно



Повторите этот урок в видео формате на [ITVDN.com](http://itvdn.com)

Доступ можно получить через руководство вашего учебного центра



Проверьте как Вы усвоили данный материал на [TestProvider.com](http://testprovider.com)

Логические операторы

C++ Starter

Логические выражения

Логическое выражение – это конструкция языка программирования, результатом вычисления которой является «истина» или «ложь».

В C++ существует три логических оператора:

- Логическое **И**
- Логическое **ИЛИ**
- Логическое **НЕ**

C++ Starter

Конъюнкция

Конъюнкция — это логическая операция, ставящая в соответствие каждому двум простым высказываниям составное высказывание, являющееся истинным только тогда, когда оба исходных высказывания истинны. Операция конъюнкции по своему применению близка к союзу «И».

В C++ обозначается как: **&&**

Таблица истинности

A	B	A && B
false	false	false
false	true	false
true	false	false
true	true	true

C++ Starter

Побитовое «И»

Побитовое И – это бинарная операция, действие которой эквивалентно применению логического И к каждой паре битов, которые стоят на одинаковых позициях в двоичных представлениях операндов.

В C++ обозначается как: **&**

Таблица истинности

A	B	A & B
0	0	0
0	1	0
1	0	0
1	1	1

result = 255 & 2

&	1 1 1 1 1 1 1 1	&	255
	0 0 0 0 0 0 1 0		2
	-----		----
	0 0 0 0 0 0 1 0		3

C++ Starter

Дизъюнкция

Дизъюнкция – это логическая операция, ставящая в соответствие каждому двум простым высказываниям составное высказывание, являющееся ложным только тогда, когда оба исходных высказывания ложны. Операция дизъюнкции по своему применению близка к союзу «**ИЛИ**».

В C++ обозначается как: `||`

Таблица истинности

A	B	A B
false	false	false
false	true	true
true	false	true
true	true	true

C++ Starter

Побитовое ИЛИ

Побитовое ИЛИ – это бинарная операция, действие которой эквивалентно применению логического ИЛИ к каждой паре битов, которые стоят на одинаковых позициях в двоичных представлениях операндов.

В C++ обозначается как: `|`

Таблица истинности

A	B	A B
0	0	0
0	1	1
1	0	1
1	1	1

result = 3 | 4

```
| 0000 0011 | 3
| 0000 0100 | 4
-----
0000 0111    7
```

C++ Starter

Отрицание

Отрицание – это логическая операция, которая с помощью связки «**НЕ**» каждому исходному высказыванию ставит в соответствие составное высказывание, заключающееся в том, что исходное высказывание отрицается.

В C++ обозначается как: !

Таблица истинности

A	!A
false	true
true	false

C++ Starter

Исключающее ИЛИ

Исключающее ИЛИ – это логическая операция, ставящая в соответствие каждому двум простым высказываниям составное высказывание, являющееся истинными тогда, когда оба операнда различаются.

В языке C++ отсутствует оператор логического ИЛИ, но его можно реализовать на основе встроенных.

$$(a \parallel b) \ \&\& \ ! (a \ \&\& \ b)$$

Таблица истинности

A	B	
false	false	false
false	true	true
true	false	true
true	true	false

C++ Starter

Побитовое отрицание

Побитовое отрицание (НЕ) – это унарная операция, действие которой эквивалентно применению логического отрицания к каждому биту, в двоичном представлении операнда.

В C++ обозначается как: \sim

Таблица истинности

A	$\sim A$
0	1
1	0

$$\sim 1 = -2$$

$$\begin{array}{r} 0000\ 0001 \\ \sim \hline 1111\ 1110 \end{array}$$

C++ Starter

Логические выражения

Что бы изменить знак числа, необходимо выполнить его отрицание, и полученный результат увеличить на 1.

Формула изменения знака числа

$$\sim (+N) + 1 = -N$$

$$\sim (-N) + 1 = +N$$

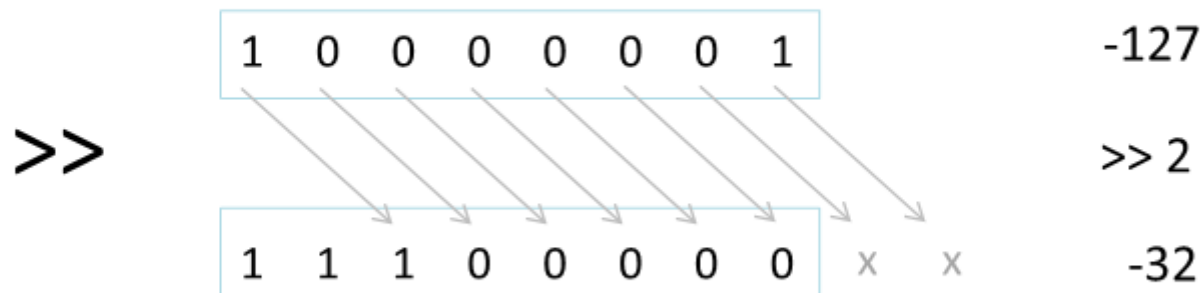
0000 0001	1
\sim 1111 1110	\sim -2
+ 0000 0001	+ 1
-----	--
1111 1111	-1

C++ Starter

Логические сдвиги числа

При логическом сдвиге значение последнего бита по направлению сдвига теряется (копируясь в бит переноса), а первый приобретает нулевое значение.

$$-127 \gg 2 = -32$$



Слева - если число положительное заходят нули,
а если отрицательное заходят единицы.

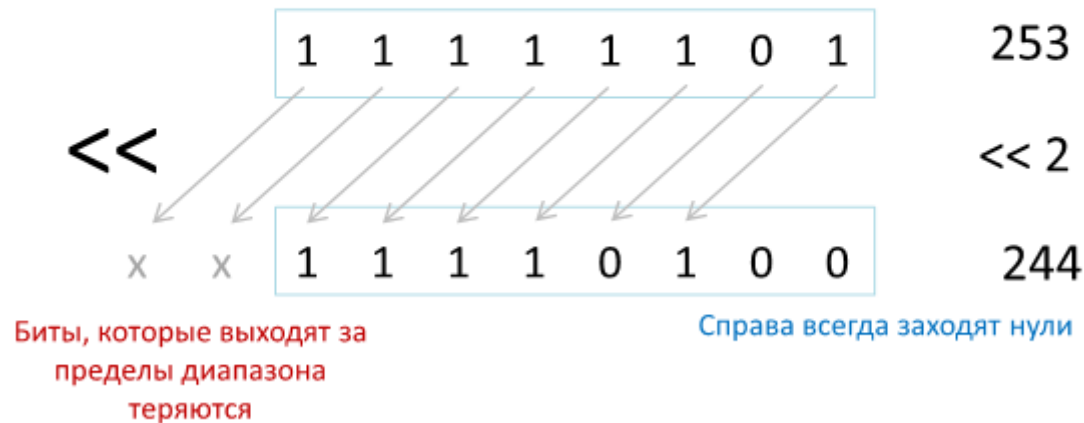
Биты, которые выходят за
пределы диапазона
теряются

C++ Starter

Логические сдвиги числа

При логическом сдвиге значение последнего бита по направлению сдвига теряется (копируясь в бит переноса), а первый приобретает нулевое значение.

$$253 \ll 2 = 244$$



C++ Starter

Символьные функции

Заголовочный файл `cctype` содержит функции обработки символов. Эта библиотека объявляет набор функций для выполнения различных классификаций и некоторых операций преобразования символов.

`isalnum()` – возвращает `true`, если аргумент – буква или десятичная цифра

`isalpha()` – возвращает `true`, если аргумент – буква

`isblank()` – возвращает `true`, если аргумент – пробел или знак горизонтальной табуляции

`iscntrl()` – возвращает `true`, если аргумент – управляющий символ

`isdigit()` – возвращает `true`, если аргумент – десятичная цифра

`isgraph()` – возвращает `true`, если аргумент – любой печатаемый символ, отличный от пробела

C++ Starter

Форматированный вывод

В языке C++ имеется возможность управлять вводом-выводом, это обеспечивают форматирующие функции члены, флаги и манипуляторы. Они выполняют одну и ту же задачу – задают определенный формат ввода\вывода информации в потоках. Различие между функциями, флагами и манипуляторами состоит в способе их применения.

Основные форматирующие функции:

```
cout.fill(/*символ заполнитель*/)
```

```
cout.width(/*ширина поля*/)
```

```
cout.setf(ios::/*имя флага*/)
```

```
cout.unsetf(ios::/*имя флага*/)
```

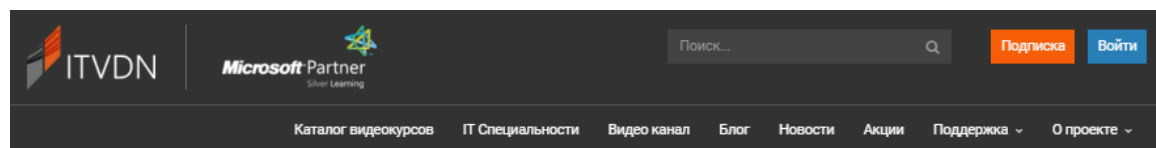
C++ Starter

Форматированный вывод

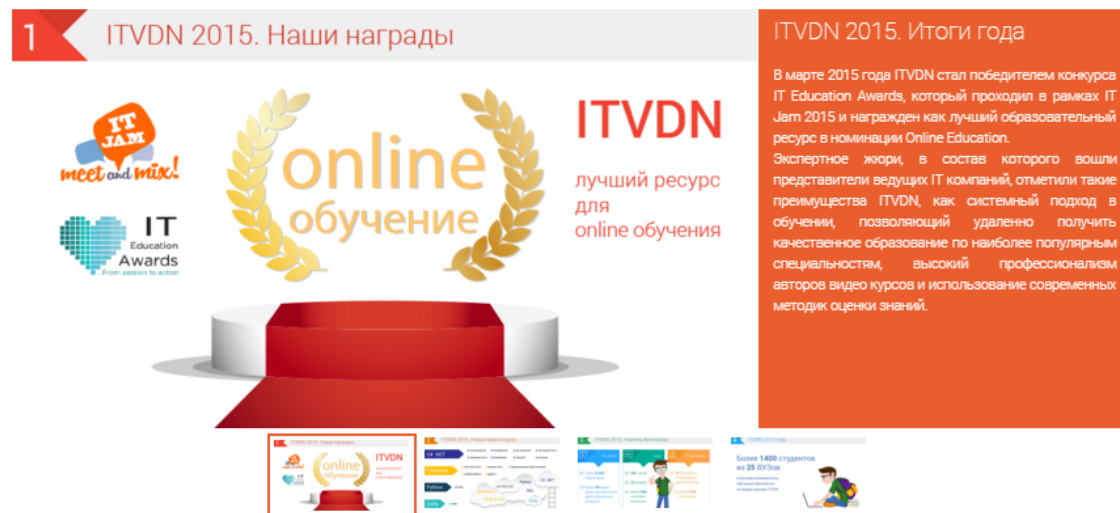
Манипулятор – объект особого типа, который управляет потоками ввода\вывода, для форматирования передаваемой в поток информации. Манипуляторы дополняют функционал, для форматирования ввода\вывода. Большинство манипуляторов выполняют тоже самое, что и функции с флагами форматирования. В некоторых случаях проще использовать функции с флагами форматирования, а иногда использовать манипуляторы. В C++ предусмотрено несколько средств форматирования.

Смотрите наши уроки в видео формате

ITVDN.com



Посмотрите этот урок в видео формате на образовательном портале [ITVDN.com](http://itvdn.com) для закрепления пройденного материала.



Все курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics

Новые видео

Исключения	0
Итераторы и генераторы	0

Популярные видео курсы

Видео курс C# Стартовый (для начинающих)	9 уроков (16 ч. 3 мин.)
Видео курс по шаблонам проектирования	29 уроков (16 ч. 7 мин.)

Теги

.NET Developer
Frontend Developer



Проверка знаний

TestProvider.com

TestProvider | Мы помогаем людям оценить себя

Регистрация Войти

Главная Каталог Сертификация Microsoft Поддержка О нас

Тестирование

Языки программирования и информационные технологии

Microsoft

C# ASP.NET MVC JavaScript Patterns Of Design SQL Architecture Guide WCF HTML&CSS XML SEO WPF HTML5&CSS3 JQuery XNA SharePoint GUI for Android Windows Azure Platform Microsoft Patterns&Practices TFS SCRUM ReSharper TDD WWF LINQ Entity Framework Windows Forms Refactoring Microsoft Expression Blend 4 Windows Phone 8 Windows 8 AppStore Visual Studio Tips&Tricks MSF MEF SilverLight AJAX MEF Service Oriented Architecture

Пройти тест

Наши партнеры

Microsoft Partner CyberBionic ITVDN PROMETRIC TEST CENTER PEARSON VUE Authorized Test Center Windows Azure Cloud Partner EBA

Дополнительные ресурсы:

Очное обучение On-line обучение Видео обучение

TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](https://testprovider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



C++ Starter

Q&A

Информационный видеосервис для разработчиков программного обеспечения

