

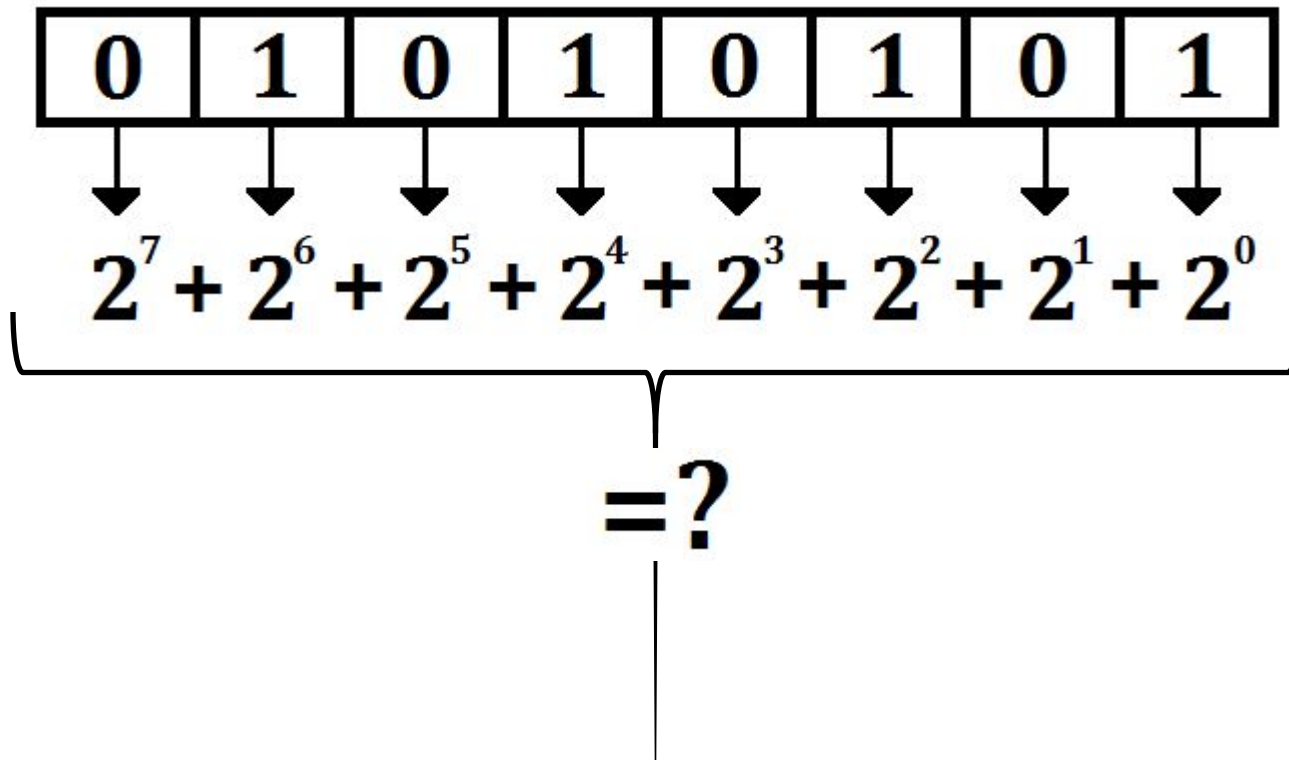
Язык программирования С.

Типы данных. Приведение типов.

ПРЕДСТАВЛЕНИЕ ДАННЫХ В ПАМЯТИ КОМПЬЮТЕРА.

Информация в памяти компьютера хранится в **двоичном коде**.

Двоичный код - это способ представления данных в одном разряде в виде комбинации двух знаков, обычно обозначаемых цифрами 0 и 1.



Язык программирования С.

Типы данных. Приведение типов.

ЧТО ТАКОЕ ТИП ДАННЫХ?

Тип данных определяет:

- 1) **внутреннее представление данных** в памяти компьютера;
- 2) **множество значений**, которые могут принимать величины этого типа;
- 3) **операции и функции**, которые можно применять к величинам этого типа.

ТИПЫ ДАННЫХ

Простые:

- ✓ символьные
- ✓ целые
- ✓ вещественные
- ✓ перечисляемый тип
- ✓ указатели

Составные:

- ✓ массив
- ✓ структура
- ✓ объединение

Язык программирования С.

Типы данных. Приведение типов.

СИМВОЛЬНЫЙ ТИП.

Символьный тип (**char**)

Размер памяти: не менее 8 бит = 1 байт.

Диапазон значений: от -128 до 127.

Интерпретация: 8-ми битное целое число со знаком – код символа в таблице ASCII.

Язык программирования С.

Типы данных. Приведение типов.

ЦЕЛЫЕ ТИПЫ.

short int (допустима аббревиатура **short**) – *короткий целый*

Размер памяти: не менее 16 бит = 2 байта.

Диапазон значений: от –32768 до 32767.

int - *целый*

Размер памяти: не менее 16 бит = 2 байта.

Диапазон значений: от –32768 до 32767.

long - *длинный целый*

Размер памяти: не менее 32 бит = 4 байт.

Диапазон значений: от –2147483648 до 2147483647.

В соответствии со стандартом языка Си:

размер переменной типа **long** должен быть не менее размера переменной типа **int**;

размер переменной типа **int** – не менее размера переменной типа **short**.

Язык программирования С.

Типы данных. Приведение типов.

ЗНАКОВЫЕ И БЕЗЗНАКОВЫЕ ЦЕЛОЧИСЛЕННЫЕ ТИПЫ.

Каждый из **целочисленных типов** может быть определён либо как знаковый (**signed**), либо как беззнаковый (**unsigned**).

По умолчанию – **signed**.

signed

Старший бит внутреннего представления воспринимается как знаковый

unsigned

Старший бит внутреннего представления входит в код представляемого числового значения, которое считается в этом случае беззнаковым

Данный выбор определяет предельные значения, которые можно представить с помощью описанной переменной.

Язык программирования С.

Типы данных. Приведение типов.

ЗНАКОВЫЕ И БЕЗЗНАКОВЫЕ ЦЕЛОЧИСЛЕННЫЕ ТИПЫ.

Данный выбор определяет предельные значения, которые можно представить с помощью описанной переменной.

Диапазоны значений

signed char: от -128 до 127;

signed int: от -32768 до 32767;

signed long: от -2147483648 до
2147483647

unsigned char: от 0 до 255;

unsigned int: от 0 до 65535 ;

unsigned long: от 0 до 4294967295

Язык программирования С.

Типы данных. Приведение типов.

ВЕЩЕСТВЕННЫЕ ТИПЫ.

float – вещественный одинарной точности.

Размер памяти: не менее 32 бита = 4 байта.

Диапазон абсолютных величин: от $3.4E-38$ до $3.4E38$.

double - вещественный удвоенной точности.

Размер памяти: не менее 64 бита = 8 байт.

Диапазон абсолютных величин: от $1.7E-308$ до $1.7E308$.

long double – вещественный максимальной точности.

Размер памяти: не менее 80 бит = 10 байт.

Диапазон абсолютных величин: от $3.4E-4932$ до $3.4E4932$.

По умолчанию всем вещественным константам присваивается тип **double.**

Если вас не устраивает тип, который компилятор приписывает константе, то тип можно указать явно при записи константы с помощью суффиксов: F или f – float; L или l - long.

Например, $3.14159F$ – константа типа float (выделяется 4 байта),
 3.14159 – константа типа double (выделяется 8 байт).

Язык программирования С.

Типы данных. Приведение типов.

ПЕРЕЧИСЛЯЕМЫЙ ТИП.

enum *тип_перечисления* {*список_именованных_констант*}
тип_перечисления – название (необязательный параметр),
список_именованных_констант – разделённая запятыми
последовательность идентификаторов или именованных констант вида:
имя_константы=значение_константы.

ПРИМЕРЫ.

enum DAY {SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY,
FRIDAY, SATURDAY};

Таким образом, SUNDAY = 0, MONDAY = 1 ... SATURDAY = 6.

enum BOOLEAN {NO, YES};

В этом случае NO = 0, YES = 1.

enum BOOLEAN {NO = 1, YES};

В этом случае NO = 1, YES = 2.

Язык программирования С.

Типы данных. Приведение типов.

УКАЗАТЕЛИ.

Указатель – переменная, значением которой является адрес объекта (обычно другой переменной) в памяти компьютера.

Таким образом, если, одна переменная содержит адрес другой переменной, то говорят, что первая переменная указывает (ссылается) на вторую.

Примеры.

unsigned int * a; /* переменная a представляет собой указатель на тип unsigned int (целые числа без знака) */

double * x; /* переменная x указывает на тип данных с плавающей точкой удвоенной точности */

char * fuffer; /* объявляется указатель с именем fuffer, который указывает на переменную типа char */

Язык программирования С.

Типы данных. Приведение типов.

ПРЕОБРАЗОВАНИЕ ТИПОВ.

- преобразования, изменяющие внутреннее представление данных;
- преобразования, изменяющие только интерпретацию внутреннего представления.

ПРИМЕРЫ.

Данные типа **unsigned int** переводятся в тип **int**: менять внутреннее представление не требуется – поменяется только интерпретация.

Данные типа **float** переводятся в значение типа **int**: недостаточно изменить только интерпретацию, необходимо изменить длину участка памяти для внутреннего представления и кодировку. При это возможен выход за диапазон допустимых значений типа **int**.

Язык программирования С.

Типы данных. Приведение типов.

ПРИВЕДЕНИЕ ТИПОВ.

Приведение типа – преобразование значения переменной одного типа в значение другого типа.

Приведение типа

Явное (*тип*) имя

Указывается тип переменной, к которому необходимо преобразовать исходную переменную.

ПРИМЕРЫ

```
int X;  
int Y = 200;  
char C = 30;  
X = (int) C * 10 + Y;  
/*переменная C приведена к типу  
int */
```

Неявное

Преобразование происходит автоматически, по правилам, заложенным в данном языке программирования.

```
int num = 2147483647;  
long bigNum = num;
```

/ компилятор неявно преобразует значение справа в тип **long** перед присвоением его типу bigNum */*

Язык программирования С.

Типы данных. Приведение типов.

ПРАВИЛА ПРЕОБРАЗОВАНИЯ ТИПОВ.

Правила стандартных арифметических преобразований

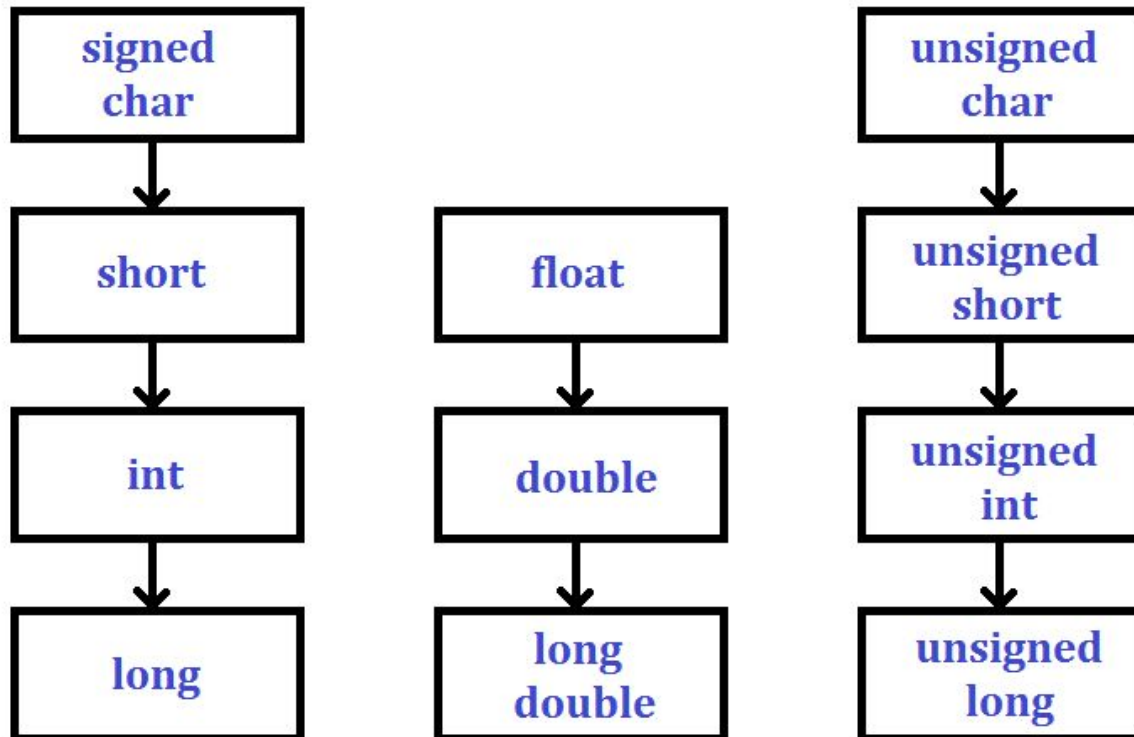
Исходный тип	Преобразованный тип	Правила преобразования
char	int	Расширение нулем или знаком в зависимости от умолчания для char
unsigned char	int	Старший байт заполняется нулем
signed char	int	Расширение знаком
short	int	Сохраняется то же значение
unsigned short	unsigned int	Сохраняется то же значение
enum	int	Сохраняется то же значение

Язык программирования С.

Типы данных. Приведение типов.

ПРАВИЛА ПРЕОБРАЗОВАНИЯ ТИПОВ.

Арифметические преобразования типов, гарантирующие сохранение



Язык программирования С.

Поразрядные логические операции.

ВЫРАЖЕНИЯ С ПОРАЗРЯДНЫМИ ОПЕРАЦИЯМИ.

Поразрядные операции позволяют конструировать выражения, в которых обработка операндов выполняется на битовом уровне (поразрядно).

- ~ - поразрядное отрицание (дополнение или инвертирование битов) (*ранг 2*);
- >> - сдвиг вправо последовательности битов (*ранг 5*);
- << - сдвиг влево последовательности битов (*ранг 5*);
- ^ - поразрядное исключающее ИЛИ (*ранг 9*);
- | - поразрядное ИЛИ (поразрядная дизъюнкция) (*ранг 10*);
- & - поразрядное И (поразрядная конъюнкция) (*ранг 8*).

Язык программирования С.

Поразрядные логические операции.

ОПЕРАЦИЯ ПОРАЗРЯДНОГО ОТРИЦАНИЯ.

Унарная операция, т.е. действует на один операнд, который должен быть целого типа.

a	1	1	1	1	0	0	0	1
~a	0	0	0	0	1	1	1	0

Пример.

```
short a = 2, b; // для a битовое представление 00000010
```

```
b = ~a; // теперь b равно 253, т.е. битовое представление 11111101
```

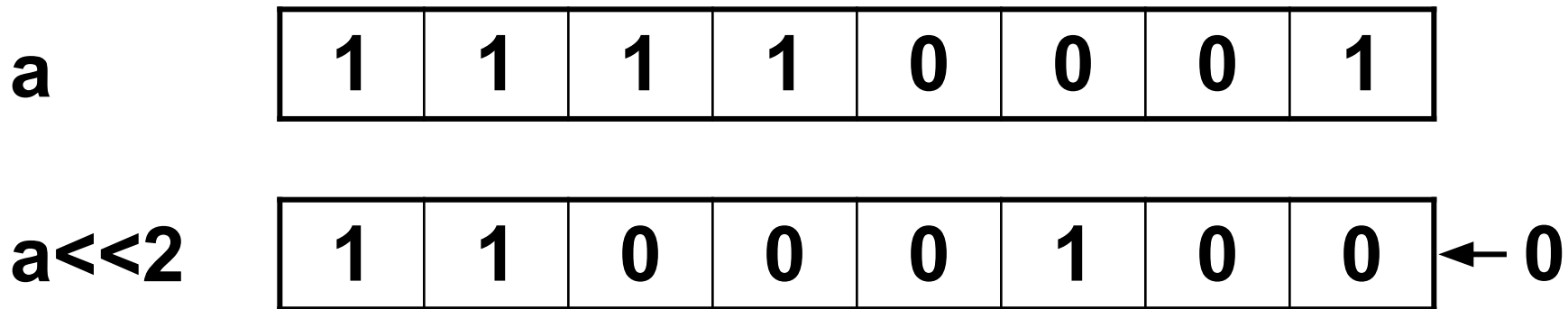
Язык программирования С.

Поразрядные логические операции.

ОПЕРАЦИЯ СДВИГОВ >> (ВПРАВО) И << (ВЛЕВО).

Бинарная операция, т.е. действует на два операнда, которые должны быть целочисленными.

Над битовым представлением значения левого операнда выполняется действие – сдвиг. Правый операнд определяет величину поразрядного сдвига.



Пример.

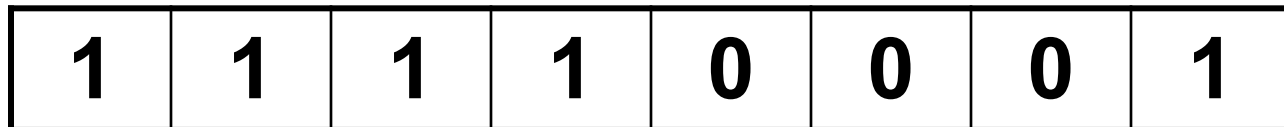
```
short a = 2, b; // для a битовое представление 00000010  
b = a<<2;      // теперь b равно 8, т.е. битовое представление 00001000
```


Язык программирования С.

Поразрядные логические операции.

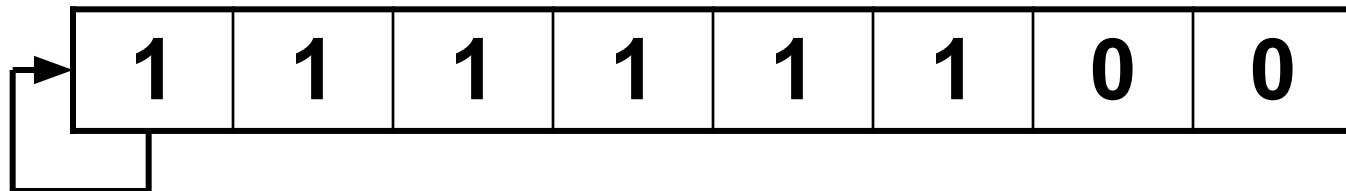
ОПЕРАЦИЯ СДВИГОВ >> (ВПРАВО) И << (ВЛЕВО).

a

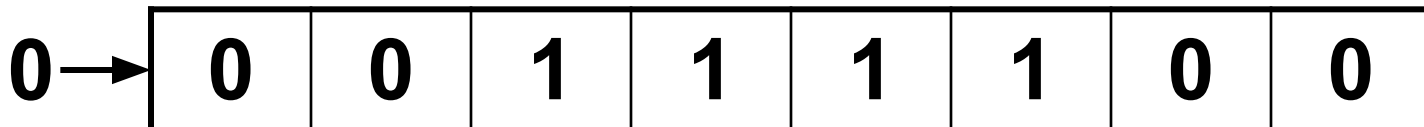


Арифметический сдвиг вправо

a>>2



Логический сдвиг вправо



```
short a = 8, b; // для a битовое представление 00001000
```

```
b = a>>2;      // теперь b равно 2, т.е. битовое представление 00000010
```

Язык программирования С.

Поразрядные логические операции.

ОПЕРАЦИЯ “ПОРАЗРЯДНОЕ ИСКЛЮЧАЮЩЕЕ ИЛИ”

Бинарная операция, т.е. действует на два операнда, которые должны быть целочисленными.

a	1	1	1	1	0	0	0	1
b	1	0	1	1	0	1	1	1
a^b	0	1	0	0	0	1	1	0

Пример.

```
short a = 2, b = 8, c; // для a: 00000010, для b: 00001000  
c = a^b; // теперь c равно 10, т.е. битовое представление  
00001010
```

Язык программирования С.

Поразрядные логические операции.

ОПЕРАЦИЯ ПОРАЗРЯДНАЯ КОНЪЮНКЦИЯ (ПОРАЗРЯДНОЕ И)

Бинарная операция, т.е. действует на два операнда, которые должны быть целочисленными.

a

1	1	1	1	0	0	0	1
---	---	---	---	---	---	---	---

b

1	0	1	1	0	1	1	1
---	---	---	---	---	---	---	---

a&b

1	0	1	1	0	0	0	1
---	---	---	---	---	---	---	---

Пример.

```
short a = 2, b = 10, c; // для a: 00000010, для b: 00001010
```

```
c = a&b; // теперь c равно 2, т.е. битовое представление
```

```
00000010
```

Язык программирования С.

Поразрядные логические операции.

ОПЕРАЦИЯ ПОРАЗРЯДНАЯ ДИЗЪЮНКЦИЯ (ПОРАЗРЯДНОЕ ИЛИ)

Бинарная операция, т.е. действует на два операнда, которые должны быть целочисленными.

a

1	1	1	1	0	0	0	1
---	---	---	---	---	---	---	---

b

1	0	1	1	0	1	1	1
---	---	---	---	---	---	---	---

a|b

1	1	1	1	0	1	1	1
---	---	---	---	---	---	---	---

Пример.

```
short a = 2, b = 10, c; // для a: 00000010, для b: 00001010
```

```
c = a|b;           // теперь c равно 10, т.е. битовое представление
```

```
00001010
```

Язык программирования С.

Поразрядные логические операции.

ОТНОШЕНИЯ И ЛОГИЧЕСКИЕ ВЫРАЖЕНИЯ.

Отношение определяется как пара арифметических выражений, соединенных (разделенных) знаком операции отношения.

Знаки операций отношения:

== равно; **!=** не равно;
< меньше, чем; **<=** меньше или равно;
> больше, чем; **>=** больше или равно.

Примеры отношений:

$a - b > 6.3$

$(y + 5) / 5 == 13$

$7 >= 3$

Логический тип в языке Си отсутствует, поэтому принято, что отношение имеет **ненулевое значение** (обычно 1), если оно **истинно**, и равно **0**, если оно **ложно**.

Язык программирования С.

Поразрядные логические операции.

ОТНОШЕНИЯ И ЛОГИЧЕСКИЕ ВЫРАЖЕНИЯ.

Логических операций в языке Си три:

! Отрицание, т.е. логическое НЕ;

&& конъюнкция, т.е. логическое И;

|| дизъюнкция, т.е. логическое ИЛИ.

Как правило, логические операции применяются к отношениям.