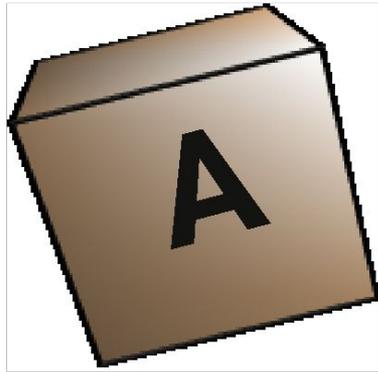
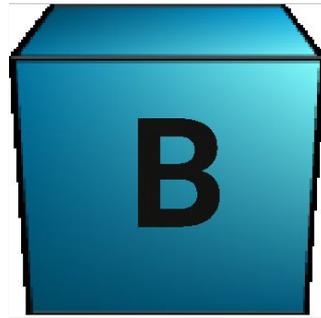


ОСНОВЫ СИНТАКСИСА ЯЗЫКА С



Именованние имен и переменных

- Имя состоит из:
 - латинских букв;
 - цифр;
 - символа подчёркивания _

- Имя не может начинаться с цифры

- Заглавные и строчные буквы различаются

- Серьёзных ограничений на длину нет

- Содержательные имена

value

m12

Value

Place и *place*

longVariableName

Property

q

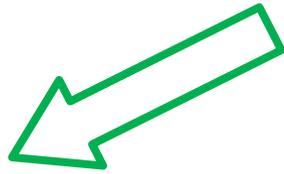
item1

item2

firstItem

seconditem

Типы данных

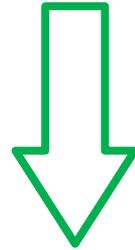


Целочисленные - знаковые

<code>char</code>	1 байт
<code>short</code>	2 байта
<code>int</code>	4 байта
<code>long</code>	4 байта
<code>long long</code>	8 байт

- беззнаковые

<code>unsigned char</code>	1 байт
<code>unsigned short</code>	2 байта
<code>unsigned int</code>	4 байта
<code>unsigned long</code>	4 байта
<code>unsigned long long</code>	8 байт



Вещественные

<code>float</code>	4 байта
<code>double</code>	8 байта
<code>long double</code>	8-12 байт



Специальные

<code>void</code>	1 байт
-------------------	--------

Символьные

<code>char</code>	1 байт
-------------------	--------

Типы данных

□ Базовые типы данных:

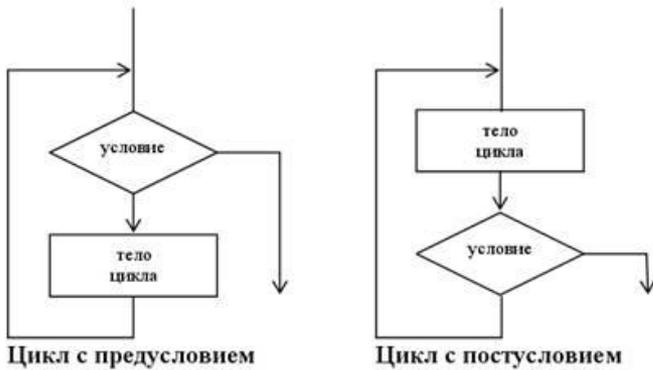
- **int** — целое со знаком
- **float** — число с плавающей точкой одинарной точности
- **bool** — булевские значение (истина или ложь: **true** или **false**)
- **char** — целое однобайтовое значение (символьное)
- **double** - число с плавающей точкой повышенной точности
- **unsigned** — беззнаковое целое
- **long** — знаковое целое
- **void** — пустое множество допустимых значений

Спецификатор ввода-вывода	Тип данных
<code>%d</code>	Десятичное целое число со знаком
<code>%f</code>	Десятичное число с плавающей точкой
<code>%8.3f</code>	Десятичное число с плавающей точкой: 8.3 – модификатор формата, под вещественное число выделяем 8 символов, причем из них 3 знака после запятой.
<code>%c</code>	Символ типа <code>char</code>
<code>%lf</code>	Десятичное число с плавающей точкой двойной точности (<code>double</code>)
<code>%x</code>	Шестнадцатеричное целое число без знака
<code>%s</code>	Строка символов (<code>string</code>)
<code>%u</code>	Десятичное целое число без знака (<code>unsigned</code>)

Стандартная точность для `float` – 6 знаков после запятой

Как выбрать тип данных?

Целочисленные



Вещественные



Логические

(A > B)
C && D
A&&D
H || K
C&&F

Символьные



Как выбрать тип данных?

	со знаком	без знака (unsigned)
char	-128 : 127	0 : 255
int	-32768 : 32767	0 : 65535
long	$-2^{31} : 2^{31} - 1$	$0 : 2^{32} - 1$
float	$\pm 10^{38}$	
double	$\pm 10^{\pm 308}$	

Определение переменных

- Определение переменной

```
тип_переменной имя_переменной;
```

```
int counter;  
double summa;
```

- Объявление переменной

```
extern тип_переменной имя_переменной;
```

```
extern int counter;  
extern double summa;
```

- Объявление с инициализацией

```
тип_переменной имя_переменной = значение;
```

```
int counter = 1;  
double summa = 0;
```

Переменная определяется непосредственно перед первым использованием с инициализацией

Правила формирования имен переменных

- Должны начинаться с латинской буквы или знака подчеркивания ‘_’;
- Являются последовательностью букв, цифр, или ‘_’;
- Пробелы не могут присутствовать в именах;
- Различаются прописные и строчные буквы;
- Нельзя использовать зарезервированные слова (if, else, while и т.д.);
- Желательно делать имена осмысленными.

Операции над переменными.

Приоритет и порядок вычислений

Высший приоритет:

- `()` - вызов функции (при использовании функций);
- `[]` - выбор элемента массива (при использовании массивов);
- `->` - косвенный выбор элемента (при использовании указателей);
- `.` - прямой выбор элемента (при использовании структуры).

Унарные операции

- ! логическое отрицание (NOT).
- ~ побитовая инверсия
- унарный минус.
- ++ увеличение на единицу.
- уменьшение на единицу.
- sizeof ()** размер операнда в байтах.

Математические операции

* умножение

/ деление

% вычисление остатка от деления

+ бинарный плюс

- бинарный минус

>> побитовый сдвиг вправо

<< побитовый сдвиг влево

Операции сравнения

<	меньше
<=	меньше или равно
>	больше
>=	больше или равно
==	равно
!=	не равно

Результатом выражений является истина (**true**) или ложь (**false**)

Операции присваивания

$=$	простое присваивание
$*=$	присвоить произведение
$/=$	присвоить частное
$\%=$	присвоить остаток от деления
$+=$	присвоить сумму
$-=$	присвоить разность
$\ll=$	присвоить сдвинутое влево
$\gg=$	присвоить сдвинутое вправо

Логические операции

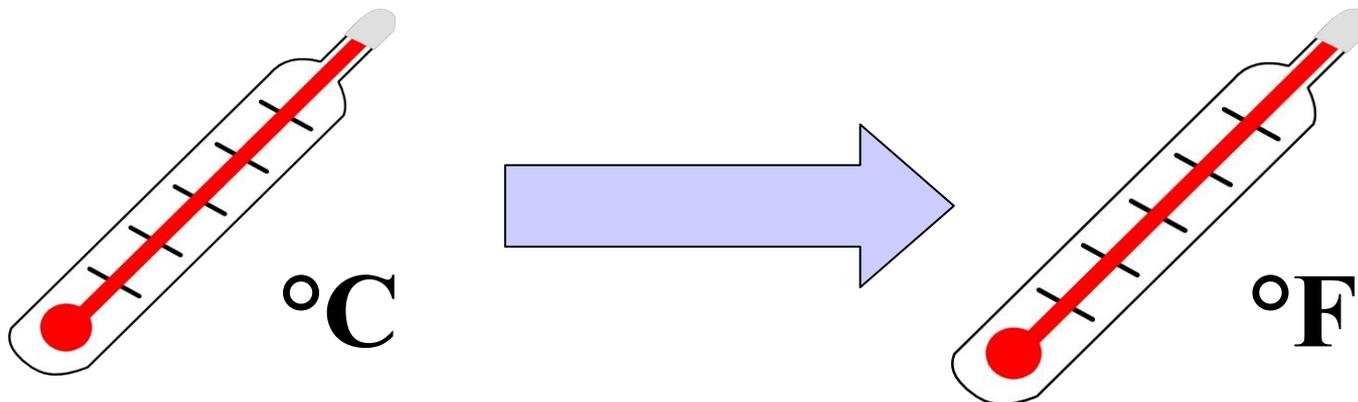
&	побитовое И (AND)
^	побитовое исключающее ИЛИ (XOR)
 	побитовое ИЛИ (OR)
&&	логическое И (AND)
 	логическое ИЛИ (OR)

выражение1 ? выражение2 : выражение3
условное выражение (тернарная операция)

/ последовательное вычисление

Постановка задачи

- Разработать программу перевода температуры из градусов по Цельсию в градусы по Фаренгейту
- Исходные данные (стандартный поток ввода):
 - значение температуры в градусах по ЦельсиюПреобразование выполняется по известной формуле:
- Вычисленное значение должно выдаваться на стандартное устройство вывода (на дисплей).



Постановка задачи

//temperature.c

//Перевод из Цельсий в Фаренгейты

Постановка задачи

```
//temperature.c
```

```
//Перевод из Цельсий в Фаренгейты
```

```
#include <stdio.h>
```

Постановка задачи

```
//temperature.c
```

```
//Перевод из Цельсий в Фаренгейты
```

```
#include <stdio.h>
```

```
int main()
```

```
{
```

Постановка задачи

```
//temperature.c
```

```
//Перевод из Цельсий в Фаренгейты
```

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("Enter the temperature in Celsius: ");
```

Постановка задачи

```
//temperature.c
```

```
//Перевод из Цельсий в Фаренгейты
```

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("Enter the temperature in Celsius: ");
```

```
    double cent;
```

Постановка задачи

```
//temperature.c
//Перевод из Цельсий в Фаренгейты
#include <stdio.h>

int main()
{
    printf("Enter the temperature in Celsius: ");
    double cent;
    scanf("%lf", &cent);
```

Постановка задачи

```
//temperature.c
```

```
//Перевод из Цельсий в Фаренгейты
```

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("Enter the temperature in Celsius: ");
```

```
    double cent;
```

```
    scanf("%lf", &cent);
```

```
    double fahr = (cent * 9.0) / 5.0 + 32.0;
```

Постановка задачи

```
//temperature.c
//Перевод из Цельсий в Фаренгейты
#include <stdio.h>

int main()
{
    printf("Enter the temperature in Celsius: ");
    double cent;
    scanf("%lf", &cent);
    double fahr = (cent * 9.0) / 5.0 + 32.0;

    printf("Fahrenheit it is: %if degree\n", fahr);
```

Постановка задачи

```
//temperature.c
//Перевод из Цельсий в Фаренгейты
#include <stdio.h>

int main()
{
    printf("Enter the temperature in Celsius: ");
    double cent;
    scanf("%lf", &cent);
    double fahr = (cent * 9.0) / 5.0 + 32.0;

    printf("Fahrenheit it is: %lf degree\n", fahr);
    return 0;
}
```

Пример выполнения

Enter the temperature in Celsius: 25
Fahrenheit it is: 77 degree

