

# ОСНОВЫ СИ;

ЧУТКА ПОЛЕЗНОЙ ТЕОРИИ



robo1474

# Обзор



- ▶ Кто: Деннис Ричи; Когда: 1970;
- ▶ Язык среднего уровня;

Высокий уровень	Средний уровень	Нижний уровень
Ада	C	Ассемблер
Модула-2	C++	
Паскаль	FORTH	
COBOL	Макро Ассемблер	
Фортран		
Бейсик		

- ▶ +: Переносим, работа с памятью (типы данных) и битами, легкость и логичность понимания;
- ▶ -: Сложность, отсутствие выраженной объектной ориентации;
- ▶ Структурированность;

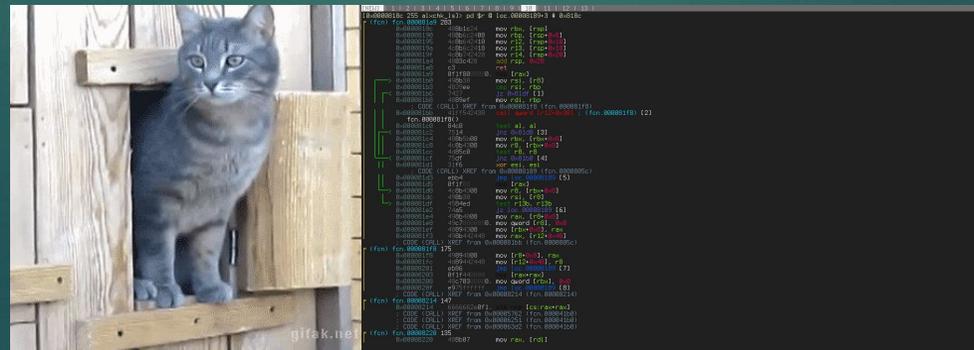
# Компилятор VS Интерпретатор

- ▶ Это способы выполнения программы



Интерпретатор – выполнение кода построчно

- ▶ Компилятор – перевод всего программного кода в машинный
- ▶ После компиляции программный код не влияет на работу программы



# ТИПЫ ДАННЫХ

Тип	Длина в битах	Диапазон
char	8	от -128 до 127
unsigned char	8	от 0 до 255
signed char	8	от -128 до 127
int	16	от -32768 до 32767
unsigned int	16	от 0 до 65535
signed int	16	от -32768 до 32767
short int	16	от -32768 до 32767
unsigned short int	16	от 0 до 65535
signed short int	16	от -32768 до 32767
long int	32	от -2147483648 до 2147483647
unsigned long int	32	от 0 до 4294967295
signed long int.	32	от -2147483648 до 2147483647
float	32	от 3.4e-38 до 3.4e+38
double	64	от 1.7e-308 до 1.7e+308
long double	80	от 3.4e-4932 до 1.1e+4932

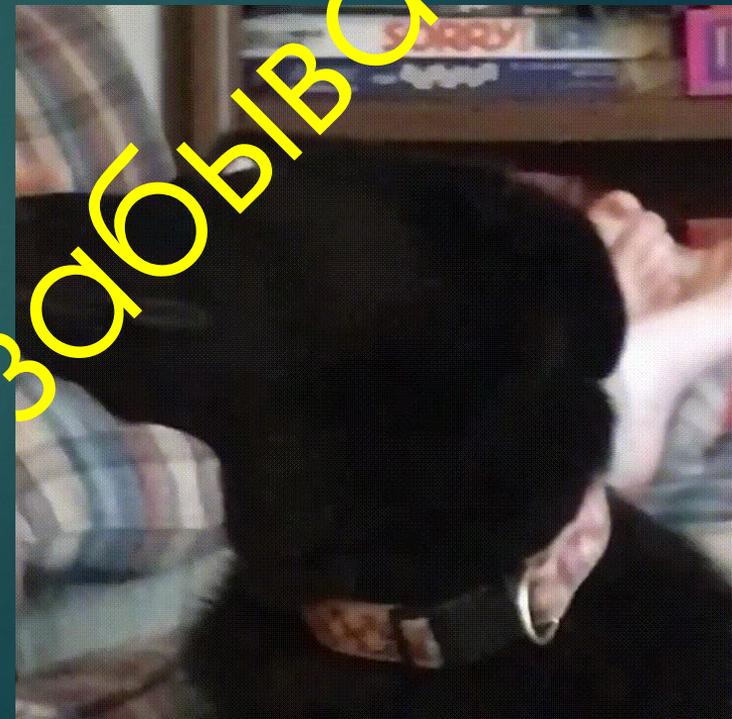


int(целые) = +-32768  
float(с точкой) = +-много  
double(с точкой) = +очмного

# Вид программы на Си;

- ▶ `#include <stdio.h>` //стандартная библиотека ввода-вывода
- ▶ `void main(){`
- ▶ `int a,c;` //инициализация переменных типа Integer
- ▶ `const int b = 10;` //инициализация константы b
- ▶ `printf("введите a");`
- ▶ `scanf("%d", &a);` //ввод a
- ▶ `c = a + b;`
- ▶ `printf("c =%d\n", c);` //вывод c
- ▶ `}`

!!!НЕ ЗАБЫВАЙТЕ!!!



# Полезности

- ▶ `x = y = z = 0;` //многочисленное присваивание
- ▶ `int lenghtOfCatTail = 100;` //название должно передавать смысл
- ▶ `printf("тестовая строка\n");` //"`\n`" перенос строки в консоли
- ▶ `X += 1;` //тоже самое что и `x = x + 1;`
- ▶ `printf("%d", x/y);` //краткость – сестра таланта, выполняй несложные операции внутри операторов



# Операторы



## ▶ Математические

```
const int c = 2;
```

```
int x = 1;
```

```
x = x + c; //x = 3
```

```
x = x%c; // "%" остаток , x = 1
```



# ▶ ++/-- Инкремент/Декремент

```
x = 10;
```

```
y = ++ x; //x = 10, y = 10;
```

```
a = 10;
```

```
b = a ++; //b = 10, a = 11;
```



- IF, IF ELSE, ?



```
x = 10;
```

```
if (x > 9) y = 100;
```

```
else y = 200;
```

```
x = 10;
```

```
y = x > 9 ? 100 : 200; //x>9? Да y=100, Нет y=200
```





Спасибо за

внимание и понимание

