

## Как все работает?

- Принцип действия большинства современных устройств состоит во взаимодействии двух сред
   аппаратной и цифровой:
- Первая это набор микросхем, датчиков и других физических элементов, которые необходимы для корректного функционирования устройства.

Вторая - набор инструкций(программного кода) для управления процессом работы.



### Программирование

 Процесс написания упорядоченного набора инструкций для аппаратуры устройства. Именно они указывают процессору и всем связанным с ним деталям, какие вычисления им необходимо производить и в какой последовательности.

```
document.getElementByld
                                               eschimental, - Georgianding "Serial - V
253
154
       ### function updatePhotoDescription() [
              (descriptions.length > (page * 5) + (currentimage substrained)
  257
                document.getElementByld(
  258
   259
   360
          function updateAllImages() {
    261
     262
                 vari = 1;
                while (i < 10)
     263
                    var elementId = 'foto' + i;
      264
                     var elementldBig = 'biglmage' + i;
      265
                     if (page * 9 + i - 1 < \text{photos.length})
                         document.getElementByld( elementId ) suc = 1
      266
                         document.getElementByld(elementldBig).scs
       267
        268
                          document.getElementByld( elementId ) sic = "
                         else {
         269
         270
```

## Работа компьютера



### Виды программирования

- Низкоуровневое близкое к программированию непосредственно в машинных кодах используемого реального или виртуального процессора.
- Высокоуровневое основная черта это абстракция, то есть введение смысловых конструкций, кратко описывающих такие структуры данных и операции над ними, описания которых на машинном коде очень длинны и сложны для понимания.

## Программа "Hello world"

#### Низкоуровневое

```
. MODEL SMALL
.STACK 100h
. DATA
    HelloMessage DB 'Hello World', 13, 10, '$'
. CODE
START:
    mov ax,@data
    mov ds, ax
    mov ah,9
    mov dx, OFFSET HelloMessage
    int 21h
    mov ah, 4ch
    int 21h
END START
```

#### Высокоуровневое

```
#include <stdio.h>
void main() {
    printf("Hello, World!\n");
}
```

## Как работают программы

- 1. Помещение приложения в оперативную память.
- 2. Передача управления приложению (точка входа).
- 3. Выполнение инструкций приложения.

```
#include <stdio.h>
void main() {
    printf("Hello, World!\n");
}
```

## Вывод данных - Функция printf

- Функция для вывода информации на стандартный вывод(напр. консоль). 0
- Для работы необходимо подключить библиотеку стандартного ввода/вывода(«stdio.h»). 0

```
#include <stdio.h>
void main() {
    printf("Hello, World!\n");
```

## Как хранятся данные?

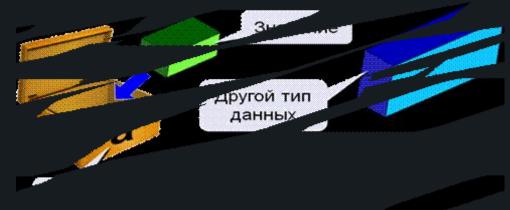
- Двоичная система счисления В двоичной системе счисления числа записываются с 0 помощью двух символов (0 и 1).
- В n битах может содержаться 2<sup>n</sup> значений. 0
- Сколько значений в 8 битах?

# Типы данных

Тип	Размер в байтах	Диапазон значений
char	1	от – 128 до 126
unsigned char	1	от 0 до 255
short	2	от -32 768 до 32 767
unsigned short	2	от 0 до 65536
enum	2	от -2 147 483 648 до 2 147 483 647
long	4	от -2 147 483 648 до 2 147 483 647
unsigned long	4	от 0 до 4 294 967 295
int	4	от -2 147 483 648 до 2 147 483 647
unsigned int	4	от 0 до 4 294 967 295
float	4	от 3,4 * 10 -38 до 3,4 * 10 38
double	8	от 1,7 * 10 <sup>-308</sup> до 1,7 * 10 <sup>308</sup>
long double	10	от 3,4 * <sup>-4932</sup> до 1,1 * 10 <sup>4932</sup>
bool	1	true или false

# Переменная

- О Именованная область памяти с данными.
- В высокоуровневых языках имеет размер в зависимости от типа хранимых данных.



## Переменная в Си

```
Объявление: Тип Имя = Значение;
0
```

Пример: int a = 2; 0

Над числовыми типами данных доступны основные арифметические  $\frac{1}{1}$ 

операции (+, -, \*, /)

```
int b = 2;
int c = 0;
c = a + b;
c = a - b;
c = a / b;
```

### Вывод данных - Функция printf

- Для вывода чисел используется форматированный вывод. В строку вставляется идентификатор параметра, а последующими параметрами идут сами значения.
- %d целое число.

```
int x = 2;
int y = x * x + x + 5;
printf("Result = %d", y);
```

```
int x = 2;
int y = x * x + x + 5;
printf("x = %d, y = %d", x, y);
```

# Практика

○ Посчитать выражение. x\*x + x + 5

## Домашнее задание

Посчитать выражение. 5 \* x - (x - 5)0

## Что установить?

- <u>http://www.jetbrains.com/clion/</u> CLion от JetBrains
- https://www.visualstudio.com/ru/vs/community/
   Visual Studio от Microsoft