

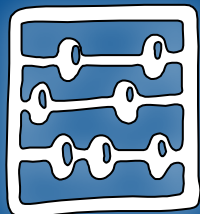


# Алгоритмы и Алгоритмические Языки

Семинар #1:

1. Работа по системам счисления;
2. Минимальная программа на Языке Си;
3. Переменные, типы данных и арифметика;
4. Написание полезной программы.

# Самостоятельная работа по системам счисления





# Минимальная программа на языке Си



# Программа "Hello, World!"



```
1  #include <stdio.h>
2
3  int main(void)
4  {
5      printf("Hello, world!\n");
6
7      return 0;
8  }
9
```

# Сборка минимальной программы



Сборка примера кода в консоли:

```
> cd ~/path/to/directory // Поиск файла с кодом
> ls // Отображение файлов в директории
hello.c
> cc hello.c // cc – это “C compiler”, есть на UNIX-системах
> ls
a.out hello.c
> rm a.out // Название – жуть :(
> cc hello.c -o hello // “-o” – флаг компиляции, output-файл
> ./hello
Hello, world!
```

# Подключение стандартных библиотек



Препроцессор – первый этап в процессе компиляции программы.

Флаг **-E** – указание компилятору произвести только первый этап компиляции, на котором обрабатываются **#include-ы**

```
> cc -E hello.c -o hello.i // Только препроцессинг
> ls
hello hello.c hello.i
> cat hello.i
# 1 "hello.c"
# 1 "<built-in>"
# 1 "<command-line>"
...
```

# Подключение стандартных библиотек



Функция определяется (и реализуется) внутри библиотеки:

```
410  
411 int printf(const char *__format, ...);  
412
```

Вызов (применение) функции – в вашей программе:

```
729 int main(void)  
730 {  
731     printf("Hello, world!\n");  
732  
733     return 0;  
734 }
```

Подключение библиотеки – это copy+paste её кода к вашему!

# Сообщения об ошибках компиляции



```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     printf("Hello, world!\n");
6
7     return "Pure evil!";
8 }
```

Что здесь может не понравиться компилятору?



# Сообщения об ошибках компиляции



Реакция компилятора:

```
> cc hello.c -o hello
hello.c: In function 'main':
hello.c:7:12: warning: returning 'char *' from a function with
return type 'int' makes integer from pointer without a cast
[-Wint-conversion]
    7 |     return "Pure evil!";
      |           ^~~~~~
```

**Warning:** компилятор нашёл вашу ошибку и выдал сообщение об ошибке.

# Сообщения об ошибках компиляции



Реакция компилятора:

```
> cc hello.c -Werror -o hello
hello.c: In function 'main':
hello.c:7:12: error: returning 'char *' from a function with
return type 'int' makes integer from pointer without a cast
[-Werror=int-conversion]
    7 |     return "Pure evil!";
      |           ^~~~~~
cc1: all warnings being treated as errors
```

**Error:** компилятор нашёл вашу ошибку и не будет компилировать программу дальше! Это хорошо/плохо?



# Переменные, типы данных и арифметика



# Пример использования арифметики



```
#include <stdio.h>

const unsigned DOLLAR_START  = 0;
const unsigned DOLLAR_FINISH = 1000;
const unsigned DOLLAR_STEP   = 100;

const float RUBLES_IN_DOLLAR = 97.8439;

int main(void)
{
    for (unsigned num_dollars = DOLLAR_START;
         num_dollars <= DOLLAR_FINISH;
         num_dollars += DOLLAR_STEP)
    {
        float num_rubles = RUBLES_IN_DOLLAR * num_dollars;

        printf("%4d USD = %9.03f RUB\n", num_dollars, num_rubles);
    }

    return 0;
}
```

# Типы данных языка Си



## Целочисленные типы данных:

- Знаковые/беззнаковые
- Бывают разных размеров
- Удовлетворяют неравенству на размеры

long int	at least <b>32</b>
unsigned long int	
long long int (C99)	at least <b>64</b>
unsigned long long int (C99)	

char	at least <b>8</b>
signed char	
unsigned char	
short int	at least <b>16</b>
unsigned short int	
int	at least <b>16</b>
unsigned int	

Besides the minimal bit counts, the C Standard guarantees that

`1 == sizeof(char) ≤ sizeof(short) ≤ sizeof(int) ≤ sizeof(long) ≤ sizeof(long long)`.



# Типы данных языка C++

Типы данных с плавающей точкой:

- **float** – 32 бита
- **double** – 64 бита
- **long double** – 128 бит

Строковый тип данных: **const char\***

Булевый тип данных (значения – true/false): **bool**

Тип-пустышка: **void**



# Написание полезной программы





## Промежуточная аттестация

- 2022/2023 учебный год 1 модуль

$$П1 = 1/8*(C1 + C2) + 1/4*K1 + 1/6*(Д31 + Д32 + Д33)$$

- 2022/2023 учебный год 2 модуль

$$П2 = 4/7*Э + 3/28*(K2 + K3) + 1/14*(Д34+Д35+Д36)$$





# Вычисление оценки по ПУДу


Что точно стоит сделать:

1. Компиляция helloworld в вашей IDE или с помощью консоли.
2. Загрузка актуальных материалов курса с Github.

 Code ▾

Что можно сделать:

1. Поиск расчётной формулы оценки по курсу;
2. Задание своих оценок с помощью `const`;
3. Произведение всех арифметических операций, заданных в ПУД.

 Download ZIP

# Это никак не оценивается!



# Источники:

## Источники:

- [EASY] Книга “Язык Си”, Б.В.Керниган, Д.М.Ритчи, части 1.1-1.3;
- [HARD] Упрощённое изложение стандарта Си ([cppreference.com](http://cppreference.com));  
В частности – [часть про типы данных](#).
- [Репозиторий с примерами кода и слайдами](#);  
В частности – [эта презентация](#).

# Вопросы?

