

## Домашнее задание - Основы C:

Эти задачи нужно оформить в соответствии с правилами оформления <http://style.vdi.mipt.ru/CodeStyle.html> и прислать мне на почту [vladimir.biryukov@phystech.edu](mailto:vladimir.biryukov@phystech.edu).

### Задача 1 - Продвинутый helloworld:

Вывести на экран строку `!\@#$$%^&%`. Если возникнут вопросы по этой или по другим задачам, то ответы можно найти на [stackoverflow](https://stackoverflow.com). Просто зауглите, например, "how to print backslash c stackoverflow".

### Цикл for:

Цикл `for` - это просто более удобная запись цикла `while`. Например, код, который печатает все числа от 1 до 100 через пробел с помощью циклов `while` и `for`:

```
int i = 0;
while (i < 100)
{
    printf("%d ", i);
    i++;
}
→
for (int i = 0; i < 100; i++)
{
    printf("%d ", i);
}
```

### Задача 2 - Число, квадрат и куб:

Напишите программу, которая будет печатать само число, его квадрат и его куб от 1 до `n`, разделённые стрелочкой. Число `n` считывается с помощью `scanf`. Например, при `n = 5`, программа должна напечатать следующее:

```
1 -> 1 -> 1
2 -> 4 -> 8
3 -> 9 -> 27
4 -> 16 -> 64
5 -> 25 -> 125
```

### Задача 3 - Таблица умножения:

Напишите программу, которая будет печатать таблицу умножения в виде квадратной таблицы (такую же, как печатают на задней стороне тетрадей). Используйте 2 вложенных цикла `for`.

### Целочисленные переменные:

Различные целочисленные типы языка C представлены в следующей таблице:

тип	размер (байт)	диапазон значений ( $2^{\#bits}$ )	спецификатор
char	1	от -128 до 127	%hhd
short	2	от -32768 до 32767	%hd
int	4	примерно от -2-х миллиардов до 2-х миллиардов	%d
long	4 или 8	такой же как у int или long long в зависимости от системы	%ld
long long	8	примерно от $-10^{19}$ до $10^{19}$	%lld
unsigned char	1	от 0 до 255	%hhu
unsigned short	2	от 0 до 65535	%hu
unsigned int	4	примерно от 0 до 4-х миллиардов	%u
unsigned long	4 или 8	такой же как у unsigned int или unsigned long long	%lu
unsigned long long	8	от 0 до $2^{64} \approx 2 * 10^{19}$	%llu
16-ричная система	-	-	%x
указатель	8	$2^{64} \approx 2 * 10^{19}$	%p

#### Задача 4: Арифметика char:

Стандартный тип `char` - это целые числа, размером в 1 байт. Ниже приведён пример программы, в которой используются числа типа `char`. Что напечатает программа? Объясните результат (нужно прислать словесные объяснения по значению каждой переменной).

```
#include <stdio.h>
int main()
{
    char a = 100;
    char b = a + 50;
    char c = 2 * a;
    char d = a * a;
    unsigned char e = 2 * a;
    printf("%hhd %hhd %hhd %hhd %hhu\n", a, b, c, d, e);
}
```

#### Задача 5: Произведение чисел:

Напишите функцию, которая вычисляет произведение 2-х положительных чисел  $a < 2^{32}$  и  $b < 2^{32}$ . Проверьте вашу функцию на следующих значениях:

ВХОД	ВЫХОД
2 2	4
2000000000 2	4000000000
1444444444 777777777	1123456788654320988
4222222222 377777777	15950617279827160494

#### Задача 6: Факториал:

Напишите функцию, которая вычисляет факториал числа  $n \leq 20$  (при больших значениях  $n$  факториал не влезит даже в `unsigned long long`). Проверьте вашу функцию на следующих значениях:

ВХОД	ВЫХОД
0	1
1	1
5	120
10	3628800
20	2432902008176640000

#### Задача 7: Размещения:

В комбинаторике размещением (из  $n$  по  $k$ )  $A_n^k$  называется упорядоченный набор из  $k$  различных элементов из некоторого множества различных  $n$  элементов. Размещения вычисляются следующим образом:  $A_n^k = \frac{n!}{(n-k)!}$ .

Напишите функцию, которая будет вычислять размещения при условии, что  $A_n^k < 2^{64}$ . Проверьте вашу функцию на следующих значениях:

ВХОД	ВЫХОД
5 2	20
20 10	670442572800
30 12	41430393164160000
60 11	13679492361575040000

#### Задача 8\*: Последняя цифра числа Фибоначчи:

Найдите последнюю цифру  $n$ -го числа Фибоначчи.

ВХОД	ВЫХОД	ВХОД	ВЫХОД
1	1	987	8
2	1	1234567	3
6	8	123456789	4
20	5	12345678987654321	9
123	2		