

# Семинар #1: Введение в язык C++ (для тех, кто знает C).

## Классные задачи.

### Hello world++

Пишем первую программу на C++

```
#include <cstdio>
int main()
{
    printf("Hello World++\n");
}
```

Все библиотеки из языка C можно использовать и в языке C++. Только название библиотеки без `.h` на конце и `c` символом в начале:

`<stdio.h>` -----> `<cstdio>`

Для компиляции используйте компилятор `g++`. Вот так:

```
g++ helloworld.cpp
./a.out
```

Или, если вы хотите установить у исполняемого файла своё имя за место `a.out`:

```
g++ -o hello helloworld.cpp
./hello
```

### Тип bool

В прошлом семестре, для хранения результатов логических операций, мы использовали целочисленные типы. В языке C++ есть встроенный тип `bool`, который может принимать 2 значения (`true` и `false`).

```
#include <cstdio>
int main()
{
    bool a = true;
    bool b = false;
    bool c = a || b;

    if (c)
        printf("Yes\n");
    else
        printf("No\n");
}
```

## Пространство имён

```
#include <stdio>
// Определяем переменные, структуры, функции внутри пространства имён foo
namespace foo
{
    int a = 5;
    int square(int x)
    {
        return x * x;
    }
}
// Чтобы получить доступ к ним вне пространства имён, нужно добавить к именам foo::
int main()
{
    printf("%d\n", foo::a);
}
```

## Ссылки

Ссылка – это переменная, которая является новым именем для существующего участка памяти.

```
#include <stdio>

int main()
{
    int a = 10;
    // Создадим ссылку r на переменную a
    int& r = a;
    // Теперь, если изменить r, то поменяется и a
    r += 5;
    printf("%d\n", a);
}
```

Ссылки часто используются для тех же целей, что и указатели (только со ссылкам работать удобнее). В отличии от указателей, ссылки:

1. Должны всегда инициализироваться при создании
2. Не могут никуда не ссылаться (т.е. не могут равняться NULL)
3. Их нельзя переприсвоить. При использовании оператора = со ссылками изменяется та переменная, на которую ссылка ссылается, а не сама ссылка.

## Перегрузка функций

## Перегрузка операторов