# «C++ for C Programmer» Notes
by Firmin Martin

## Week 1

### From C to C++

| C | C++ |
|---|---|
| *#include <stdio.h>* | *#include <cstdio>* |
|  | using namespace std |
| *#define PI 3.14* | const float PI = 3.14 |
| *#define max(a,b) (...)* | inline max(a, b) {...} |

### Simple I/O functions

```
cout << "Print something" << endl
```

```
cin >> input
```

### Cast

**static cast** : (*safe cast*)   `static_cast<double> 5/4`
Convert if there is a rule based
conversion, otherwise error

**reinterpret cast** :

**dynamic cast** : used with
object

**const cast** : cast away
const-ness

### Function call

Call by value

Call by pointer

**Call by reference**

📄 An Introduction to Reference

### </>c++ Overload

```cpp
inline void swap(int &i, int &j) {
    int tmp = i
    i = j
    j = tmp
}

inline void swap(double &i, double &j) {
    double tmp = i
    i = j
    j = tmp
}
```

### </>c++ Generic

```cpp
template<class T>
inline void swap(T &i, T &j) {
    T tmp = i
    i = j
    j = tmp
}
```

## Week 2

### Function default parameter

```cpp
T sum (T arr[], int count, T s = 0)
```

### </>c++ Multiple template arguments

```cpp
template <class T1, class T2>
void copy (const T1 src[], T2 dest[], int size) {

    for (int i = 0; i < size; ++i) {
        dest[i] = static_cast<T2>(src[i]);
    }

}
```