C++ Fundamentals Including C++ 17

UNDERSTANDING C++ IN CONTEXT

Kate Gregory

@gregcons www.gregcons.com/kateblog

In This Course

Context

Tools

Language basics

- Variables: fundamental and user-defined types
- flow of control
- operators

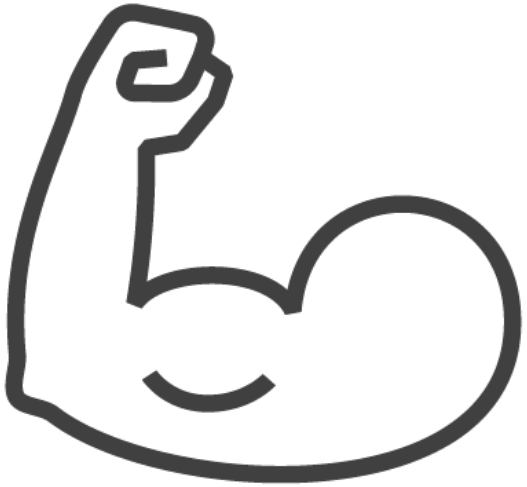
Templates

Indirection

- Pointers
- References
- Polymorphism
- Memory management

Why C++?

C++ is a general purpose language

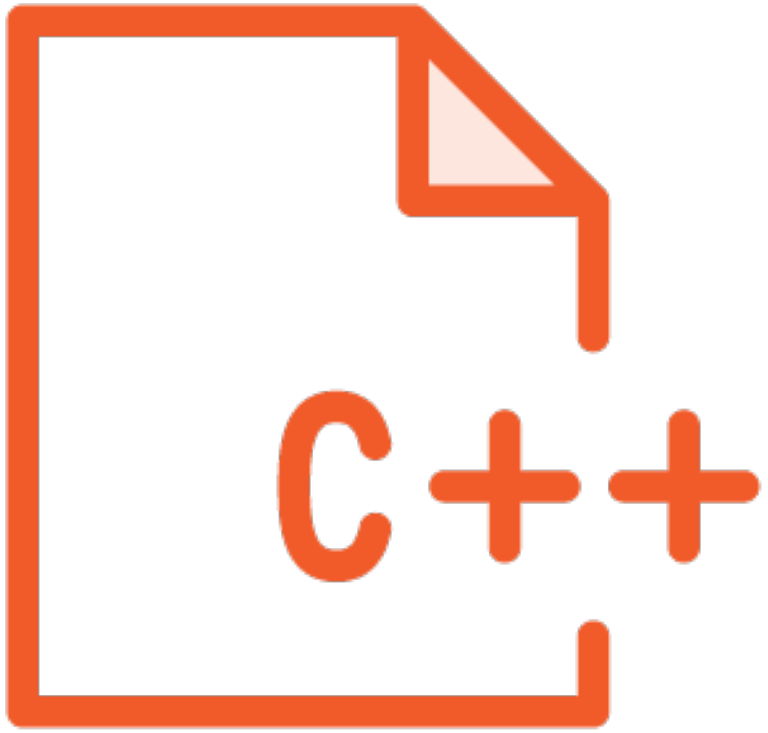


C++ emphasizes power and performance

C++ is popular

Isn’t it really hard?

Modern C++ Is Not Too Hard for Beginners



You do not need to learn C first

- It might even make things harder

If you know C# or Java, you can learn C++ quickly

If you know old school C++, learning modern C++ may be harder

If you are familiar with what a loop is, why you might want to test a condition, and what an object is, you can learn C++

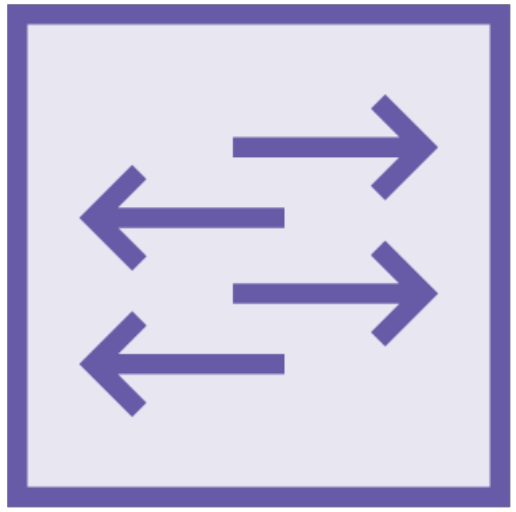
- Just different punctuation

Even if you don't know any programming languages yet, you can learn C++ as a first language

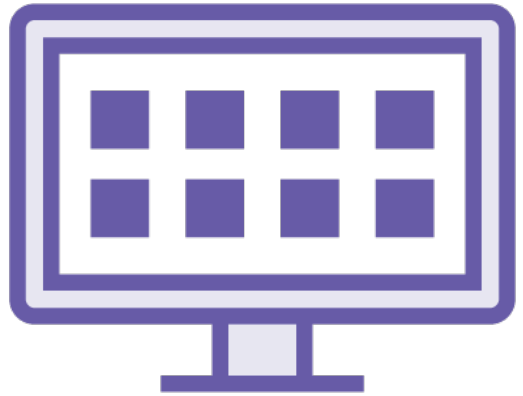
The Standard Library: Built in Capabilities

String class

Collections



Smart pointers



File and Screen IO

Modern C++

Managing memory yourself is "old-school"

So is pointer arithmetic for string or array work

Modern C++ looks a lot like C# or Java

But faster!

Standardization

C++ as a language belongs to no person or company

There is a standard that defines the language

Compiler vendors implement a lot in advance

Different compilers get to 100% at different paces

Summary

C++ is a powerful language used by millions of developers

C++ has a reputation for being difficult

- But it doesn't have to be

Modern C++ is clean and simple

- The Standard Library does a lot of work for you