21.1 — The Standard Library

Congratulations! You made it all the way through the primary portion of the tutorial! In the preceding lessons, we covered many of the principal C++ language features (including a few from the C++11/14/17 extension to the language).

So the obvious question is, "what next?". One thing you've probably noticed is that an awful lot of programs use the same concepts over and over again: loops, strings, arrays, sorting, etc… You've probably also noticed that programs written using non-class versions of containers and common algorithms are error-prone. The good news is that C++ comes with a library that is chock full of reusable classes for you to build programs out of. This library is called The C++ Standard Library.

The Standard Library

The Standard library contains a collection of classes that provide templated containers, algorithms, and iterators. If you need a common class or algorithm, odds are the standard library has it. The upside is that you can take advantage of these classes without having to write and debug the classes yourself, and the standard library does a good job providing reasonably efficient versions of these classes. The downside is that the standard library is complex, and can be a little intimidating since everything is templated.

Fortunately, you can bite off the standard library in tiny pieces, using only what you need from it, and ignore the rest until you're ready to tackle it.

In the next few lessons, we'll take a high-level look at the types of containers, algorithms, and iterators that the standard library provides. Then in subsequent lessons, we'll dig into some of the specific classes.