The Standard C++ Library

Below you'll find the code for the *f2c* program. Use the arrow on the left to show and hide the code as we discuss its various features.

▶ *The f2c Source Code*

Libraries are collections of useful, pre-written components (functions and classes). The **standard library** comes with your C++ compiler. It is divided into a number of "packages", known as **headers** in C++. Here are the headers you'll use most often. You'll meet more as the semester goes on. Find the whole list at cppreference.com.

```
#include <iostream> // input-output for reading and printing
#include <iomanip>  // formatting output
#include <cmath>    // all math functions
#include <string>   // The C++ string class
```

The `#include` directive instructs the **preprocessor** to read the declarations from the **header file** and insert them, exactly as if you had typed them into your source code.

- Instructions to the preprocessor are called **preprocessor directives**; these always appear on a line by themselves, always start with a `#`, and do **not** end in a semicolon.
- Angle brackets indicate a header is a **system library**, part of standard C++.

In *f2c*, you'll find the `#include` statements on lines 7-8.

The Standard Namespace

In C++, libraries are combined into larger groups called **namespaces**. The standard library is in the namespace called *std*, usually pronounced **standard** instead of "es-tee-dee". For CS 150, add a **using directive** to the top of your source code, like this:

```
using namespace std;
```

In *f2c*, you find this on line 9. Think of `using namespace std;` as one more incantation that the C++ compiler requires to work its magic on your code. This only works correctly inside .cpp file. In header files, you must use a different technique, which we'll cover in lecture.