

# COMP220 - Code::Blocks and the StanfordCPPLib

Fall 2015

## 1 Adding the StanfordCPPLib Library

The standard C++ libraries like *iostream* are placed on the server in locations that the compiler and linker know to search. The Google testing libraries are installed in the usual places so the compiler has no problem resolving the include. However, the gTest libraries are not standard C++ libraries so we must explicitly include link options for their compiled objects. When we write our own libraries we use quotes in the include and this directs the compiler to search the current working directory. The linker also has no problem finding the objects for our libraries in the current working directory. So, what do we do when we have a library that is neither placed in the standard locations nor is it in the current working directory? This is the problem we have with the StanfordCPPLib.

It's not uncommon to use libraries that cannot be installed system wide. It's also not uncommon to use those libraries in multiple projects. What we want to avoid is copy the library files into each project. To do this we must:

- Direct the compiler and linker to search in the directories where headers and objects are stored.
- Directly link any pre-compiled library code into the project.

In what follows

### 1.1 Building StanfordCPPLib