Complex Option Processing

Some programs, however, perform more complex option processing—taking a variety of flags and options, some of which require arguments (and in some cases, optional arguments). Going back to ourgccexample, When we write

gcc -o myProgram myProgram.c

the **-o** option itself takes an argument—the next command line argument after it specifies what the output is exactly because **-o** came right before it. However, gcc does not require **-o** to come in this position, we could write other arguments first.

For more complex option processing (such as would be required by a program like gcc), the **getopt** function is quite popular. **getopt** is part of the C library, and parses the command line arguments, accounting for such considerations as options which have (potentially optional) arguments, and short and long name versions (for example, some programs may accept a short name, like **-o** and a long name like **--output**with the same meaning). We will not go into the details of **getopt** here, but should you need it, you can, of course, read about it in its man page (be sure to specify **man -S3 getopt**, as there is a program by the same name in section 1 of the man pages).