CITS5501 Software Testing and Quality Assurance Semester 1, 2020
Syntax-based testing

Command-line programs often take a range of arguments and options.

For instance, we might have a command `delete_file`, that takes a file to delete, and a *flag* (boolean option) stating whether the removal is *recursive* (i.e., if the file is actually a directory, then remove it and all contents).

Documentation will typically show the way the command can be called as follows:

```
1    delete_file [--recursive] FILE_NAME
```

The brackets mean something is optional. A pipe ("|") is used to separate alternatives. Words in capital letters usually mean a string representing, e.g., a file, directory, URL, etc.

If we are testing the program, we can write invocations of the program as following a *grammar*.

Exercise 1

How can we describe the grammar for the `delete_file` program invocations, using the BNF syntax we've seen?

Assume we have a terminal symbol `file_name` we don't have to define.

Exercise 2

(a) Could we write an exhaustive set of tests for this syntax? (b) What sort of coverage would that give us?

(b) What is a set of tests that would give us production coverage?