**Recitation 1**

**Topics**

* data types
* I/O
* loops.
* Submitting your code!

**Task**

We want to read a file, display it to the screen and report how many lines and words there are.

There are several ways to do this. Here's one way:

* Read the file line-by-line, printing each line to the screen.  Count the lines along the way and print the line count at the end.
* Read the file again, word-by-word, keeping a count of the words read and displaying the count when you are done.  Note you need to again start your reading from the *beginning* of the file.  You could do this by first closing the file and then reopening it.

**Opening**

Remember to check that your attempt to open the file worked!  You can find yourself wasting a lot of time trying to debug a *correct* program just because you ran it on an input file that wasn't there.

And what file should you be running your program on?  We have attached a file, *jabberwocky.txt*, for you to use.  Feel free to hardcode the name of file into your program.  Or make your program more general by asking the user what file to use.  We'll leave that part up to you.

**Character by character**

If you have time, try reading the file one more time *character by character*.  Count the characters and print out the result.  For the attached file, I got a count of 797.  You are likely to instead get 648.  Why? Remember that C++'s input operator is implemented to *skip whitespace* by default.  So your program was failing to count 149 whitespace characters, e.g. blanks and newlines.

You can change the behavior of the input operator by first executing the line:

ifs >> noskipws;

After that line, the input operator will not longer skip whitespace. If you want to turn that behavior back on, then execute the line:

ifs >> skipws;

**Functions**

If you have even more time and know how, break the problem into functions. What are good ways to do that? What parameters should you pass to the function? More time? Try counting each distinct "word".

**Issues**

* What file should you read?  So that you can check your counts, I have attached a file *jabberwocky.txt* to this assignment.  I find there are 29 lines, 144 words and 797 characters.  (I also used a unix utility called wc that agreed with my counts.)
* Development environment
  + Many students like to use Microsoft's Visual Studio. It can be downloaded free (legally) from www.dreamspark.com.
  + You are welcome to use whatever development environment that you like, however the TA's may not be familiar with your choice, so don't expect help with your choice of tools.
* How to open a file. And how to check that you were successful.
* How to read a line
* How to read a "word". And what is a "word", anyway?
* How to loop and how to know when you finished reading the file.
* How to read a character.
* How to control skipping whitespace.

**Submit**

* Submit your code (just the source file) before you leave the lab!

#### Additional resources for assignment

* ile attachment [jabberwocky.txt](https://newclasses.nyu.edu/access/content/attachment/d5110493-7072-4a26-b7c9-8e41d1b52934/Assignments/cae484bf-9629-481a-8c43-1564a7c89130/jabberwocky.txt) ( 1 KB; Aug 28, 2015 5:04 pm )