# Assignment 8: Word Length

## Objectives

* Open a file for reading.
* Open a file for writing.
* Use a while loop.

## Motivation

In two of the videos currently on the course website, the procedure on how to open a file for both reading and writing is done. In this assignment, you should prompt the user for a filename to read and a filename to write. You should then read from the file for reading (an example will be provided along with this assignment) and then you should write to the file for writing these contents:

* a word
* that word's length

Each word and word length should be printed on a line all by itself.

## Tips.

The libraries used to open a file for reading are:

* File
* Scanner

The libraries used to open a file for writing are:

* FileWriter
* PrintWriter

Please watch the videos and read Chapter 4 for a detailed look at how to use the libraries. The libraries also require that you throw certain exceptions. The NetBeans "Fix Code" feature will properly throw these errors in your code. You are always encouraged to learn the features of NetBeans to make your programming life easier.

Once you begin reading from the input file, you'll have to **loop while** (hint, hint) there is another token in the file to read. The method **hasNext** will detect if there is another token to read in a file. See the example in the book on page 241 and 242. You'll read in the book's example about how to read an entire line using **nextLine**. I don't want you to do that for this assignment. I want you to read in a single word. This can be accomplished using the **next** method. The **next** method returns a **String** object.

Once you have your word, you need to print that word and the word's length. You can determine the length of a **String** object using "word.length()".

## Input

The input file provided in this assignment is the text of the Gettysburg Address. The file contains only lowercase letters and one space per word and no punctuation. There are several lines in the file representing the paragraphs of the original speech. That should not impact your program. Here are the first ten words of the file:

four score and seven years ago our fathers brought forth

## Instructions

Name your project FirstnameLastnameAssignment8

Have your program do the following.

1. Create the project.
2. Add the input file to the base directory of the project.
   * This is the same directory as the "build.xml" file.
3. Greet the user: "Welcome to Dr. Church's Word Length Program."
4. Prompt the user for a filename from which to read.
5. Prompt the user for a filename by which to write.
6. Open the input file for the purpose of reading.
7. Open the output file for the purpose of writing.
8. While there are still tokens in the file, write a word and that word's length.
9. Close the two file handles. Be kind to your operating system and your operating system will be kind to you.

My solution (not including the required documentation) was 37 lines of code. If you go over 50, you should probably ask questions in the discussion or send an email to your instructor.

## Documentation

Your source code must include the following documentation:

* Your name
* The class (CS 2070) and the section number (on ground is 08, online is W1).
* The date on which you turned in the assignment.
* A short description of the software. Usually a sentence or two is sufficient.

## Example Run

The example run is kinda boring since it only displays a few prompts and then quits.

Welcome to Dr. Church's Sentence Splitter.  
Enter a file to read: gettysburg.txt  
Enter a file to output: out.txt

The contents of out.txt will be each word and that word length. Here are the first 10 entries.

four 4  
score 5  
and 3  
seven 5  
years 5  
ago 3  
our 3  
fathers 7  
brought 7  
forth 5

## Turning it in.

To turn in your application, find the folder containing your entire project (not the folder with the "java" file), zip it up, and turn it in.