

# TCSS142 Introduction to Object-Oriented Programming

## Assignment 6

DUE: Monday, August 11, 2014 by 12 midnight – Name your file “Assign6.java” and Submit to Canvas

The purpose of this programming project is to demonstrate understanding of basic file input and output.

You are to write a program that reads lines of text from an input file until there are no more lines to read. As each line is read, your program should:

- ✓ Keep track of how many lines are in the input file
- ✓ Echo each line of input out to an output file
- ✓ Keep track of how many words are in the entire input file
- ✓ Count the occurrences of the following most commonly used words:
  - The
  - Of
  - To
  - And
  - A

Once the end of the input file has been reached, your program should then report all the above findings (counts) to the output file (see the sample run on page 2).

The above description clearly implies the use of `nextLine` to read each line into a `String` variable from the input file and then use a `Scanner` on the `String` variable to tokenize and count the occurrences of the 5 words. (Also implied is the need for 7 integer variables used for counting the 5 common words, total words, and total lines).

To compare strings when counting the five most common words, do not be concerned with upper or lower case as the `String` class provides a means to “ignore” such matters (review the Object Equality section in Chapter 4 pp. 246 & 247).

Your program should prompt the user for an input file name (be sure the file exists in the same folder as your program and you type the name exactly correct) and an output file name. When testing, you might want to omit this step and simply include a literal filename string with your call to the `File` constructor. Once your program is running correctly, then add the user input of file names.

You can add the safeguard feature to assure a correctly typed file name and that it exists (discussed in lecture and listed in the text) but, this is not a requirement.

You can open your files in `main` but do all other operations inside of methods. Remember, try to reduce redundancy and complexity through the use of methods.

As you have discovered, this assignment file is zipped with other files. Please note the other 3 files included can be used for testing as input files:

InputA.txt  
ILL Cry Instead.txt  
InputB.txt

### SIDE NOTE:

If you use the `printf` method for formatting and Word Pad to view your output files there should be no problem. However, if you use Notepad you might need to include the carriage return escape sequence character along with the newline character to start new lines: `\r\n` or the `printf` format specifier for newlines: `%n`

**A sample run based on the criteria above is listed on page 2.**

Like all programs, use the same documentation guidelines used in all other programming assignments. A method's purpose, and now, input and output should be clearly described before each method. Complicated code also needs documentation. Proper use of indentation and braces needs to be consistently implemented.

You should also include documentation at the beginning to identify yourself, course, date, instructor, etc., and a brief description of what the program does as you have done on all previous assignments.

## **SAMPLE RUN**

### **Includes user interaction and the contents of both the input and output files**

#### **User interaction:**

Please enter the input file name: `inputA.txt`  
Please enter the output file name: `z2.txt`

#### **Contents of inputA.txt:**

```
To test the program a file
has to be created of
words to count and to
count the occurrences
of the
words
the of
to
and
and a .
```

**(Note: The contents of the input file will be echoed to the console)**

#### **Contents of z2.txt:**

The input file "inputA.txt" has the following qualities:

A total number of:

Lines -	10
Words -	29
The word THE -	4
The word OF -	3
The word TO -	5
The word AND -	3
The word A -	2

#### **Using the input file inputB.txt produces the following output file:**

The input file "inputB.txt" has the following qualities:

A total number of:

Lines -	62
Words -	519
The word THE -	27
The word OF -	18
The word TO -	23
The word AND -	10
The word A -	18

**Using the input file ILL Cry Instead.txt produces the following output file:** (see next page ---->)

### Contents of ILL Cry Instead.txt:

I got every reason on earth to be mad  
'Cause I just lost the only girl I had  
If I could get my way  
I'd get myself locked up today  
But I can't, so I'll cry instead

I got a chip on my shoulder that's bigger than my feet  
I can't talk to people that I meet  
If I could see you now  
I'd try to make you sad somehow  
But I can't, so I'll cry instead

Don't want to cry when there's people there  
I get shy when they start to stare  
I'm gonna hide myself away  
But I'll come back again someday

And when you do you better hide all the girls  
I'm gonna break their hearts all round the world  
Yes, I'm gonna break 'em in two  
And show you what your lovin' man can do  
Until then I'll cry instead

Don't want to cry when there's people there  
I get shy when they start to stare  
I'm gonna hide myself away  
But I'll come back again someday

And when you do you better hide all the girls  
'Cause I'm gonna break their hearts all round the world  
Yes, I'm gonna break 'em in two  
And show you what your lovin' man can do  
Until then I'll cry instead

-John Lennon

### **Output:**

The input file "ILL cry instead.txt" has the following qualities:

A total number of:

Lines -	35
Words -	214
The word THE -	5
The word OF -	0
The word TO -	7
The word AND -	4
The word A -	1