CS 280
Spring 2025

Short Assignment 3
Words Recognizer

February 10, 2025

Due Date: Friday, February 14, 2025, 23:59
Total Points: 10

Write a C++ program that accepts one or more command line arguments for a file name and optional input flags. However, if no file name is provided, the program should print on a new line "NO SPECIFIED INPUT FILE NAME.", and then exit. If the file cannot be opened, print on a new line "CANNOT OPEN THE FILE ", followed by the file name, and then exit. The program should read from the file words until the end of file. The objective of the program is to recognize and count the number of three types of words. Those are the *Special Words, Keywords*, and *Identifiers*. If the input file is empty, the program should print out on a new line the message "The File is Empty." and then exit.

A word is defined as a sequence of one or more non-whitespace characters separated by whitespace. The three types of words are defined as follows. A *Special Word* is defined as a word that starts with either $, @ or % followed by zero or more letters, underscores, and digits. A *Keyword* is defined as a one of the following list of words: {begin, end, if, else, while, for, break, continue, case, switch, class, public, private, abstract, final}. Finally, an *Identifier* is defined as a word that starts by a letter and can be followed by zero or more letters or digits. Note, *Special Words* and *Identifiers* are case sensitive, while *Keywords* are not.

The first command line argument must be a file name. However, the input flags can be listed in any order using the following notations:

- -kw (optional): If it is present, the program prints the number of *Keywords* in the file.
- -sp (optional): If it is present, the program prints the number of each type of a *Special Word* in the file.
- -id (optional): If it is present, the program prints the number of *Identifiers* in the file.

If no input flags are present, only the total number of words in the file is printed out. The program should detect and display error messages of invalid *Identifiers* or *Special Words*. See the example below for the format of displayed error messages and printed results.

For example, given an input file of the following contents:

Line number	File contents
1	# Simple array constructs.
2	@fred = ("How", "are", "you", "today?");
3	print "\@fred contains (@fred).\n";
4	
5	$mike = $fred[1];
6	print " $mike $fred [3]\n";
7	
8	# The array name in a scalar context gives the size.
9	$fredsize = @fred;
10	print ' @fred has ', " $fredsize elements.\n";
11	
12	# The $#name gives the max subscript (size less one).
13	print "Max sub is $#fred\n";
14 →	
End of File	

The displayed output with all flags set would be as shown below:

```
Invalid Identifier Word at line 1: constructs.
Invalid Special Word at line 5: $fred[1];
Invalid Identifier Word at line 8: size.
Invalid Special Word at line 9: @fred;
Invalid Identifier Word at line 10: elements.\n";
Invalid Special Word at line 12: $#name
Invalid Identifier Word at line 12: one).
Invalid Special Word at line 13: $#fred\n";
Total number of words: 61
Number of Keywords: 0
Number of Identifiers: 25
Number of Special Words Starting with $: 5
Number of Special Words Starting with @: 3
Number of Special Words Starting with %: 0
```

Hints:

1. There are 9 test cases, these are case1-case9. Case1 is checking whether your program displays a message if there is no file name provided as an argument to your program. Note that case2 is for “infile” which does not exist. See the grading table below for descriptions of case3 to case9

2. If you want to look at the input for one of the test cases, use the linux "cat" command. The cases are in the directory $LIB/public/SA_Spring2025/SA3. For example, you can look at

“case3” file by using the command "cat $LIB/public/ SA_Spring2025/SA3/case3", and you can look at the expected output for this case by saying "cat $LIB/public/SA_ Spring2025/SA3/case3.correct".

Submission Guidelines

1. Please name your file as “SAx_firstinitial_lastname.cpp”. Where, “firstinitial” and “lastname” refer to your first name initial letter and last name, respectively, and “x” refers to the recitation assignment number (e.g., 1, 2, etc). Your program Submission is to Vocareum environment. Follow the link of Recitation Assignment 2 on Canvas in the Modules or Assignments pages to connect to the current assignment on Vocareum.
2. **Submissions after the due date are accepted with a fixed penalty of 25%. No submission is accepted after Sunday 11:59 pm, February 16, 2025.**

Grading Table:

Testing Cases	Points
Case 1: No file name is found	1.0
Case 2: File cannot be opened (myfile)	1.0
Case 3: Empty File (infie1)	1.0
Case 4: infile2 with no flags	1.0
Case 5: infile2 with all flags	1.0
Case 6: infile3 with “-sp”	1.0
Case 7: infile3 with “-id”	1.0
Case 8: infile3 with “-kw”	1.0
Case 9: infile4 with “-id -kw”	1.0
Compiles Successfully	1.0
Total	10