CS 280
Spring 2025

Short Assignment 4
Counting Occurrences of Word Types

February 17, 2025

Due Date: Friday, February 21, 2025, 23:59
Total Points: 11

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 occurrences of words of three types of words (as defined in SA 3). 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.

The program prints out the total number of lines and words in the file, and the number of words of each category of Words. However, the program prints out each word of a category and the number of its occurrences in the input file, based on a specified command line argument for that category. The words of each category are listed in order. Invalid words of a category are printed out as error messages similar to SA 3. The command line arguments, that are accepted by the program, are listed in any order. Those are defined as follows:

- A file name.
- Optional arguments that are specified using the following flags:
  - "-id": The program prints out each *Identifier Word* and the number of its occurrences in the file only.
  - "-sp": The program prints out each *Special Word* and the number of its occurrences in the file only.
  - "-kw": The program prints out each *Keyword* and the number of its occurrences in the file only.

In case more than one flag is specified, words and their occurrences are printed out in the order of *Special Words* first, followed by *Identifiers*, then *Keywords*.

An example of a test case (infile4) is shown next. Given the following input file,

```
begin formatting a paragraph with italic form @
An International Standard Book Number (ISBN) is a code of
%10 characters, referred to in this case as the class $ ISBN-10 ,
separated by dashes such as private number $0-7637-0798-8
$End the formatting
begin
If the ISBN number starts with ISBN-10 it consists of four parts:
group code, a publisher code, a code that uniquely identifies the book
among those published by
a particular publisher, and a check character.
The check character is used to validate an ISBN. For the ISBN 0-7637-0798-
8, the group code is 0,
which identifies the book as one from an English-speaking country.
While the %publisher code @_ 7637 is for @Jones and @Bartlett Publishers
```

The printed output with "-kw" and "sp" flags is as follows:

```
Invalid Special Word: $0-7637-0798-8
Total Number of Lines: 13
Number of Words: 118
Number of Special Words: 8
Number of Identifiers: 84
Number of Keywords: 9

List of Special Words and their number of occurrences:
$: 1
$End: 1
%10: 1
%publisher: 1
@: 1
@Bartlett: 1
@Jones: 1
@_: 1

List of Keywords and their number of occurrences:
begin: 2
case: 1
class: 1
for: 2
if: 1
private: 1
while: 1
```

Hints:

1. There are 10 test cases. These are described in the Grading Table below. Download the zipped file for the test cases from Canvas. These are the test cases you will be graded against on your submission to Vocareum. Use the test cases to test your implementation.
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/SA4. You can, for example, look at infile2 by saying "cat $LIB/public/SA_Spring2025/SA4/infile2", and you can look at the expected output by saying "cat $LIB/public/SA_ Spring2025/SA4/case5.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). Uploading and submission of your program is via Vocareum. Follow the link on Canvas for SA 4 Submission page.

2. Submissions after the due date are accepted with a fixed penalty of 25%. No submission is accepted after Sunday 11:59 pm, February 23, 2025.

Grading Table

Testing Cases	Points
Case 1: No input file name	1.0
Case 2: Cannot open a file	1.0
Case 3: Empty file (infile1)	1.0
Case 4: Unrecognized flag (infile1 -ids)	1.0
Case 5: "infile2" file with no flags	1.0
Case 6: "infile2" file with "-sp" flag	1.0
Case 7: "infile3" file with "-kw" flag	1.0
Case 8: "infile3" file with "-id" and "sp" flags	1.0
Case 9: "infile4" file with "kw" and "sp" flags	1.0
Case 10: "infile4" file with "-id" flag	1.0
Compiles Successfully	1.0
Total	11