Austin Pena (est892)  
CS 5103  
5/4/2021

Formal Specification for Word Statistics Project

Structure Specification:

|  |  |
| --- | --- |
| Word Frequency Counter | |
| Function | Count the lines and characters in a file, and compute the frequency of each unique word. Also, optionally replace a requested word. |
| Description | Attempts to read the words within a file specified by the user and then output the frequencies of each word along with the character count and line count. Also, the software will ask the user if a given word should be replaced in the file, then prompt the user for a word to replace. |
| Inputs | File name, replacement word info. |
| Source | Standard input. |
| Outputs | Word/line/character-frequencies printed to standard output. |
| Action | The frequency of each word can be computed by scanning each line of a file and adding the unique words as keys into a HashMap. The values of the HashMap will track the number of times each unique word has appeared. Total characters are added up while scanning the words. Total lines are calculated in a separate loop by scanning each line. Values are adjusted to accommodate words which are replaced. |
| Requirements | A file to be scanned must be locally available. |
| Pre-condition | None. |
| Post-condition | The file must be closed. |
| Side effects | None. |

Natural Language Specification:

1. Program shall create a HashMap with String keys and Integer values.
2. Program shall create integer variables for the count of lines and count of characters.
3. Program shall prompt the user for the name of a file to scan.
4. Program shall take in a String from standard input as the file name.
5. Program shall prompt the user if a word in the file should be replaced.
6. Program shall take in a String from standard input to determine if a word should be replaced. Program will keep taking in Strings until the String reads ‘y’, ‘Y’, ‘n’, or ‘N’.
7. If the String from standard input was ‘y’ or ‘Y’, then the program shall prompt the user for a word to be replaced and take in a String from standard input as the word to be replaced. The program shall then prompt the user for a word to be placed in, and the program shall take in another String from standard input as the word to be placed in.
8. Program shall attempt to open a file from the file name.
9. If the file fails to open, then the program shall print an error code, close the scanner, and exit.
10. Program shall read the lines of the file and increment the line count once per line.
11. Program shall read the file one word at a time.
12. Program shall replace any String to be replaced with the String that should be placed in.
13. Program shall increment the character count by the length of each word.
14. For each word, if the word is a key in the HashMap, then the program shall increment the value for that key by 1.
15. If the word is not a key in the HashMap, then the program shall add an element to the HashMap with a key matching the string and a value of 1.
16. After all words have been scanned, the program shall close the scanner.
17. Program shall print the character counts and line counts to standard output.
18. The program shall finally iterate through the HashMap and print to standard output each key and value pair. Each pair will be on a new line with a single space between the two.