CS280 Programming Assignment 1
Spring 2019

For this assignment, we want to process WORDS in input files and, potentially, change some of those WORDS before printing them out.

For our assignment, a WORD is a sequence of characters, separated by whitespace (that is, spaces, tabs, newlines; what cctype's isspace() returns true for).

It is possible that a WORD may begin and/or end with one or more punctuation characters (what cctype's ispunct() returns true for). Such a WORD therefore has leading and/or trailing punctuation.

A SUBSTITUTION is a line containing two WORDS.

Your program will process a file containing SUBSTITUTIONS and a file containing WORDS. Each SUBSTITUTION in the first file is a replacement rule for WORDS in the second file. Your program should process the input files and apply the SUBSTITUTIONS to each input WORD before printing it out. If a WORD in the input matches the first WORD in a SUBSTITUTION, it is replaced with the second word in the SUBSTITUTION, and then it is printed. If no SUBSTITUTION applies, the WORD is printed unchanged.

For example, imagine that we have these SUBSTITUTIONS in the first file:

foo many
hi hello
fish bicycle

And suppose we are given this second file:

So, hi everyone! This reminds me of foo things.
I need a new fish for my birthday.

The resulting output would be:

So, **hello** everyone! This reminds me of **many** things.
I need a new **bicycle** for my birthday.
NUMBER OF CHANGES: 3

There are a few rules about the file of SUBSTITUTIONS:

1. Blank lines are ignored
2. Lines that do not have two WORDS are ignored
3. Any leading and trailing punctuation in WORDS in a SUBSTITUTION are discarded
4. All of the letters in the WORDS in the SUBSTITUTIONS file should be converted to lower case
5. If there are multiple SUBSTITUTIONS with the same first WORD, the last SUBSTITUTION is used, and all prior SUBSTITUTIONS for that WORD are discarded.

Because we discard punctuation in the SUBSTITUTIONS, and because we convert to lower case, the following three lines in the SUBSTITUTIONS file all have the same first WORD. According to the rules, the last SUBSTITUTION is the only one that is retained.

Hi!!!   hello
“hi”   hello
Hi Hello

When deciding if an input WORD matches a word in a SUBSTITUTION, the following rules should apply:

1. Any leading and trailing punctuation is ignored when deciding if a WORD matches
2. Any difference in case are ignored when deciding if a word matches
3. When performing a replacement, leading and trailing punctuation from the WORD is preserved
4. If the first letter after any leading punctuation in the input WORD is a capital letter, then the first letter in the resulting replacement should also be capitalized
5. At most one replacement should be performed for each WORD

For example, imagine that we have these substitutions:

foo
hi hello
fish bicycle

And suppose we process this input:

“Hi, everyone!” said the boy. “Yeah, hi! I want a brand new fish!”

The resulting output would be:

“**Hello**, everyone!” said the boy. “Yeah, **hello**! I want a brand new **bicycle**!”
NUMBER OF CHANGES: 3

The program should keep a count of the number of substitutions applied. If any replacements were made, the last line of output, after processing the entire input file, should be the line:
NUMBER OF CHANGES: N
Where N is the number of replacements that were made.

The program takes two command line arguments: the first is the name of a file containing SUBSTITUTIONS, and the second is the name of a file containing WORDS. The program should process the SUBSTITUTIONS file according to the rules described above. Then, it should read the file containing WORDS, apply any replacements indicated from the SUBSTITUTIONS, and print out the result.

Note that any and all whitespace between WORDS in input is printed unchanged to output.

If exactly two filenames are not provided, print “TWO FILENAMES REQUIRED”, and stop. If a file cannot be opened or read for any reason, print the error message “BAD FILE FILENAME”, and stop. It is possible that either or both of the files your program reads may be empty. It is also possible that the second file may contain no words. Neither of these situations are errors.

The program will be submitted and graded in separate parts:

Part 1

Recognizing error cases associated with the wrong number of file names and files that cannot be opened.

Handling cases with an empty SUBSTITUTIONS file.

Handling simple substitutions (no punctuation, no case changes).

Part 2

Handling more complex cases in SUBSTITUTIONS files (duplicated words, mixed case words, punctuation, etc).

Handling cases with punctuation at the start and/or end of WORDS in the WORDS file.

Handling cases with mixed case for matches.

Handling cases with capitalized WORDS that are substituted.

Note that both in your “work” directory, and also on Moodle, there are files:

- cases.txt
- Cases.tar.gz
- runcase

The cases.txt file covers all the possible cases, as in what arguments are used for each test case.

The cases.tar.gz is a zipped file containing all the input and output files.

The runcase file is a shell script that lets you run a single test case.