

Censored Copy

In the folder "CensoredCopy" is an implementation and header file. You need to write a function, named "ReplacementCensor", that takes 3 arguments. The first argument is an istream of text to be processed. The second argument is an ostream to where the processed text should be copied too. The last argument is a map of strings to strings where the key is a string that needs to be replaced the value in the text of the first arguments contents. This function should return a set containing the words that where replaced. Please note that the keys should be matched in a case insensitive way and that text to replace may not be white space delimited (meaning they can appear in longer words).

For example, if the text in the stream is:



```
note: this is a line with multiple WORDs that should be rePLACed.  
all instances of word eveninlargerWordssould be replACed.
```

and the third argument is a map with the following pairs:



```
{"word", "Grouped-Letter-Unit"},  
{"be", "wasp"},  
{"not found", "not appearing"},  
{"PlaCe", "LoCation"}
```

then the resulting text written to the ostream should be:



```
note: this is a line with multiple Grouped-Letter-Units that should wasp reLoCation  
all instances of Grouped-Letter-Unit eveninlargerGrouped-Letter-Unitssould wasp r
```

lastly, the function should return a set of these strings:



```
{"PLACE", "WORD", "word", "be", "pLAcE", "word"}
```