[Plagiarism Detection](https://docs.dev.tripadvisor.com/display/DEV/Plagiarism+detection)

Your assignment is to write a command-line program that performs plagiarism detection using a N-tuple comparison algorithm allowing for synonyms in the text.  
  
Your program should take in 3 required arguments, and one optional.  In other cases such as no arguments, the program should print out usage instructions.

1. file name for a list of synonyms
2. input file 1
3. input file 2
4. (optional) the number N, the tuple size.  If not supplied, the default should be N=3.

The synonym file has lines each containing one group of synonyms.  For example a line saying "run sprint jog" means these words should be treated as equal.

The input files should be declared plagiarized based on the number of N-tuples in file1 that appear in file2, where the tuples are compared by accounting for synonyms as described above.  For example, the text "go for a run" has two 3-tuples, ["go for a", "for a run"] both of which appear in the text "go for a jog".

The output of the program should be the percent of tuples in file1 which appear in file2.  So for the above example, the output would be one line saying "100%".  In another example, for texts "go for a run" and "went for a jog" and N=3 we would output "50%" because only one 3-tuple in the first text appears in the second one.  
  
**Format:**  
  
We prefer Java SE for the solution, but if you are not comfortable coding in Java, we can also accept ANSI C++ or Python.  No libraries outside the core language should be used.  The solution can be one or more source files.  
  
**Style:**  
  
While getting the right answer is important, we are also interested in how well thought out your solution is; are there easier, or faster ways? Is the code understandable to another engineer picking it up? If there are obvious ways it could be abstracted or extended, is it designed to support that?  For example, this N-tuple detection algorithm may end up being used in other contexts (like a website) and so should be easy to reuse.

**Questions?**

Some aspects of the problem are intentionally vague. We recommend that you make your own decisions rather than ask us and wait for an answer. Document any important assumptions that you make in your solution.

**Example inputs from above:**  
  
syns.txt:  
run sprint jog  
  
file1.txt:  
go for a run  
  
file2.txt:  
go for a jog  
  
Sample output for above inputs:  
100%