Corporate Headquarters | 19900 MacArthur Blvd., Suite #370, Irvine, CA 92612
909.217.3344 | info@smartenergywater.com | www.smartenergywater.com

Hi,

Thank you for your interest in SEW's dynamic engineering team. At SEW, regardless of the role we play, we think we are passionate technologists and strong analytical thinkers. As we prepare things on our side for an in-person interview with you, we'd like you to take a shot at solving an interesting problem that will help us understand you a little better.

We'd prefer if you wrote this in .NET, but if you're more comfortable with another language, that would be totally fine as well. Please treat this as an opportunity to demonstrate your engineering judgment in considering factors like performance, modularity and extensibility in a way that you cannot do in personal interviews.

Write a program that reads a file containing a sorted list of words (one word per line, no spaces, all lower case), then identifies

1. The longest word in the file that can be constructed by concatenating copies of shorter words also found in the file.
2. The program should then go on to report the 2nd longest word found
3. Total count of how many of the words in the list can be constructed of other words in the list.

For example, if the file contained:

cat

cats

catsdogcats

catxdogcatsrat

dog

dogcatsdog

hippopotamuses

rat

ratcatdogcat

Corporate Headquarters | 19900 MacArthur Blvd., Suite #370, Irvine, CA 92612
909.217.3344 | info@smartenergywater.com | www.smartenergywater.com

The solution would be 'ratcatdogcat' - at 12 letters, it is the longest word made up of other words in the list. The program should then go on to report how many of the words in the list can be constructed of other words in the list. Please find attached a file, "wordlist.txt", containing a word list, with 173k rows, that you can use as an input to test your code.

Please upload your solution as a github repository and email us the link. It is highly recommended that you add a Readme file to the repo where you can state any assumptions you may have made. We'd also like you to include the following information in that file, as a starting point for us:

The 1st and the 2nd longest words that the program finds and the count of words that can be constructed as an output in the body of the email and any comments you have on the approach you took.

If you have any questions about the problem, please feel free to ask. Good luck!