Coding test 5: Word processing

You are writing the software **CountYourWords**, a word processing system that is capable of counting words in a text DOCUMENT. The document is basically a text file that needs to be parsed. The following remarks below apply:

- Numbers in the document are ignored and are not processed
- Other characters than words should be filtered out of the input, so ## or @ or !! are ignored
- You do not have to take in account strange combinations like: love4u or mail@address.nl, combinations like these are out of scope for this assignment
- Next to showing the total number of words in the document, the number of occurrences of each word is also calculated
- The total number of occurrences next to the word must be shown on screen one by one (in lowercase)
- Counting the occurrences per word is case insensitive (so Matchbox, matchbox, and MATCHBOX are all the same word)
- When printing the occurrences, the words must be in alphabetical order
- You cannot use a built-in .NET sort routine, so produce on of your own, it does not have to be the fastest sort-routine as long as it is reliable end reusable.
- The document is a text file that will be read by your console app and has the fixed name: *input.txt*

  *(see **examples** at the end of this document)*

  and…

- The code must be production code, use everything you would use in production. You are allowed to use third party components if required (except for the sorting 😉)
- All classes can be in one .NET core console application (assembly), as long as you show clear separation
- Everything must be done using testing and TDD
- Use SOLID and object-oriented programming
- Use .NET 8 core
- Any questions or something not clear? feel free to ask …
- **[BONUS, if you have time left]** Since the CIA and RUSSIA are monitoring every move of what you are doing the list of words must also be returned by your CountIt in a scrambled form, that is the letters must in reversed order and every EVEN letter in the word must be capitalized. Mind you the word scrambling routine you have to write is used throughout the system, and might be replaced later on

Example:

If your document is

“….. The big brown fox number 4 jumped over the lazy dog. THE BIG BROWN FOX JUMPED OVER THE LAZY DOG. The Big Brown Fox 123 !!”

Your console output would be:

```
Number of words: 23

big 3
brown 3
dog 2
fox 3
jumped 2
lazy 2
number 1
over 2
the 5
```

Bonus (looks similar to):

```
gIb
nWoRb
gOd
xOf
dEpMuJ
yZaL
rEbMuN
rEvO
eHt
```