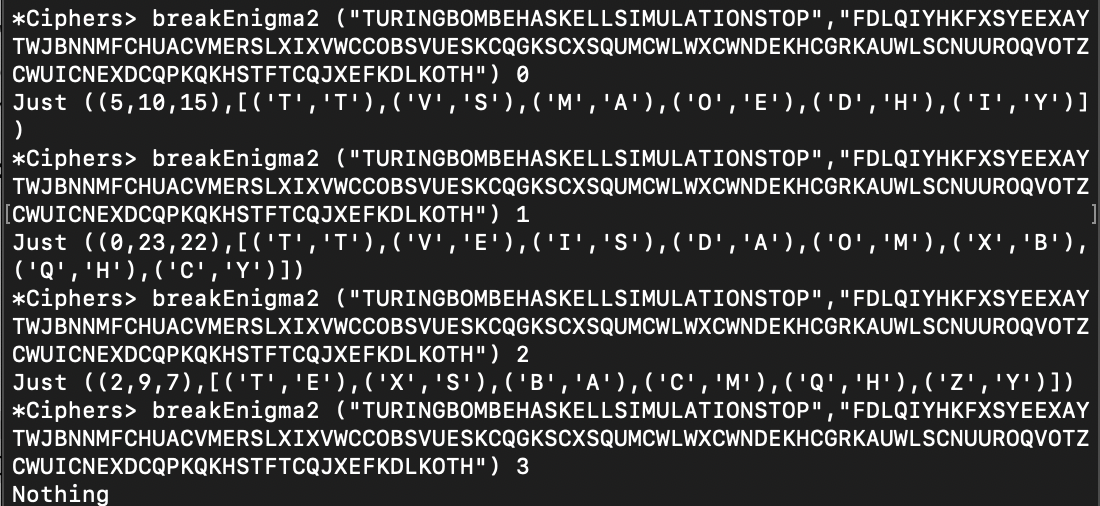
Experiments and Testing

Daniel Matyas Perendi

As the longestMenu function only returns one of the possibly many longest menus, I decided to experiment with all possible longest menus. Different menus in the same crib can give different initial offsets and steckerboards. I have created a function which takes an integer as well as a crib, and then returns the initial offsets and the steckerboard. The integer refers to the index of the menu we are using in the list of all possible longest menus. This way if we try different menus, we will get different results. Different menus with different starting positions will produce different offset settings, and hence different steckerboards. The steckerboard with more elements might be more useful than other solutions when the decrypted message is longer than the plain text, however it’s not always true, if we added all the repeated pairs to the steckerboard they might be the same in size, but as we can’t add duplicates to the list, these pairs are “hidden” even though they are just as useful as other pairs in a different list. When we pass a too large integer onto this breakEnigma2 method it simply returns Nothing without “thinking”, so we know there are no more longest menus. I found it so interesting how it found different settings every time I tried different menus. What I found even more interesting is that some of the solutions made no sense when I tried to decrypt the message using a SteckeredEnigma with the settings I got from breaking the enigma. It however makes a lot of sense, because an enigma machine can only decrypt a message if and only if it has the exact same initial settings as the machine which was used to encrypt it.



Nowadays it is not very hard to make experiments on the enigma machine, because we get told how it works and what we have to do in order to build a bombe “simulator” to break it, however I can’t even imagine how hard it was for the Bletchley Park codebreakers to understand how it works, then find out the settings to decrypt messages.

**Testing:**

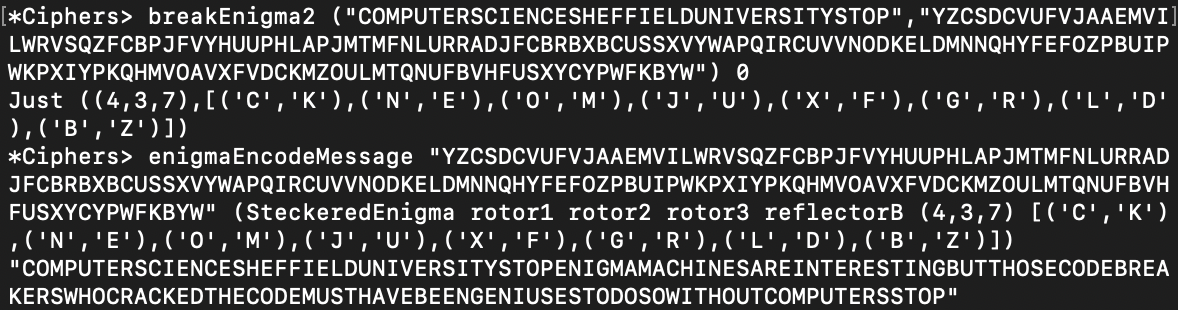
Contrived examples:





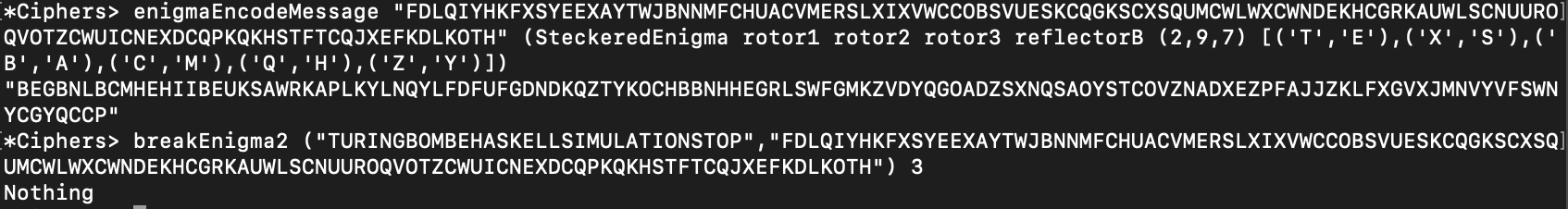
These examples gave the expected results.

Another test from mole:



In this example breakEnigma found all the pairs and the right initial offsets, however in the next example…

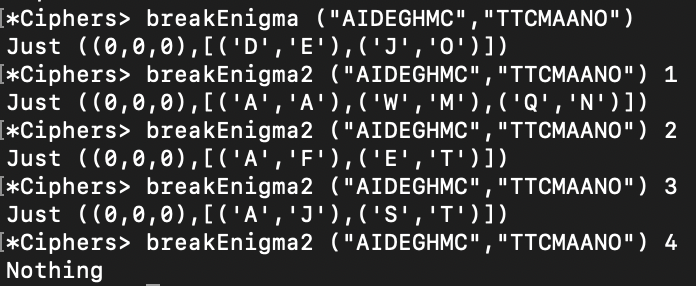
****

****

…I tried every possible longest menu, then I tried to decode the message with the settings I got, but it was too messy, I couldn’t work out the missing letter pairs.

**Tests from bombeTesting16.hs:**

Test 1:



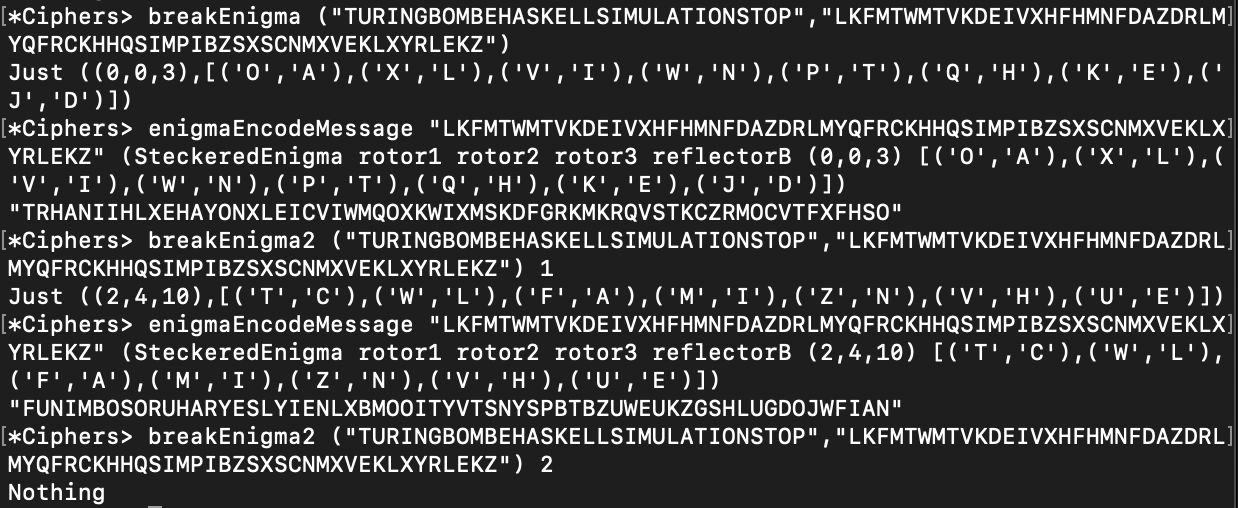
This test failed because of the short menu size we get. I tried every single longest menu and none of them gave the expected result.

Test 2:



I got a result, but it’s not good enough to guess the remaining letters.

Test 3:



Same as the previous test, I got multiple results for different menus, and none of them is good enough to guess the remaining letters.