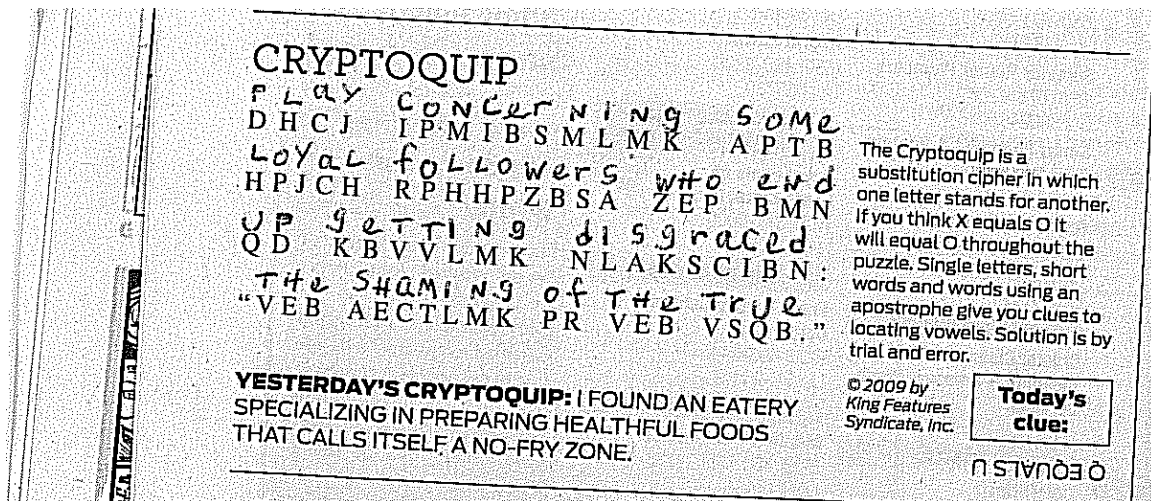


CmpE 101 Fall 2009 Programming Assignment 2

We're going to build a Cryptoquip helper. Cryptoquip is a puzzle that appears in many newspapers, including the San Francisco Chronicle. As far as I know it can't be accessed on line, at least not for free anyway. The Cryptoquip is a substitution cipher in which one letter stands for another and always has the same encoding. Spacing and punctuation marks are preserved, and a letter is never encoded into itself. Only uppercase letters are used.



A Cryptoquip looks like this:

WS KNC RYYO ELYHVWXD BTHB KNC'LY BHRWXD H QVHJJ ELWXR, WQ BTHB H SLYCEWHX QWO?

Today's clue: Q equals S

Check out <http://members.shaw.ca/macbride/crypto/> for an example Cryptoquip helper in Java. Our program will work the same way, minus the nice user interface.

Since the Cryptoquip is solved daily by thousands of newspaper readers across the US, it can be concluded that it is not exactly the state of the art in cryptography. Nevertheless it would be fairly challenging to write a program to solve it. (A possible senior project?) However, it would be helpful to have a program that would assist in solving the puzzle, since a lot of guesswork is required and you have to keep track of your guesses.

What your code should do

1. Your code should ask the user to enter a file name from the keyboard.
2. Your code should open the file, read it character by character and store it in a character array. The file contains the original Cryptoquip. If there are any lower case letters they should be changed to upper case. Newline characters should be removed (but not blanks).
3. If the file did not open correctly, you should print an appropriate message and then exit. You need to test for this.
4. Assume that the input file contains just text (the kind that you get when you create a file using **Notepad**). Don't worry about what happens with non-text files.
5. Now the program goes into a loop. Each iteration of the loop starts by printing several things:
 - A. The original encoded message you read in at the start (cyphertext).
 - B. The encoding, which is printed as one line with the letters A-Z, and immediately underneath on the next line, the corresponding decoded letter.
 - C. The decoded message, where the decoded letters have been substituted (guess at the plaintext).
6. Then the user is prompted to enter a one letter substitution. That is one time through the loop.
7. When every letter in the encoded message has been substituted, the user can be prompted to ask if she is finished. Not until then.
8. Since when you start no encodings are known, the encoding for all letters should be set to '#' initially. Also, you should be able to set a letter back to '#' (if you made a wrong guess).

Hints

1. You need to use arrays of char.
2. The project doesn't encode anything, it only decodes. Make sure you do things the right way around.
3. The encoding must be unique. If the user enters a new guess for a plaintext letter, any old guess should prevent the change. For example, if I guess "M=L" and later I guess "K=L", two different ciphertext letters can't map to the same letter (in this case L), so the guess isn't accepted.
4. Make sure you can set your guess back to '#' if you decide it is wrong.