Project 3

Cryptoquote

Introduction

Cryptoquote is a popular game that appears in many newspapers. A statement by some notable person is encrypted using a simple substitution code. The encrypted quote is printed in the newspaper and the reader is supposed to figure out what the original quote is. For example, the quote

```
IF YOU CAN'T STAND THE HEAT, GET OUT OF THE KITCHEN. - HARRY TRUMAN
```

might be encrypted as

```
RJ EMK DPS'W HWPSY WQG QGPW, OGW MKW MJ WQG ZRWDQGS. - QPCCE WCKLPS
```

where every `I` is replaced by `R`, every `F` is replaced by `J`, and so on. Seeing the word `DPS'W`, the player might guess that `W` represents either `S` or `T`. Seeing the word `WQG` repeated and suspecting that `W` is `T`, he might guess that this is the word `THE`. If he makes enough correct guesses, then the rest of the pieces will fall into place.

The Program Interface

The user (player) will be presented with a puzzle chosen at random from a list of quotes stored in the file `Quotes.txt`, which will be provided. Then the program will create a random permutation of the letters of the alphabet and use this permutation to encrypt the quote. The encrypted version will be displayed. Underneath the encrypted version will be printed a copy of the puzzle where the letters are replaced by underscores, followed by the frequencies of the letters, as an aid to guessing. For example, the program might print

```
RJ EMK DPS'W HWPSY WQG QGPW, OGW MKW MJ WQG ZRWDQGS. - QPCCE WCKLPS
-- --- ---'- ----- --- ----, --- --- -- --- -------. - ----- ------

Frequencies of the letters
```

```
W 9     G 5     P 5     Q 5     S 4     C 3     K 3     M 3     D 2     E 2
J 2     R 2     H 1     L 1     O 1     Y 1     Z 1
```

Notice that non-letters are not encrypted. As the player makes guesses, the underscores will be replaced by his guesses. After making the guesses described above (`W` → `T`, `Q` → `H`, `G` → `E`), the puzzle will look like this:

```
__ ___ ___'T _T___ THE HE_T, _ET __T __ THE __T_HE_. - H____ T_____
```

For each guess, the program will prompt the player to enter the letter to change. After the user enters that letter, it will prompt him for the letter to change it to. After entering that letter, the program will print the updated puzzle and continue. The user may enter each letter in either uppercase or lowercase; the output will be in uppercase. If the player enters a question mark (`?`) instead of an encrypted letter to be replaced, then the program will print the answer and quit. If the player enters a period `'.'` instead of a replacement letter, then the previous guess for that letter will be replaced with a blank space.

Procedure

RJ EMK DPS'W HWPSY WQG QGPW, OGW MKW MJ WQG ZRWDQGS. - QPCCE WCKLPS

The program should begin by getting a quote and encrypting it. Then enter a loop that will continue processing guesses until either the user's answer matches the original quote or until the user enters a question mark `?`. On each iteration of the loop, after receiving the encrypted letter and its replacement letter, each occurrence of the encrypted letter should be replaced by the replacement letter and the result displayed. The functions described below will facilitate that.

Functions

In the file `Functions.cpp` you will find some of these functions described below. Copy and paste them into your file `Cryptoquote.cpp`. You must write the other functions and the main function.

```
string getQuote(;
```

This function will open the file `Quotes.txt`. It first reads the number of quotes in the file and then chooses a random integer with that range. It will then read down to the randomly selected quote and return it. This function is provided. However, you must enter the exact pathname to the file of quotes.

`string toupper(string s);`

This function will have one string parameter. It will convert the string `s` to uppercase and return the uppercase version. Note that this function overloads the library function `toupper()` which converts a `char` to uppercase. This function is provided.

`string permute();`

This function will randomly permute the letters of the alphabet and return the permuted letters as a string. The string will later be used as the "key" to encrypt the quote. This function will be provided. This function is provided.

`void frequencies(string s):`

This function displays a table of all the letters in the string `s` along with their frequencies in the string. This information is useful when making guesses. This function is provided.

`string encrypt(string quote, string key);`

This function will use the permuted alphabet, called the `key`, to encrypt the text of the quote that was read from the file. For example, if the key begins `"RHN..."`, then every `A` in the quote should be replaced with `R`, every `B` with `H`, every `C` with `N`, and so on. The encrypted string should be distinct from the quote string. Do not change the letters in the quote string itself.

`string fillUnderscores(string quote);`

This function will return a copy of the encrypted string with the letters replaced with underscores.

`void replaceLetter(string encrypted, string& answer, char encrypted_char, char replace_char);`

This function will replace each occurrence of `encrypted_char` in the string `encrypted` with `replace_char` in the `answer` string.

Turning in Your Work

Place your file `Cryptoquote.cpp` in a folder named `Project 3` and drag the folder to the dropbox. Be absolutely sure that the file name and folder name are correct. Do not include any files in the `Project 3` folder that you did not write. Your work is due by midnight, Monday, October 21, 2019.