Imperial College of Science, Technology and Medicine
Department of Computing

M.Sc. C++ Programming – Unassessed Exercise No. 5

Issued: Friday 15 October 2021

Problem Description

Pig Latin[1] is an elegant but archaic pseudo-language that was first spoken in the Middle Ages. Invented by commoners so that they could sound as fancy as their lords, it is based on manipulating the letters of English words so that they sound like Latin words in plural feminine form (i.e. they end in an "ay" sound).

It is a little known fact that the Magna Carta was first written in Pig Latin and only later translated into lesser tongues. Today, Pig Latin is used mostly for amusement although it also has more serious applications such as obfuscating song titles to evade copyright restrictions on music web sites.

Any English word may be changed into its Pig Latin equivalent as follows:

1. If the word begins with a vowel, add "way" to it. For example, Pig Latin for `apple` is `appleway`.
2. If the word begins with a letter that is not a vowel, find the first occurence of a vowel, move all the characters before the vowel to the end of the word, and add "ay". For example, `grape` becomes `apegray` and `strong` becomes `ongstray`.
3. If the word contains no vowels just add "ay" to it.
4. For the purposes of this exercise the vowels are `a`, `e`, `i`, `o`, `u` and `y`; but `y` is only considered a vowel if it is not the first or last letter of the word. So `yeti` becomes `etiyay`, `my` becomes `myay` and `crying` becomes `yingcray`.
5. If the word begins with a character that is a digit, leave the word as is. For example, `300` remains `300`.
6. If the word begins with an initial upper case (captial) letter then so should the corresponding Pig Latin word. For example, `Banana` becomes `Ananabay`. You do not need to handle other capitalization patterns (e.g. all uppercase words).

[1] an important language to learn in case you ever find yourself on a Roman farm.

Specific Tasks

1. Write a function `findFirstVowel(word)` which returns the position of the first "vowel" in the given word (see the Pig Latin rules on the previous page for the precise definition of "vowel"). If the word does not contain a vowel then the function should return -1. The first parameter to the function (i.e. word) is a read-only string containing a single English word.

   For example, the code:

   ```
   int vowel;
   vowel = findFirstVowel("passionfruit");
   ```

   should result in the integer `vowel` having the value 1.

2. Write a function `translateWord(english, piglatin)` which produces a Pig Latin translation for a given English word. The first parameter to the function (i.e. `english`) is an input string containing the English word. The second parameter (i.e. `piglatin`) is an output parameter which should contain the corresponding Pig Latin translation.

   For example, the code:

   ```
   char translated[100];
   translateWord("Banana",translated);
   ```

   should result in the string `translated` having the value `Ananabay`.

3. Write a function `translateStream(inputStream, outputStream)` which takes words from an input stream and writes a corresponding Pig Latin translation to an output stream. The first parameter (i.e. `inputStream`) is the input stream (e.g. `cin` or a file input stream) and the second parameter is the output stream. You may assume that both the input and output streams have been initialised/opened appropriately before the function is called, and that no word in the file is longer than 64 characters.

   For full credit, your solution (to question 3) should be recursive. However, partial credit (up to 75%) will be awarded for a working iterative solution.

For example, given an input file called `fruit.txt` that contains the text:

```
Time flies like an arrow,
but fruit flies like a banana!
(Groucho Marx 1890-1977)
```

then the code:

```
ifstream input;
input.open("fruit.txt");
translateStream(input, cout);
input.close();
```

should result in the following output written to `cout` (i.e. displayed on the screen):

```
Imetay iesflay ikelay anway arrowway,
utbay uitfray iesflay ikelay away ananabay!
(Ouchogray Arxmay 1890-1977)
```

What to hand in

Place your function implementations in the file **piglatin.cpp** and corresponding function declarations in the file **piglatin.h**. Use the files **main.cpp** and **fruit.txt** (available from the URL: `http://www.doc.ic.ac.uk/~wjk/C++Intro/piglatin/`) to test your functions. Create a **makefile** which compiles your submission into an executable file called **piglatin**.

How You Will Be Marked

If this was assessed, you would be assigned a mark according to whether your program works or not, whether your program is clearly set out with adequate blank space, comments and indentation, whether you have used meaningful names for variables and functions, and whether you have used a clear, appropriate and logical design.

Hints

1. Feel free to define any auxiliary functions which would help to make your code more elegant. For example, in Question 1 you might find it useful to define a helper function `bool isVowel(char ch, int position, int length)` that returns true if the letter `ch` at position `position` in a string with `length` characters is a "vowel" according to the Pig Latin rules.

2. The standard header `<cctype>` contains some library functions that you may find useful. In particular:

   - `int isupper(char ch)` returns nonzero if `ch` is an uppercase letter from `'A'` to `'Z'`.
   - `int isalpha(char ch)` returns nonzero if `ch` is a letter from `'A'` to `'Z'` or a letter from `'a'` to `'z'`.
   - `int isdigit(char ch)` returns nonzero if `ch` is a digit between `'0'` and `'9'`.
   - `int isalnum(char ch)` returns nonzero if `ch` is a letter from `'A'` to `'Z'`, a letter from `'a'` to `'z'` or a digit between `'0'` and `'9'`.
   - `char toupper(char ch)` returns the upper case equivalent of character `ch`.
   - `char tolower(char ch)` returns the lower case equivalent of character `ch`.