Assignment 3 – Pig Elvish and Largest Number

Goals

- For loops
- While loops
- String processing and manipulation

Part 1 – Igpén Lvísheá (Pig Elvish)

- **Background**
  - Pig Elvish is a gibberish (made-up) language similar to Pig Latin, but designed to sound like the Elvish language spoken by the Elves in *Lord of the Rings.* (example of Elvish)
  - To "translate" an English word into Pig Elvish, follow these rules (adapted from this site):
    1. Take the first letter of the word and move it to the end of the word
    2. If the word is four letters or more, append a random vowel to the end of the word (**aeiou**)
    3. If the word is three letters or fewer, append "**en**" to the end of the word
    4. Change all **k**'s to **c**'s
    5. If there is an **e** at the end of the word, replace it with **ë** (**e** with an umlaut)
    6. Handle capital letters properly:
       - If the first letter of the English word is capitalized, make it lower case when you append it to the end
       - Then capitalize the new first letter of the Pig Elvish word
         - Tolkien Olcienti
         - Trojan Rojantu
  - Examples:
    - a aen
    - while hilewa
    - python ythonpë
    - Quick Uiccqi
  - Extra credit: Complete these two optional steps
    7. *Randomly add accents (**áéíóú**) some vowels*

- In order to randomly add accents to vowels in a word, you can use the **`random.choice()`** function (see hint below)
- Examples:
  - a aén
  - while hilewá
  - python ythónpë
  - quick uíccqí

8. Reverse the program and create a Pig Elvish to English translator following the same general structure.
   - Question (include answer in your comments): Is this translation complete reversible? That is, can you be guaranteed to be back the original word given only the translated word? Explain

- **Requirements**
  - Create a program to translate a single word at a time from English into Pig Elvish
    - Important: You are **only** required to implement **steps 1-6** above
  - Using a **while** loop, first ask the user enter a word in English
  - "Translate" the user's word into Elvish
  - Display the word in Elvish
  - Ask the user if they want to continue
    - If yes, ask them for another word to translate
    - If no, print a goodbye message in Elvish
  - Hint #1: Strings have methods that may be useful
    - **`someString.isupper()`** checks whether all the letters in the string are uppercase and returns a Boolean: **`True`** if the string is all uppercase, or **`False`** otherwise.
      - For example,

```
# consider that letter is a string
if letter.isupper() == True:
   # letter is uppercase
else:
   # letter is lowercase
```

    - **`someString.capitalize()`** returns a copy of the string with only the first letter capitalized

- For example,

```
# consider that letter is a string
capitalLetter = letter.capitalize()
```

- **someString.replace(old, new)** returns a copy of the string with all of the old letters replaced by the new letter
  - For example,

```
# consider someString = "hello world"
someString = someString.replace("l", "x")
# now someString = "hexxo worxd"
```

- Hint #2 for Extra Credit: If you have a sequence (e.g. a string) and you want to randomly select one element from the sequence, you can use the **random.choice()** function
  - For example,

```
import random
message = "hello world"
letter = random.choice(message)
# letter now holds a random character from message
```

Sample Output for Part 1

```
Elcómewó óten heten Igpén Lvísheá ránslátórtë!
(Welcome to the Pig Elvish translator!)

Please enter a word you would like to translate: gandalf
'gandalf' in elvish is: andalfgi

Would you like to translate another word? (y/n): y

Please enter a word you would like to translate: orc
'orc' in elvish is: rcoen

Would you like to translate another word? (y/n): n

Oodbyega! Aveha aen icenë ayden!
```

```
(Goodbye! Have a nice day!)
```

Part 2 – Largest Number

- **Requirements**
  - Using a **while** loops, find the largest number that a user enters.
  - Ask the user to input an integer greater than or equal to **0** or **-1** to quit.

    ```
    Input an integer greater than or equal to 0 or -1 to quit:
    ```
  - When the user enters **-1**, print out the largest number found.
  - Solve this part of the assignment **without** using lists
  - After finding the largest number, ask the user if they would like to find another largest number, or if they would like to quit.
    - Hint: How many while loops do you need to complete this part?

Sample Output for Part 2

```
Input an integer greater than or equal to 0 or -1 to quit:
> 1
> 8
> 34
> 9
> -1
The largest number is 34

Would you like to find the largest number again? (y/n): y
Input an integer greater than or equal to 0 or -1 to quit:
> 7
> 2
> 3
> 2
> 2
> -1
The largest number is 7

Would you like to find the largest number again? (y/n): n

Goodbye!
```

Deliverables and Submission Instructions

- Create a folder on your computer called
  `ITP115_a#_lastname_firstname`
  (replace # with this lab number)
- Inside the folder, include your python source code
- Compress the folder (make a zip file) called
  `ITP115_a#_lastname_firstname.zip`
  (replace # with this assignment number)
- Upload zip file to Blackboard site for our course
- Assignments that do not run are subject to 50% penalty

Grading

Item	Points
Part 1: Pig Elvish	15
Part 2: Largest Number	10
Total*	**25**

** Points will be deducted for poor code style, or improper submission.*