CPS109 Lab 2

Most of the questions in this lab come from Chapter 3 (or earlier chapters) of the course text, Introduction to Computing and Programming in Python, by Guzdial and Ericson. Please put your answers (numbered) in a document and submit it on D2L as a PDF file. Other formats are not accepted.

The **learning objectives** for Chapter 3 are to be able to:

- manipulate strings
- build strings with concatenation
- use loops to iterate over the characters in a string
- convert strings into lists for word manipulation
- use array notation s[i] for accessing elements of strings and lists
- use if-statements to branch in the instruction sequence

To do:

1) You are running a Bingo game and you want to tweet the name of how won and how much they won. Write a function with two parameters, one being the name, and the other being the number of dollars won. For now, rather than tweet, just print out a statement saying who won and how much. Remember that the dollar amount is a number which needs to be changed to a string using str() in order to concatenate it into the sentence that you print. For example, **bingo('Mark', 50)** would print '**Mark called Bingo and won $50!**'.
2) You are a race official on an ultra marathon (100 mile) race, and you want to display the number and elapsed time for each runner on a big display. Write a program to generate the phrase for the display based on the runner number, the mile marker, and the elapsed time. For example, **runner(42, 10, "1:12:09)** would print **Runner #42 passed mile 10 at time 1:12:09**.
3) Write a function like the pyramid function, except that it prints an inverted pyramid, and it takes in a character as a paramter. For example, **invertedpyramid('a')**, prints:

```
aaaaaaaaa
 aaaaaaa
  aaaaa
   aaa
    a
```

4) Using for loops, create a **textsquare(ch, n)** function that prints a square pattern like the following example for **textsquare('t', 5)**:

```
ttttt
t   t
t   t
t   t
ttttt
```

5) Write a function **justConsonants(string)** which uses the condition **not (letter in 'AEIOUYaeiouy'** to avoid printing the vowels. For example, **justConsonants('The Old Gray Mare')** would print **Th ld Gr Mr**.
6) Write a function **justConsonants2(string)** which uses the condition **not (letter.lower() in 'aeiouy'** to avoid printing the vowels. For example, **justConsonants2('The Old Gray Mare')** would print **Th ld Gr Mr**.
7) Which of the following programs for **dup_('rubber duck')** where the underscore is a digit produces '**kcud rebburrubber duck**'?

```
def dup1(s) :
  target = ''
  for letter in s :
    target = target + letter
  return target

def dup2(s) :
  target = ''
  for letter in s :
    target = target + s
  return target

def dup3(s) :
  target = ''
  for letter in s :
    target = letter + target + letter
  return target

def dup4(s) :
  target = ''
  for letter in s :
    target = letter + target
  return target
```

8) Write a program **dup5(string)** which mirrors a string and puts '-' between letters as in the following example: **dup5('rubber duck')** produces '**k-c-u-d- -r-e-b-b-u-r--r-u-b-b-e-r- -d-u-c-k**'?

9) Write a program that takes a string as a parameter, then prints the vowels on one line and the consonants on another. For example, **separate('elephant DUMBO')** produces:

Vowels: eeaUO
Consonants: lphnt DMB

10) Recall the keyword cipher programs from Chapter 3: **buildCipher(key)** and **encode(string, alpha2)**. There was a little trouble with blanks, since they turned out to be 'z'. For example, below are the programs and the sample run. Your job is to write **encode2(string, alpha2)** which ignores blanks and punctuation, in fact anything that is not in the alphabet. The example run with **encode2** is in bold.

```
def buildCipher(key) :
    alpha1 = 'abcdefghijklmnopqrstuvwxyz'
    alpha2 = key
    for letter in alpha1 :
        if letter not in key :
            alpha2 = alpha2 + letter
    return alpha2

def encode(string, alpha2) :
    alpha1 = 'abcdefghijklmnopqrstuvwxyz'
    secret = ''
    for letter in string :
        i = alpha1.find(letter)
```

```
        secret = secret + alpha2[i]
    return secret

>>> alpha2 = buildCipher('earth')
>>> print alpha2
earthbcdfgijklmnopqsuvwxyz
>>> secret = encode('this is a test', 'earthbcdfgijklmnopqsuvwxyz')
>>> print secret
sdfqzfqzezshqs
>>> secret = encode('this is a test!!!', 'earthbcdfgijklmnopqsuvwxyz')
>>> print secret
sdfqzfqzezshqszzz

>>> secret = encode2('this is a test', 'earthbcdfgijklmnopqsuvwxyz')
>>> print secret
sdfqzfqeshqs
```