

Michelle M. Khalifé
Fall 2019

ASSIGNMENT 2

1. Candy or Bowtie

Write a program that draws a candy or a bowtie, given the following user specifications:

- candy (c) or bowtie (b)
- width (even or odd integer)
- tight or relaxed pattern (default is tight, see first two diagrams)

<pre> ***** ***** ***** **** *** ** * * ** *** **** ***** </pre>	<pre> ***** ***** ***** **** *** ** * * ** *** **** ***** ***** ***** ***** ***** ***** **** *** **** ***** ***** ***** </pre>	<pre> * </pre>	<pre> * </pre>
>>> bowtie, 5	>>> candy, 6	>>> candy, 3, r	>>> bowtie, 4, r

2. Boolean expressions. Do not use conditionals.

- a. Write a function **is_leapyear()** that returns True if a year is a leap year, and False otherwise. A leap year is divisible by 400 or by 4 but not by 100.
- b. Write a function **is_vowel()** that returns True or False depending on whether its arguments is a vowel or consonant.

Michelle M. Khalifé
Fall 2019

3. List comprehension

- a. Use list comprehension to output 5 random numbers between 1 and 1000, divisible by 5 and 7.
- b. Use list comprehension to print a list after deleting even numbers.

4. Shuffle

Write a program that reads a sentence from the command line as a list of strings and returns the sentence in a shuffled order. The strings in the list need to change positions in place: do not create a duplicate list.

5. Pascal Triangle

Write a program that prompts the user for an integer and builds and prints the equivalent Pascal triangle. In a Pascal triangle, every row may be computed from the previous row by adding adjacent pairs of values together. Using a two-dimensional list (list of lists), Pascal's rule stipulates that, for any non-negative integer n and any integer k , between 0 and n , inclusive: $a[n][k] = a[n-1][k-1] + a[n-1][k]$.

The following is for $n = 7$

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1
```

6. Student Records

Students have been registered in the Computer Science program on a first come, first served basis. But now that the dust has settled, the administration wants to sort the student records in alphabetical order. Each student record is a tuple consisting of the following information: (last name, first name, Month-of-birth, Year-of-birth). Write a function that return a sorted list of tuples on last name, first name, year, then month.

7. Word Frequency

- a. Write a function called *word_frequencies(myList)* that accepts a list of strings called *myList* and returns a dictionary where the keys are the words from *myList* and the values are the number of times that word appears in *myList*.
- b. Write a function to invert key and value pairs in the dictionary you constructed. Sort the dictionary according to the new keys. The function should accept a dictionary as parameter and return a tuple with the most frequent word and its frequency.