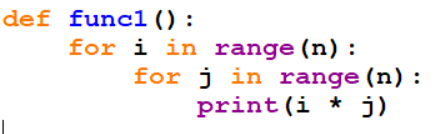
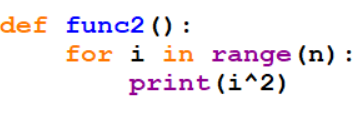
Level 5: Writing Searches

Answer the following questions:

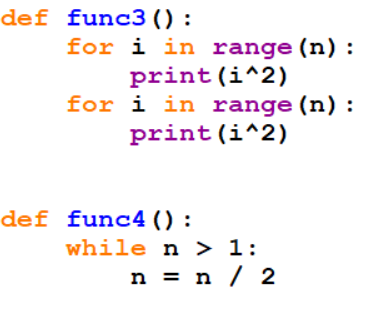
What is the big-o of the following functions:

****

**O(n^2)**



**O(n)**



**O(n)**

**And**

**O(1)**

What is the big-O programs with the following runtimes:

X = n^3 + 2n^2 + 12

**O(n^3)**

X = n \* log(n)

**O(n(log)(n)))**

X = 5

**O(1)**

**Part B: Programming**

Write 2 python functions for this assignment.

Your first should linearly search a list for a word and return the index of a word in a list of words **WITHOUT CALLING THE PREDEFINED SEARCH FUNCTIONS**. Your function should be able to check each word in the lis t in order to find it.

Your second function should use recursive binary search to find the index of the words. This is more challenging but a much faster algorithm. At each step, you will find the middle word in the list, and remove the half of the list which does not contain the word, until you reach the base case of a dictionary with 1 word, or find the word. You will need to keep track of your word index as you go so you can return the index at the end.

You should download the wordlist.py file from this directory and import it into your python program. You can access the word list with wordlist.words once it has been imported.

After the search algorithms are complete, add in the timer functionality from previous weeks to determine which algorithm runs faster in real time.

Sample words and indexes (if you give the word you should get the index):

a  -> 0

abandon -> 9

abbot -> 16

O(