

CS 106
Introduction to Computer Science I

07 / 14 / 2021

Instructor: Michael Eckmann

Today's Topics

- Questions / comments?
- More about Programmer-defined functions
- How to choose which loop to use and whether or not to use indices.

Programmer-defined functions

Let's rewrite the following but let's use for loops (without worrying about the indices)

count_words (return the number of words in a str)

number_of_positives

(return the # of positive #s in a list)

number_of_odds

(return the # of odd #s in a list)

Can write traverse a list or a str without worrying about the indices if you do not need them. Some problems will need the indices (the ones above do not). Let's try to write contains from the lab without indices?

Programmer-defined functions

Let's write a function that takes in a list (assumed to be numbers) and creates and returns a new list that is 1 shorter in length and each element is the sum of the corresponding one in the original list and the next one in the original list.

e.g. if this list: $[5, 8, 10, -3]$ is passed in, it should return $[13, 18, 7]$

Because $13 = 5 + 8$, $18 = 8 + 10$, $7 = 10 - 3$

This one is easier done with the indices.

Which loop to use in different situations

1. If don't know the number of times you will iterate:
 - Use a while loop when you don't know how many times the loop should iterate (e.g. loop until user types 'q')
2. If don't need indices
 - Use a for loop to traverse a list without indices: e.g. for item in mylist: (see count_words, numodds, numpositives on 7/14)
3. If need indices
 - Can use a while loop with an index variable to traverse a str or a list (see count_words, numodds, numpositives on 7/13)
 - But it is usually preferred to use a for loop for the same purpose like: for i in range(len(mylist)):

Programmer-defined functions

Suppose there's a lottery where the player chooses 1 number from 1 to 72. Grand prize is won if the player matches the first or second number drawn. Second prize if the player matches one of the next three numbers drawn. Nothing is won otherwise.

Let's write a function that takes in the player's number, a list of grand prize numbers, and a list of second prize numbers and returns 'Grand prize', 'Second prize', or 'Sorry'

Programmer-defined functions

Let's write a function that simulates rolling a 6 sided die.

Programmer-defined functions

Let's write a function that randomly recommends a Monty Python's Flying Circus episode to watch. According to wikipedia, there are 4 series (aka seasons) of the show, the first 3 of which have 13 episodes and the last has 6 episodes.

Expect a return string like this: We recommend you watch Monty Python's Flying Circus Series 2, Episode 11.

Searching

Given an item and a list, determine whether the item is in the list or not.

Linear search: compare the item to the first element of the list (if not ==), compare to the second and so on, until we find it (return True) or we go through the entire list and don't find it (return False)

This is like the contains function from lab and that we rewrote

Suppose we have a sorted list (from low to high) can we make fewer comparisons for search in the worst case and on average?