

15-110 Recitation Week 5

Reminders

- Check 3 due Monday 2/21 @ Noon EDT
- Check 2 and HW 2 revisions due Tuesday 2/22 @ Noon EDT
- Quiz 2 is Wednesday 2/23, same procedures as Quiz 1
 - Quiz 2 review session Sat 2/19 3-4PM NSH 3305
- Code reviews 2/18 - 2/22

Overview

- Lists
- 2D lists
- Recursion (code writing)
- Aliasing

Problems

LIST CODE WRITING: ALTERNATING SUM

Write a function `alternatingSum(L)` that takes in a list of numbers `L`, and then returns the alternating sum (where the sign switches from positive to negative or negative to positive at each index).

For example, `alternatingSum([5,3,8,4])` returns 6 as $(5-3+8-4) = 6$

See starter file for more tests and function header!

RECURSION INTRO

General notes on recursion:

Recreate the following function using recursion (write on the right empty space):

```
def double(lst):  
    result = []  
    for i in range(len(lst)):  
        result.append(2 * lst[i])  
    return result
```

```
#double([1,2,3]) -> [2,4,6]
```

```
def doubleRecursive(lst):
```

RECURSIVE CODE WRITING

Write the function `rangeSum(lo, hi)` which takes in two integers (where $lo \leq hi$) and sums all values in between them inclusive.

LIST ALIASING

Code trace and compare the following two options for ways to create “empty” 2D lists:

Option 1:

```
inner = [0, 0, 0, 0]
outer = [inner, inner, inner]
```

Option 2:

```
rows = 3
outer = []
for row in range(rows):
    outer.append([0, 0, 0, 0])
```

For each option, after running the code above, what are the values in outer?

Option 1: outer = _____

Option 2: outer = _____

After adding the following line of code and running it:

```
outer[0][0] = 42
```

What are the values in outer?

Option 1: outer = _____

Option 2: outer = _____

Be sure you can explain what difference you are seeing, and which option you should use and why.