15-110 Recitation Week 8

Reminders

- 10/25 Tue - Check3/HW3 revisions due (Tuesday after break)
- Have a restful and rejuvenating break!

Overview

- Big-O Exercise
- Tree Code Writing
- Dictionary Review
- Dictionary Code Writing

Problems

BIG-O EXERCISE

Calculate the Big-O for the following examples:

Returning the last character in a string	

`def powersOfTwo(n):`
`  m = 1`
`  while m <= n:`
`  print(m)`
`  m *= 2`	

`def foo(L): # len(L) = n`
`  if L == []:`
`  return 0`
`  else:`
`  L.append(L[0])`
`  n = L.index(10)`
`  L.pop(len(L)-1)`
`  return n`
`# .index(), .pop() are O(n) worst case!`	

`def tripleLoop(L):`
`  for i in range(20):`
`  for row in L:`
`  for elem in row:`
`  print(elem)`
`#You are guaranteed L is a nxn 2D list`	

TREE CODE WRITING

Write the function addEvenLeaves(t) that takes in a dictionary representation of a tree and returns a sum of **only** the even values held by leaves.

```
def addEvenLeaves(tree):
    # base case: if tree is empty
    if _____________:
        # What should you return if the tree is empty?
        return _____________
    # base case: leaf node
    if _________________ and _____________:
        # check if leaf's value is even
        if _____________:
           # returns the leaves value
           return _____________
        else:
           # what should you return if the leaf isn't even?
           return ____
    else:
        value = 0
        # recursive case if left subtree is not None
        if _____________:
            value += _____________________
        # recursive case if right subtree is not None
        if _____________:
            value += _____________________
        return value
```

DICTIONARY REVIEW

Notes on dictionaries:

Here is an example of a type of problem that uses dictionaries. Read through the problem statement and solution and note the key points of the code.

Problem:

Kelly's Bakery is doing an inventory of their freshly baked goods. This morning, they baked new items and now they need to update their inventory to represent these items. You are given a dictionary that represents the inventory at Kelly's Bakery, which maps the name of the item to how many items of that baked good are available. Write the function updateInventory(d, newItems) that takes the current inventory and a new dictionary called newItems and updates it accordingly. The function should also handle the case that there is an item in newItems that doesn't exist in d.

Solution:

```
def updateInventory(d, newItems):
  for item in newItems:
     if item in d:
        d[item] += newItems[item]
     else:
        d[item] = newItems[item]
  return d
```

DICTIONARY CODE WRITING

Given a list of wins by CMU, Pitt, OSU, PennState's, and another unspecified number of football teams, return the team with the most wins. There will be no ties. For example,

mostWins["CMU", "Pitt", "OSU", "OSU", "PennState", "OSU"]) returns "OSU".

We have provided the general form of the function, fill in the blanks with the code.

```
def mostWins(L):
    # Initialize an empty dictionary
    wins = dict()
    # Loop through the input list
    for _________________ in _________________:
        team = L[i]
        # What do we do if the team is in our dictionary?
        if team in wins:
           wins[team] = _________________
        # Otherwise?
        else:
           wins[team] = _________________
    # Initialize variables to store the team that has won
    # the most so far and how many times they had won
    mostWinTeam = _________________
    mostWins = _________________
    # Loop through the dictionary
    for _________________ in _________________:
       # What do we do if the current team has won more
       # than the team with the most wins so far?
       if _________________ > mostWins:
          mostWins = _________________
          mostWinTeam = _________________
    # Return the team that has won the most
    return mostWinTeam
```