New York University

CSC-101 Intro To Programming Language

Tutor: Raiyan Reza
Student Name: Ameen

Class Problem Set 3: Functions, Lists,Tuples, Dictionaries

July 24, 2022

Instructions

Unless stated otherwise assume you cannot access any additional materials or resources. You cannot use any IDE or programming environments such as VS Code. You may write your code using pen and paper, word processor such as MS Word, or plain text editor such as NotePad.

Functions, Lists, Tuples, Dictionaries

1. **Two Sum Problem**
   Given an array of integers **nums** and an integer **target**, return *indices of the two numbers such that they add up to* ***target***. You may assume that each input would have **exactly one solution**, and you may not use the *same* element twice. You can return the answer in any order.

   Example 1:
   Input:
   nums=$[2, 7, 11, 15]$, target=9
   Output:
   $[0, 1]$

   Example 2:
   Input:
   nums=$[3, 2, 4]$, target=6
   Output:
   $[0, 1]$

   Example 3:
   Input:
   nums=$[3, 3]$, target=6
   Output:
   $[0, 1]$

   (a) Produce a naive solution to this question using nested loops.

   (b) Produce a more time efficient solution by taking advantage of dictionaries in Python. The formalism of why using dictionaries is beyond the scope of this class. But an intuitive explanation is comprehensible. One of the features of dictionaries is that it can return values associated keys in a constant time regardless of how many entries are in dictionary. This simplifies searching for a specific item because you no longer have to go through multiple items before finding a specific item. For example, the list=[“Dhaka”, “Washington DC”, “ Paris”, “ Ottawa], of capital cities of different countries a passenger has visited. If you want to check if the passenger has ever been to Ottawa you need to iterate through the items in the list. Since Ottawa is last element of this list, you need to iterate through all the elements in a list. On the other hand if you are given a dictionary, The catch here is, this solution is less memory efficient.
   capital={“Bangladesh: Dhaka”,
   “USA: Washington DC”, “France: Paris”, “Canada: Otawa”},
   you just need to check if the key,value pair Canada, Ottawa is there. Dictionary lets you do this in constant time because there is no need to check all the other key,value pairs provided you know the key here, which is Canada.

2. **Sorting A List**
   You are given a function sortList(unsortedList:list[n])→ list[int]. Your input would be a list of integers that are unsorted. The function must returns a list that is ordered in ascending order.

   Example:

   Input: $[10, -2, 4, 1, 8]$

   Output: $[-2, 1, 4, 8, 10]$

3. **Pascal's Triangle**
   The following is a Pascal's triangle that displays the Pascal's triangle of height/depth 6. You should be able to infer the pattern and produce Pascal's Triangle of any arbitrary height/depth.

$n = 0$							1						
$n = 1$						1		1					
$n = 2$					1		2		1				
$n = 3$				1		3		3		1			
$n = 4$			1		4		6		4		1		
$n = 5$		1		5		10		10		5		1	
$n = 6$	1		6		15		20		15		6		1

   You need to define a function pascalTrianlge(n:int)→list[int] that accepts an integer $n \leq 0$ and returns a list of lists.

   Example
   Input: 3

   Output: $[[1], [1, 1], [1, 2, 1], [1, 3, 3, 1]]$