Interview Questions

Author: Jubin Soni ***Technical Questions Project: Data Analyst Nanodegree*** ***GitHub***

Question 1:

I am checking two strings are anagram by checking the count of all characters in both string. If all counts are the same, then the two strings are anagram. I first compile a dictionary of counts for t and check with every possible consecutive substring sets in s. If any set is anagram of t, then return True, else False. Comparing counts of all characters will can be done in constant time since there are only limited amount of characters to check (26 characters). Therefore, the time complexity of this algorithm is O(len(s)).

Question 2:

To find if the string contains longest palindrome or not, I have implemented Manachars algorithm since it is the most efficient way of solving this problem in O(n) linear time. I get the left position mirror for current right position and current right position is within the center right position, I attempt to expand the palindrome centered at current right position. If match is found then I increment the LPS length by one, if even position I just increment LPS by one. Time complexity: O(n).

Question 3:

I have created few helper functions find and union, the find function finds set of an element I and union function does union of two sets of x and y. I have used Kruskals Minimum spanning tree algorithm for solving this question. First, I sorted all the edges in non-decreasing order of their wright. Picked the smallest edge, check if it forms a cycle with the spanning tree. If the cycle is formed added this edge in a dictionary else discarded it. I only did this till V-1 edges as per the algorithm. Time complexity: O(E log V).

Question 4:

In this question we have to find the least common ancestor so first I have written find parent utility function, that returns parent if found in the 2x2 matrix else it returns -1. Then I search for n1's parent and keep adding all elements in my list until I find the root, r. Then I find the parent of n2 and check if any parent of n2 is also in the parent list I created for n1, if parent exists that is the LCA and I return that value. Time complexity: O(n).

Question 5:

This was easy question, first I create a Node and a Linked list. In the question5(ll, m), I initialize the linkedlist and push elements at the beginning (head) of the linked list. After all elements are added, I reverse the linkedlist and then do linkedlist traversal with a counter. Once the counter matches m element, I return that mth element. Time complexity: O(n).

References:

- Udacity Interview Prep Course
- GeeksforGeeks
- Stackoverflow
- TutorialsPoint: Data structures course