

Interview Questions

Author: Jubin Soni

Technical Questions Project: Data Analyst Nanodegree

[GitHub](#)

Question 1:

In this question my approach was to first generate all permutations of the sub-string (t), then if any of those substring (t) permutations match the string (s) my functions returns True else False. For solving this, I created a helper function called permutations that uses recursion to compute all the permutations of a string.

Question 2:

To find if the string contains longest palindrome or not, I am breaking each string into characters and loop from first character till the end left to right. If the reverse of the substrings match each other it's a palindrome. Then I store the first palindrome into another variable called longest_palindrome, if the next palindrome found is of greater length than the longest_palindrome. I replace it with the longest one. I created two helper functions palindromize that matches reverse of the string is similar to the string or not, and reverse function computes reverse of any string using recursion.

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 weight. 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.

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.

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.

References:

- [Udacity Interview Prep Course](#)
- [GeeksforGeeks](#)
- [Stackoverflow](#)
- [TutorialsPoint: Data structures course](#)