Han Zhu's Study Notes

Mind nodes, path-finding algorithms, and random garbage collections.

[LeetCode] 115. Distinct Subsequences

May 28, 2017

Given a string **S** and a string **T**, count the number of distinct subsequences of **T** in **S**.

A subsequence of a string is a new string which is formed from the original string by deleting some (can be none) of the characters without disturbing the relative positions of the remaining characters. (ie, `"ACE"` is a subsequence of `"ABCDE"` while `"AEC"` is not).

Here is an example:
S = `"rabbbit"`, **T** = `"rabbit"`

Return `3`.

Thought process:

1. Sub-problem: count the number of distinct subsequences of a substring of T and a substring of S.
2. Function:
    1. If the last character of S and T are not the same, the last character of S will not contribute to the subsequences. f[i][j] = f[i - 1][j].
    2. Otherwise, we add the number of subsequences where the last character of S is in. f[i][j] = f[i - 1][j] + f[i - 1][j - 1].
3. Initialization:
    1. If T is empty, there is one way to get an empty string out of S, which is to choose nothing. f[i][0] = 1.
    2. If S is empty and T is not, there is no way to get T out of S. f[0][j] = 0.
4. Answer: f[s.length][t.length].

Solution:

```
public class Solution {
    public int numDistinct(String s, String t) {
        int sLength = s.length();
        int tLength = t.length();

        int[][] f = new int[sLength + 1][tLength + 1];

        for (int i = 0; i <= sLength; i++) {
            f[i][0] = 1;
        }

        for (int i = 1; i <= sLength; i++) {
            for (int j = 1; j <= tLength; j++) {
                if (s.charAt(i - 1) == t.charAt(j - 1)) {
                    f[i][j] = f[i - 1][j] + f[i - 1][j - 1];
                } else {
                    f[i][j] = f[i - 1][j];
                }
            }
        }

        return f[sLength][tLength];
    }
}
```

Time complexity:

Say m = t.length and n = s.length, the overall time complexity is O(mn).

DP LeetCode String

Location: San Jose, CA, USA

Popular posts from this blog

[LeetCode] 269. Alien Dictionary

June 29, 2017

There is a new alien language which uses the latin alphabet. However, the order among letters are unknown to you. You receive a list of non-empty words from the dictionary, where words are sorted lexicographically by the rules of this new language . Derive t …

READ MORE

[LeetCode] 253. Meeting Rooms II

March 11, 2017

Given an array of meeting time intervals consisting of start and end times [[s1,e1],[s2,e2],...] (s i < e i), find the minimum number of conference rooms required. For example, Given [[0, 30],[5, 10],[15, 20]] , return 2 . Thought process: The idea is to first …

READ MORE

[LeetCode] 261. Graph Valid Tree

July 29, 2017

Given n nodes labeled from 0 to n - 1 and a list of undirected edges (each edge is a pair of nodes), write a function to check whether these edges make up a valid tree. For example: Given n = 5 and edges = [[0, 1], [0, 2], [0, 3], [1, 4]] , return true . Given n = 5 …

READ MORE