Problem 115: Distinct Subsequences

Problem Information

Difficulty: Hard
Acceptance Rate: 51.06%
Paid Only: No
Tags: String, Dynamic Programming

Problem Description

Given two strings s and t, return _the number of distinct_ **_subsequences_** _of_ s _which equals_ t.

The test cases are generated so that the answer fits on a 32-bit signed integer.

Example 1:

Input: s = "rabbbit", t = "rabbit" **Output:** 3 **Explanation:** As shown below, there are 3 ways you can generate "rabbit" from s. **_rabb_** b** _it_** **_ra_** b** _bbit_** **_rab_** b** _bit_**

Example 2:

Input: s = "babgbag", t = "bag" **Output:** 5 **Explanation:** As shown below, there are 5 ways you can generate "bag" from s. **_ba_** b _**g**_ bag **_ba_** bgba** _g_** _**b**_ abgb** _ag_** ba _**b**_ gb _**ag**_ babg** _bag_**

Constraints:

* `1 <= s.length, t.length <= 1000` * `s` and `t` consist of English letters.

Code Snippets

C++:

```
class Solution {
public:
int numDistinct(string s, string t) {

}
};
```

Java:

```
class Solution {
public int numDistinct(String s, String t) {

}
}
```

Python3:

```
class Solution:
def numDistinct(self, s: str, t: str) -> int:
```