

代码中的一些细节

1. 字符串前添加空格

- 通过在字符串前添加空格，使字符串的索引从 1 开始，方便处理 DP 数组，因为数组 `dp[i][j]` 与 `t[i]` 和 `s[j]` 对应。

2. 使用 double 类型

- 在代码中，dp 数组被定义为 `vector<vector<double>>`，使用了 double 类型。
- **注意：**在计数问题中，使用浮点数可能会引入精度问题，通常建议使用整数类型。
- **更好的选择：**使用 `long long` 类型来避免整数溢出，同时保持精度。

代码：

```
1 class Solution {
2 public:
3     int numDistinct(string s, string t) {
4         int n = s.size();
5         int m = t.size();
6         s = " " + s;
7         t = " " + t;
8         vector<vector<double>> dp(m + 1, vector<double>(n + 1, 0));
9         for (int i = 0; i <= n; i++) {
10             dp[0][i] = 1;
11         }
12         for (int i = 1; i <= m; i++) {
13             for (int j = 1; j <= n; j++) {
14                 dp[i][j] = dp[i][j - 1];
15                 if (t[i] == s[j])
16                     dp[i][j] += dp[i - 1][j - 1];
17             }
18         }
19         return dp[m][n];
20     }
21 };
22
```