

## Distinct Subsequences : 90 分解答。

类似与去年查找公共子序列的 dp 题，这次是查找子序列的个数。用  $dp[i][j]$  表示匹配串 T 的前 i 个与主串 S 的前 j 个匹配成功的个数。

那么可以得到递推方程：

当  $t[i - 1] == s[j - 1]$  时： $dp[i][j] = dp[i][j - 1] + dp[i - 1][j - 1]$

当 不相等 时： $dp[i][j] = dp[i][j - 1]$

如图所示

		0	n	j	n	u	n	j	u
		0	1	2	3	4	5	6	7
	0	0	1	1	1	1	1	1	1
n	1	0	1	1	2	2	3	3	3
j	2	0	0	1	1	1	4	4	4
u	3	0	0	0	0	1	1	1	5

C++部分代码如下：

```
int judge(string s, string t)
{
    vector<vector<int>> dp(t.size() + 1, vector<int>(s.size() + 1, 0));
    int i, j;
    for (i = 0; i <= s.size(); i++)
        dp[0][i] = 1;
    for (i = 1; i <= t.size(); i++)
    {
        for (j = i; j <= s.size(); j++)
        {
            if (t[i - 1] == s[j - 1])
                dp[i][j] = dp[i][j - 1] + dp[i - 1][j - 1];
            else
                dp[i][j] = dp[i][j - 1];
        }
    }
    return dp[t.size()][s.size()];
}
```