如何写出状态转移方程

原创 蓝鑫 LOA算法学习笔记 2021-01-15 15:21

01 方法

通常采用三步法，一是定含义，二是找关系，三是找初值。

1) 定含义

通常会使用一维数组或二维数组来保存历史信息，那么需要明确一维数组dp[i]，或是二维数组dp[i][j]代表的是什么含义。

2) 找关系

动态规划有点类似于归纳法（找规律），最后需要利用dp[n-1],dp[n-2],..., dp[1]来总结出与dp[n]之间的关系。

3) 找初值

假设在第二步中已经找出递推关系为：dp[n]=dp[n-1]+dp[n-2]，那么追溯到最开始，得出的关系式为dp[3]=dp[2]+dp[1]，且dp[2]和dp[1]不可再分解，那么初值就应该是dp[2]和dp[1]。

02 举例

- **例一(一维数组)**

一只青蛙一次可以跳上1级台阶，也可以跳上2级。求该青蛙跳上一个n级的台阶总共有多少种跳法。

1) 定含义

dp[i]的含义：跳上一个i级台阶总共有dp[i]种跳法。

2) 找关系

青蛙可以选择跳1级台阶，也可以是跳2级台阶，所以到达第n级台阶时，青蛙有可能是从第n-1个台阶跳上去的，也可能是从第n-2个台阶跳上去的，本题需要计算所有可能的跳法，所以是dp[n]=dp[n-1]+dp[n-2]。

3) 找初值

当n=1时，dp[1]=dp[0]+dp[-1]，因为n不允许为负数，所以说dp[1]需要直接给出它的数值，dp[1]=1。

关于dp[0],暂时当作跳0级台阶有0种跳法，也是需要直接给出其数值的，dp[0]=0。

当n=2时，dp[2]=dp[0]+dp[1]=1，但事实上，跳到第2个台阶，可以从第0个台阶跳2级，也可以是第1个台阶跳1级，共有2种跳法，**出现矛盾**，所以正确的应该是dp[2]=2

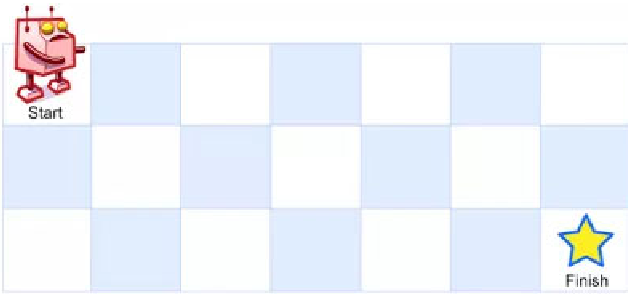
4) 状态转移方程

总结步骤(2)(3)，可以得出：

$$dp[n]=\begin{cases} n & n=0,1,2 \\ dp[n-1]+dp[n-2] & n>2 \end{cases}$$

- **例二（二维数组）**

一个机器人位于一个 m * n 网格的左上角（起始点在下图中标记为 “Start”）机器人每次只能向下或者向右移动一步。机器人试图达到网格的右下角（在下图中标记为“Finish”）。问总共有多少条不同的路径?



1) 定含义

dp[i][j]的含义：当机器人走到(i,j)这个位置时，总共有dp[i][j]条不同的路径。

2) 找关系

机器人可以向下移动，或是向右移动，所以机器人到达位置(i,j)有两种方式，一种是从位置(i-1,j)向右走一步，另一种是从位置(i,j-1)向下走一步到达。因为是计算所有不同路径的数量，所以dp[i][j]=dp[i-1][j]+dp[i][j-1]。

3) 找初值

当i或j为0时，i-1或j-1就为负数，不符合要求，所以dp[0][j]=1,相当于网格的第0行，机器人只能往右走；同理dp[i][0]=1,相当于网格的第0列，机器人只能往下走。

4) 状态转移方程

总结步骤(2)(3)，可以得出：

$$dp[i][j]=\begin{cases}1 & i=0 \text{ 或 } j=0 \\ dp[i-1][j]+dp[i][j-1] & else\end{cases}$$

03 小结

通常动态规划的状态方程都可以通过以上三个步骤得出，步骤一中dp的含义往往与题目的问题是密切相关的。需要注意的地方是步骤二中的初值确定，正如例一中，稍有不慎就会遗漏n=2这种情况。在写出状态转移方程后如果不放心，可以将自己写的方程再代入原题，设置具体的数值进行验证。以上为个人总结，欢迎批评指正。