

96 不同的二叉搜索树

题目:

给定一个整数 n ，求以 $1 \dots n$ 为节点组成的二叉搜索树有多少种？

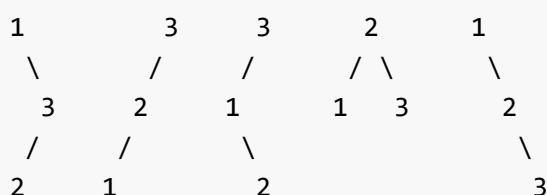
示例:

输入: 3

输出: 5

解释:

给定 $n = 3$ ，一共有 5 种不同结构的二叉搜索树:



思路:

采用动态规划的思路求解

设要求的以 1 到 n 为节点组成的二叉搜索树的数目为 $G(n)$ 。分析可知， 1 到 n 的任何一个节点都可以作为根节点，令以节点 i 为根节点的二叉搜索树的数目为 $f(i)$ 。于是有：

$$G(n) = f(1) + f(2) + f(3) + f(4) + \dots + f(n)$$

两种特殊情况是 n 为 0 和 n 为 1 的情况，这两种情况对应的结果都为 1 。即：

$$G(0) = G(1) = 1$$

然而， $f(i)$ 也与 G 函数有关系。 $f(i)$ 代表的以 i 节点为根的可以生成的二叉搜索树的个数，等于其左子树个数和右子树个数的乘积。左子树的节点个数为 $i-1$ ，右子树的节点个数为 $n-i$ 。而且， $G(n)$ 和序列的内容无关，只和序列的长度有关，因此有：

$$f(i) = G(i-1) * G(n-i)$$

综合以上公式，可以得到卡特兰数公式：

$$G(n) = G(0) * G(n-1) + G(1) * (n-2) + \dots + G(n-1) * G(0)$$

代码:

```

class Solution {
    public int numTrees(int n) {
        int[] G = new int[n+1];
        G[0] = 1;
        G[1] = 1;
        for(int i = 2; i <= n; i++)
  
```

```
        for(int j = 1; j <= i; j++)
            G[i] += G[j - 1] * G[i - j];
    return G[n];
}
```

G(n)语句的执行次数为：

$$\sum_{i=2}^n i = \frac{(2+n)(n-1)}{2}$$

因此时间复杂度为 $O(n^2)$

空间复杂度为 $O(n)$