

斐波那契数列(递归次数分析)

1. 项数为 1-46 时递归函数的执行次数

项数	递归次数	项数	递归次数
1	1	24	92735
2	1	25	150049
3	3	26	242785
4	5	27	392835
5	9	28	635621
6	15	29	1028457
7	25	30	1664079
8	41	31	2692537
9	67	32	4356617
10	109	33	7049155
11	177	34	11405773
12	287	35	18454929
13	465	36	29860703
14	753	37	48315633
15	1219	38	78176337
16	1973	39	126491971
17	3193	40	204668309
18	5167	41	331160281
19	8361	42	535828591
20	13529	43	866988873
21	21891	44	1402817465
22	35421	45	2269806339
23	57313	46	3672623805

2. 递归函数执行次数的递推公式

设 $Count(n)$ 表示斐波那契项数为 n 时，递归函数的执行次数。

$$Count(n) = \begin{cases} 1 & (n = 1, 2) \\ Count(n-1) + Count(n-2) + 1 & (n > 2) \end{cases}$$

3. 公式推导

求斐波那契数列的函数如下：

```
int fibonacci(int n)
{
    if (n == 1 || n == 2)
        return 1;
    return fibonacci(n - 1) + fibonacci(n - 2);
}
```

若 $n = 1$ 或 $n = 2$ ，函数只执行一次，返回 1。

若 $n > 2$ ， $fibonacci(n)$ 会调用 $fibonacci(n-1)$ 和 $fibonacci(n-2)$ ，

而 $fibonacci(n-1)$ 和 $fibonacci(n-2)$ 的执行次数分别为 $Count(n-1)$ 和 $Count(n-2)$

加上 $fibonacci(n)$ 这一次，

所以 $Count(n) = Count(n-1) + Count(n-2) + 1$ 。

综上所述，

$$Count(n) = \begin{cases} 1 & (n = 1, 2) \\ Count(n-1) + Count(n-2) + 1 & (n > 2) \end{cases}$$