

### Comparison between Recursive and Dynamic Solutions to Towers of Hanoi

K	Dynamic	Recursive
2	700	700
3	1200	1500
4	1500	3100
5	1800	6300
6	2100	12700
7	2400	25500
8	2700	51100
9	3000	102300
10	3300	204700
11	3600	409500
12	3900	819100
13	4200	1638300
14	4500	3276700
15	4800	6553500
16	5100	13107100
17	5400	26214300
18	5700	52428700
19	6000	104857500

**Figure 1.** Number of recursive calls for each K.

There are pretty clear patterns for both the dynamic and recursive columns. Besides the jump from K=2 to K=3, every other increase in K involves exactly 300 more recursive calls. When it comes to the recursive algorithm the pattern is a little less obvious. As K increases by 1 the number of calls doubles and increases by 100,  $K_a = 2K_{(a-1)} + 100$ .

The recursive algorithm doesn't take what it has already done into account which is why it calls essentially double the previous K. When you program with dynamic programming techniques, you require exponentially less recursive calls because many of them are redundant.