



Online Judge	Problem Set	Authors	Online Contests	User
<a href="#">Web Board</a> <a href="#">Home Page</a> <a href="#">F.A.Qs</a> <a href="#">Statistical Charts</a>	<a href="#">Problems</a> <a href="#">Submit Problem</a> <a href="#">Online Status</a> Prob.ID: <input type="text"/> <input type="button" value="Go"/>	<a href="#">Register</a> <a href="#">Update your info</a> <a href="#">Authors ranklist</a> <input type="text"/> <input type="button" value="Search"/>	<a href="#">Current Contest</a> <a href="#">Past Contests</a> <a href="#">Scheduled Contests</a> <a href="#">Award Contest</a>	<a href="#">lydliyudong</a> <a href="#">Log Out</a> <a href="#">Mail:5(0)</a> <a href="#">Login Log</a> <a href="#">Archive</a>

## Strange Towers of Hanoi

Language:  ▾

**Time Limit:** 1000MS **Memory Limit:** 30000K  
**Total Submissions:** 2886 **Accepted:** 1873

## Description

### Background

Charlie Darkbrown sits in another one of those boring Computer Science lessons: At the moment the teacher just explains the standard Tower of Hanoi problem, which bores Charlie to death!



Figure 4: The standard (three) Towers of Hanoi.

The teacher points to the blackboard (Fig. 4) and says: "So here is the problem:

- There are three towers: A, B and C.
- There are  $n$  disks. The number  $n$  is constant while working the puzzle.
- All disks are different in size.
- The disks are initially stacked on tower A increasing in size from the top to the bottom.
- The goal of the puzzle is to transfer all of the disks from tower A to tower C.
- One disk at a time can be moved from the top of a tower either to an empty tower or to a tower with a larger disk on the top.

So your task is to write a program that calculates the smallest number of disk moves necessary to move all the disks from tower A to C."

Charlie: "This is incredibly boring—everybody knows that this can be solved using a simple recursion. I deny to code something as simple as this!"

The teacher sighs: "Well, Charlie, let's think about something for you to do: For you there is a fourth tower D. Calculate the smallest number of disk moves to move all the disks from tower A to tower D using all four towers."

Charlie looks irritated: "Urgh. . . Well, I don't know an optimal algorithm for four towers. . . "

### **Problem**

So the real problem is that problem solving does not belong to the things Charlie is good at. Actually, the only thing Charlie is really good at is "sitting next to someone who can do the job". And now guess what — exactly! It is you who is sitting next to Charlie, and he is already glaring at you.

Luckily, you know that the following algorithm works for  $n \leq 12$ : At first  $k \geq 1$  disks on tower A are fixed and the remaining  $n-k$  disks are moved from tower A to tower B using the algorithm for four towers. Then the remaining  $k$  disks from tower A are moved to tower D using the algorithm for three towers. At last the  $n - k$  disks from tower B are moved to tower D again using the algorithm for four towers (and thereby not moving any of the  $k$  disks already on tower D). Do this for all  $k \in \{1, \dots, n\}$  and find the  $k$  with the minimal number of moves.

So for  $n = 3$  and  $k = 2$  you would first move 1 ( $3-2$ ) disk from tower A to tower B using the algorithm for four towers (one move). Then you would move the remaining two disks from tower A to tower D using the algorithm for three towers (three moves). And the last step would be to move the disk from tower B to tower D using again the algorithm for four towers (another move). Thus the solution for  $n = 3$  and  $k = 2$  is 5 moves. To be sure that this really is the best solution for  $n = 3$  you need to check the other possible values 1 and 3 for  $k$ . (But, by the way, 5 is optimal. . . )

## **Input**

There is no input.

## **Output**

For each  $n$  ( $1 \leq n \leq 12$ ) print a single line containing the minimum number of moves to solve the problem for four towers and  $n$  disks.

## **Sample Input**

No input.

## **Sample Output**

REFER TO OUTPUT.

## **Source**

[\[Submit\]](#) [\[Go Back\]](#) [\[Status\]](#) [\[Discuss\]](#)



[Home Page](#)



[Go Back](#)



[To top](#)

---

All Rights Reserved 2003-2013 Ying Fuchen,Xu Pengcheng,Xie Di  
Any problem, Please [Contact Administrator](#)