Tower of Hanoi

The Towers of Hanoi puzzle consists of three pegs and a number of disks. The disks slide up and down on the pegs and can be moved from peg to peg, and are all different sizes.

The puzzle starts with all the disks in a pyramid on one peg, stacked from largest on the bottom to smallest on the top. The objective of the puzzle is to move the entire stack to another peg, obeying the following rules:

- Only one disk may be moved at a time.
- Each move consists of taking the upper disk from one of the pegs and sliding it onto another peg, on top of the other disks that may already be present on that peg.
- No disk may be placed on top of a smaller disk.

There are several applets on the Web that will let you play this game. One to try: [http://www.softschools.com/games/logic_games/tower_of_hanoi/](http://www.softschools.com/games/logic_games/tower_of_hanoi/)

It turns out (though it is not always obvious!) that it is always possible to move the stack from one peg to another, no matter how many rings you start with in the original stack. The interesting question then becomes how efficiently you can solve the puzzle.

1. What is the least number of moves it will take if there are only two disks in the original stack?
   Three moves—one to move the small disk, one to move the larger disk, and one last to put the small disk back on top.

2. What if there are three disks in the original stack?
   7 is the best

3. How about four? How about if there is only one disk?
   Four disks take 15 moves; one disk takes one move!

4. Would it make sense to talk about the puzzle with zero disks? How many moves would it take to “solve” the puzzle in this case?
   If there are no disks, the “puzzle” is already solved—it doesn’t take any moves to solve it!
5. Make a table (leave entry blank if you aren’t sure of the number of moves):

<table>
<thead>
<tr>
<th>Number of disks</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fewest number of moves needed</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>7</td>
<td>15</td>
<td>31</td>
<td>63</td>
</tr>
</tbody>
</table>

6. You might not have filled in the entire table. Makes reasonable guesses about any missing entries. Circle them to remind yourself they are guesses and not real data.

7. Looking at the numbers, especially the ones you are sure of, do you see a pattern? Is there a formula? If \( n \) is the number of rings in the original stack, and \( H(n) \) is the minimum number of moves needed to transfer the stack to a different peg, then a formula is:

\[
H(n) = 2^n - 1
\]

8. The Legend of the Tower tells of a monastery where monks have to move a stack of 64 golden disks among three diamond needles, and that when they complete their task the world will come to an end. If the monks were able to make one move each second, then about how long would we have according to your formula until the world’s end if they started today?

Using the formula above, \( 2^{64} - 1 = 18446744073709551615 \). That’s a lot of seconds. Dividing by 60 gives about \( 3.07 \times 10^{17} \) minutes, or about \( 5.1 \times 10^{15} \) hours, \( 2.1 \times 10^{14} \) days, or about 585 billion years. This is approximately 40 times the current estimated age of the universe!

9. If the world was supposed to end in 50,000 years, how many disks should the monks have started with?

50,000 years is 18,250,000 days, or 438 million hours or about 26 billion minutes, or about 1.6 trillion seconds. So we need \( 2^n \) to be about 1.6 trillion. To solve \( 2^n = 1.6 \text{ trillion} \), take \( \log_2 \) of both sides, 

\[ n = \log_2(1.6 \text{ trillion}) \]

which is about 40 or 41 disks.

(Fun connection: read the short story *The Nine Billion Names of God* by Arthur C. Clarke.)