# Some Variations of the Hanoi Tower Problem
# Homework for ITT9131 Concrete Mathematics

Ahto Truu

September 23, 2016

## 1 Introduction

The Tower of Hanoi is a classic puzzle invented by Edouard Lucas in 1883.

We are given a tower of $n$ disks, each of a different size, initially stacked in decreasing order on one of three pegs. The objective is to transfer the entire tower to one of the other pegs, moving only one disk at a time and never placing a larger disk onto a smaller one.

It is well known that $T(n)$, the minimum number of steps needed to solve the $n$-disk problem, is given by the recurrent equations

$$\begin{cases} T(1) = 1 \\ T(n) = T(n-1) + 1 + T(n-1) \qquad \text{for } n \geq 2, \end{cases}$$

which yield the closed-form solution

$$T(n) = 2^n - 1 \qquad \text{for } n \geq 1.$$

## 2 The Double Tower

Next we consider a variation given as Exercise 1.11 in the course textbook.

A Double Tower of Hanoi contains $2n$ disks of $n$ different sizes, exactly two of each size. As before, we can move only one disk at a time and must never place a larger one onto a smaller one.

- How many steps does it take to transfer a double tower from one peg to another, if disks of equal size are indistinguishable from each other?
- What if we are required to reproduce the original order of all the equal-size disks in the final arrangement?

Let the number of steps required to move a tower of $2n$ disks be $D(n)$ for the first and $C(n)$ for the second case.

### 2.1 The Case of the Identical Twins

A common sense solution to the first version of the double tower problem is to notice that we can apply the algorithm for solving the single tower problem by viewing each pair of equal-size disks as a group that takes two steps to move. This implies that $D(n)$ can't exceed $2T(n)$. However, it's not clear wheteher this is the optimal solution, so a more rigorous derivation is needed.

We clearly have $D(1) = 2$, as we can simply move the two equal-size disks directly from the source to the destination one at a time and we obviously can't move two disks in less than two steps.

For $n > 1$, we can observe that we can solve the problem in $D(n-1) + 2 + D(n-1)$ steps by first moving the $2n-2$ smaller disks from the source to the auxiliary peg in $D(n-1)$ steps, then the two largest disks directly from the source to the destination peg in two steps, and finally the $2n-2$ smaller disks from the auxiliary to the destination peg in another $D(n-1)$ steps.

This is also the best we can do. First, we can't move the largest disks until we have removed to $2n-2$ smaller ones initially on top of them, so they can't move earlier than after we have already spent at least $D(n-1)$ steps. Likewise, we can't place any of the $2n-2$ smaller disks on their final destinations before the two largest disks are below them, so we have to spend at least $D(n-1)$ steps after both of the largest disks are in place. And similarly to the $n = 1$ case, we can't move the two largest disks in less than two steps.

So, we have the equations

$$\begin{cases} D(1) = 2 \\ D(n) = D(n-1) + 2 + D(n-1) \qquad \text{for } n \geq 2, \end{cases}$$

which are suspiciously similar to the ones we had for $T(n)$, except both the constant value in the base case and also the constant to be added in each recurrent step are exactly twice as big. So, it seems the common sense was right after all and indeed

$$D(n) = 2T(n) = 2(2^n - 1) = 2^{n+1} - 2 \qquad \text{for } n \geq 1.$$

Let's now prove this by induction:

Base: $D(1) = 2^2 - 2 = 2$, as it should be.

Step: Assume $D(n) = 2^{n+1} - 2$. Then

$$\begin{aligned} D(n+1) &= D(n) + 2 + D(n) \\ &= (2^{n+1} - 2) + 2 + (2^{n+1} - 2) \\ &= 2^{n+1} + 2^{n+1} - 2 \\ &= 2^{n+2} - 2, \end{aligned}$$

which again matches.

## 2.2 The Case of the Fraternal Twins

Starting to consider the second version of the double tower problem, where the relative order of the equal-size disks must be reproduced on the destination peg, we can first obtain $C(1) = 3$.

We can transfer the two equal-sized disks in 3 steps by first moving the upper one from the source to the auxiliary peg, then the lower one directly from the source to the destination peg, and finally the upper one from the auxiliary to the destination peg. We also can't do better than that, as the only way to move two disks in two steps would be to move both of them directly from the source to the destination peg and then they would end up in the wrong order on the destination peg.

Looking at the case where $n > 1$, we can observe that the solution we found for the first version of the double tower problem only swaps the two largest disks and all others actually maintain their original order.

Indeed, the two largest disks are swapped, as the upper one is moved first and ends up being the lower one on the destination peg. However, the second-largest pair is flipped twice: once when the $2n-2$ smaller disks are moved from the source to the auxiliary peg, and then once more when they are moved from the auxiliary to the destination peg.

In general, it is not hard to see that the $k$-th largest pair is flipped exactly $2^{k-1}$ times, but this is even unnecessary. It is sufficient to notice that whatever number of times a pair is flipped during the move from the source peg to the auxiliary peg, it is flipped again the same number of times during the move from the auxiliary peg to the destination peg, for a total of even number of flips.

This means that we have two viable strategies for solving larger instances of the second version of the problem.

In the first strategy, we move the lowest disk directly from the source to the destination peg. To achieve that, we have to move the $2n-1$ top disks from the source to the auxiliary peg in $C'(n)$ steps, then the lowest disk directly from the source to the destination peg in one step and then the $2n-1$ top disks from the auxiliary to the destination peg in another $C'(n)$ steps, for a total of $C'(n)+1+C'(n)$ steps. The reason this plan is optimal implementation of the chosen strategy is the same as why the algorithm for solving the regular (single tower) version of the problem is optimal.

In a similar vein, each of the two moves of the $2n-1$ top disks from peg $A$ to peg $B$ has to be done by moving the $2n-2$ smaller disks from $A$ to the third peg in $D(n-1)$ steps, then the largest disk to $B$ in one step, and then the $2n-2$ smaller disks from the third peg to $B$ in $D(n-1)$ steps, for a total of $D(n-1)+1+D(n-1)$ steps. Note that we perform the process of moving the $2n-2$ smaller disks an even number of times, which means they will end up back in the initial order.

The total number of steps needed under this strategy is

$$\begin{aligned} C(n) &= C'(n)+1+C'(n) \\ &= (D(n-1)+1+D(n-1))+1+(D(n-1)+1+D(n-1)) \\ &= 4D(n-1)+3 \\ &= 4\cdot(2^n-2)+3 \\ &= 2^{n+2}-5 \qquad \text{for } n \geq 2. \end{aligned}$$

In the second strategy, we move the lowest disk from the source to the destination via the auxiliary peg. The only way this detour could be useful is if it would let us move the two largest disks as a block.

Thus, we have to move the $2n-2$ smaller disks from the source to the destination peg in $D(n-1)$ steps, then the two largest disks from the source to the auxiliary peg in two steps (flipping them in the process), then the $2n-2$ smaller disks back from the destination to the source peg in $D(n-1)$ steps, then the two largest disks from the auxiliary peg to the destination peg in two steps (flipping them back to the original order), and finally we have to move the $2n-2$ smaller disks from the source peg to the destination peg in $C(n-1)$ steps to ensure their correct order.

The total number of steps needed under this strategy is

$$\begin{aligned} C(n) &= D(n-1) + 2 + D(n-1) + 2 + C(n-1) \\ &= (2^n - 2) + 2 + (2^n - 2) + 2 + C(n-1) \\ &= 2^n + 2^n + C(n-1) \\ &= 2^{n+1} + C(n-1) \qquad \text{for } n \geq 2, \end{aligned}$$

which we can solve by repeated substitution until we reach the base case

$$\begin{aligned} C(n) &= 2^{n+1} + C(n-1) \\ &= 2^{n+1} + 2^n + C(n-2) \\ &\dots \\ &= 2^{n+1} + 2^n + \ldots + 2^3 + C(1) \\ &= 2^3(2^{n-2} + 2^{n-3} + \ldots + 1) + C(1) \\ &= 2^3(2^{n-1} - 1) + 3 \\ &= 2^{n+2} - 5 \qquad \text{for } n \geq 2. \end{aligned}$$

Surprisingly, both strategies turn out to have the exact same cost! And since the expressions derived for $n > 1$ give the correct answer also for $n = 1$, we can simplify the whole result to

$$C(n) = 2^{n+2} - 5 \qquad \text{for } n \geq 1.$$