ISSN 2182-1976

Recreational Mathematics Magazine

Number 7
May, 2017

DE GRUYTER OPEN

Ludus

CIUHCT

Games and Puzzles

SEVERAL BOUNDS FOR THE k-TOWER OF HANOI PUZZLE

Stephen B. Gregg, Britney Hopkins, Kristi Karber, Thomas Milligan, Johnny Sharp

University of Central Oklahoma

TMilligan1@uco.edu

Abstract: *We consider special cases of a modified version of the Tower of Hanoi puzzle and demonstrate how to find upper bounds on the minimum number of moves that it takes to complete these cases.*

Keywords: Tower of Hanoi Puzzle, minimum number of moves.

Introduction

The Tower of Hanoi puzzle is a game that is played with disks of graduated sizes on three pegs. When the game begins, the disks are arranged on the first peg according to their size. The largest disk is located at the bottom of the peg and above it are all of the other disks arranged in increasing size. The smallest disk is therefore at the top of the peg. One must relocate all of the disks to the third peg, but there are strict rules about how the disks are moved. Moving one disk at a time, and never placing a larger disk on top of a smaller one, the player slowly makes their way to the end of the puzzle, with the stack of disks back in their original order on the third peg. The number of moves that it takes to complete the puzzle grows exponentially with the number of starting disks. It is for this reason, that when the priests at the temple of Benares were asked to complete the same task, but with sixty-four disks, the world would end at their completion. The legend speaks of sixty-four disks of gold being placed on a brass plate with diamond needles acting as the pegs. The priests were required to move each disk, one at a time, until the disks were back in their original order on the third diamond peg. Using the same condition that no gold disk ever be placed upon a smaller one, their task would exceed 500 billion years before finishing.

Suppose we relaxed the rule demanding the priests not place a golden disk on top of a smaller one by only requiring the bottom disk in any stack to be the largest.

Then how long would it take the priests to finish their divine task? What about if we only required the bottom two disks in any stack be the largest and allowed them to place any smaller disks on top of those? Furthermore, suppose we continue this and ask how long it will take to complete this puzzle if we allow a disk to be placed on top of a smaller disk, provided that the k disks on the bottom of the stack be the largest in the stack such that those k disks are arranged from largest on bottom, increasing in size to the kth disk? It is this question that is of interest in this paper. The resulting puzzle will take longer to solve for larger values of k, and it can be solved relatively quickly for smaller values of k. We will discuss several upper bounds on the optimal solutions for the minimum number of moves that it takes to complete these type of puzzles.

In Section 2, we discuss the fundamentals of the approach we will take to obtain our upper bounds. Next, in Section 3, we give upper bounds for some specific cases when the value k depends on the number of starting disks, n. Then in Section 4, we give an upper bound for the case whenever k is three and the number of starting disks can be any value greater than or equal to k. Finally, in Section 5, we discuss our future work to give a bound for an arbitrary value of k and an arbitrary number of disks, n.

Preliminary

The structure of the proofs for the upper bounds of the modified Tower of Hanoi puzzles (called k-Tower of Hanoi puzzles) is based on the concept of building stacks of disks, from the n starting disks. Because the goal of the puzzle is to relocate all disks to the third peg, at some point in the puzzle, the largest disk must be placed onto the empty third peg. This can only be done when the largest disk is the only disk on the first peg and the remaining $(n-1)$ disks are on the second peg. These remaining disks form a stack. Now before we can have a stack of $(n-1)$ disks, we must have a stack of $(n-2)$ disks so that the second largest disk can be relocated, and so on. So, looking at the puzzle from the forward direction, we must first form a stack of one, then a stack of two, etc. until a stack of $(n-1)$. We refer to this as the ***midpoint***. After this, we move the largest disk onto the third peg. To complete the puzzle, we then reverse the process. We will look at the puzzle from the viewpoint of building these stacks of disks, and we will look at some patterns concerning how these stacks are formed.

It should be noted that in order to minimize the number of moves to complete the puzzle, we cannot simply minimize the number of moves that it takes to form each stack, and then add those together. We find that in minimizing the number of moves to form some stacks, we greatly increase the number of moves that it takes to form the next stack. However, there is a way to build successive stacks such that the number of moves of the two together is kept to a minimum. We see that the value of k will determine how we need to build these stacks.

We have written a computer program to give us possible moves from one stack to the next, with the minimum paths to each configuration stored (since we do not know which configuration we want to use in the overall minimum solution). Then for each of these minimum paths, the program repeats the

process, stopping at every configuration of the next stack. Once the midpoint of the puzzle has been reached, the program takes the overall minimum path (which is the concatenation of the paths from the first stack to the second stack, and then from the second stack to the third stack, etc.) and uses that as the minimum solution. Thus, what we are doing is minimizing the number of moves that it takes to build each stack, such that the disks are in some desired configuration in each stack.

When completing any k-Tower of Hanoi puzzle, one always makes the first move based on whether n is odd or even. If n is odd, then one will move the first disk to the third peg. If n is even, then one will move the first disk to the second peg. Because of this, and because our proofs involve an arbitrary n, when we speak of moving disks, we will use the terms ***source*** peg, ***intermediate*** peg, and ***destination peg***. For odd values of n, the intermediate peg will be the second peg, and the destination peg will be the third peg. For even values of n, the intermediate peg will be the third peg, and the destination peg will be second peg. For both cases, the source peg will be the first peg. Thus, when we build stacks of disks containing an even number of disks, they will always be on the intermediate peg, and stacks of disks containing an odd number of disks will always be on the destination peg. So, whenever we are making moves that depend on whether n is even or odd, we will use this terminology. Otherwise, if the peg is known, we will use the traditional terms.

With some of these proofs, when forming stacks of disks, we can take advantage of the special rule involving k. For this, we will use the concept of ***flipping*** a group of disks. A ***flip*** of a group of disks is the process of moving disks, which are at the top of a stack, in succession to some other peg so that their new arrangement is the inverted arrangement that they started in. Note that a flip of x disks takes x moves. Also, the peg that the disks are being moved to must contain at least k disks that are larger than each disk being flipped (so as to not violate the rule involving k).

Also, the moves that we can make that take advantage of the rule involving k will be highlighted in the proofs. Such moves will be considered ***saves*** as compared to moves made while completing the classic Tower of Hanoi puzzle, as these are the moves that allow the puzzle to be completed more quickly. Whenever we make moves that follow the traditional Tower of Hanoi puzzle, the case where $k = (n - 0)$, we say that we follow the ***classic rules***. We often leave these moves to the reader for the sake of simplicity. Additionally, because our proofs involve many specific moves, we have included figures for clarity. The figures use arrows connecting distinct steps of the puzzles and these arrows denote either a single move or a group of moves. For groups of moves, we will make them using the classic rules, using a flip, or by reversing moves that were previously shown. The type of move will be listed beneath the arrow. For the case of a flip, which always results in some number of moves being saved (as compared to how the classic Tower of Hanoi puzzle is completed), we typically indicate the number of saves by a negative number, which will be listed above the arrow. Color-coded disks are used to help identify the disks being moved.

Rules for the k-Tower of Hanoi Puzzle

- Disks can only be moved one at a time.
- Only the top disk on a stack can be moved from that peg.
- A disk can be placed on top of a smaller disk, provided that the bottom k disks of that stack are the largest in the stack (where the largest disk is at the bottom, the next largest on top of it, etc., up to the kth largest disk).

Upper bound on the minimum number of moves to complete the k-Tower of Hanoi Puzzle whenever $k = (n - x)$

Here we consider the k-Tower of Hanoi puzzle where k depends on the number of disks, n. We show an upper bound $T_k(n)$ for the cases of $k = (n - x)$ for $x = 0, 1, 2, 3, 4, 5$.

$$T_{n-x}(n) = \begin{cases} 2^n - 1 & \text{if } x = 0, 1, 2 \\ 2^n - 3 & \text{if } x = 3 \\ 2^n - 4n + 3 & \text{if } x = 4, n \geqslant 7 \\ 2^n - 20n + 57 & \text{if } x = 5, n \geqslant 9 \end{cases}$$

Upper bound on the minimum number of moves to complete the k-Tower of Hanoi Puzzle whenever $k = (n - 0)$

Since $k = (n - 0) = n$, this puzzle is the classic Tower of Hanoi puzzle (since no disk can be placed on top of a smaller one), so its upper bound is $2^n - 1$.

Upper bound on the minimum number of moves to complete the k-Tower of Hanoi Puzzle whenever $k = (n - 1)$

This puzzle requires the same number of moves as the $k = (n-0)$ puzzle because a disk can be placed on top of a smaller disk provided the bottom $k = (n - 1)$ disks are the largest in that stack. If this condition is true, then there is only one disk that is not in the stack of $(n - 1)$ disks. Thus, there does not exist another disk that can be placed on top of some smaller disk.

Upper bound on the minimum number of moves to complete the k-Tower of Hanoi Puzzle whenever $k = (n - 2)$

Once again, this puzzle requires the same number of moves as the $k = (n - 0)$ puzzle because the only time we can place disks out of order is whenever we have a stack of $(n - 2)$ disks with the two remaining disks being smaller than each of the $(n - 2)$ disks. Obviously the only configurations that allow this are the beginning and ending configurations, but these configurations are fixed.

Upper bound on the minimum number of moves to complete the k-Tower of Hanoi Puzzle whenever $k = (n-3)$

This puzzle has a bound of $2^n - 3$. Notice that this is 2 fewer moves than using the classic rules. So, there are two places in the $k = (n-3)$ case where a move is saved as compared to the $k = (n-2)$ case. The saves occur when we are building a stack of $(n-1)$ disks. We complete this puzzle differently based upon whether n is even or odd.

- If n is even, we make moves using the classic rules until we reach the configuration where the first peg contains the nth and $(n-1)^{\text{st}}$ largest, and the largest disks; the third peg contains no disks; and the second peg contains the remainder of the disks.
- If n is odd, we make moves using the classic rules until we reach the configuration where the first peg contains largest disk; the third peg contains the nth and $(n-1)^{\text{st}}$ largest disks; and the second peg contains the remainder of the disks.

Next, we flip the nth and $(n-1)^{\text{st}}$ largest disks onto the second peg, which saves 1 move, getting us to the midpoint. Then we move the largest disk onto the third peg. Next, we reverse the steps prior to moving the largest disk, which saves 1 more move and completes the puzzle. See Figure 1.

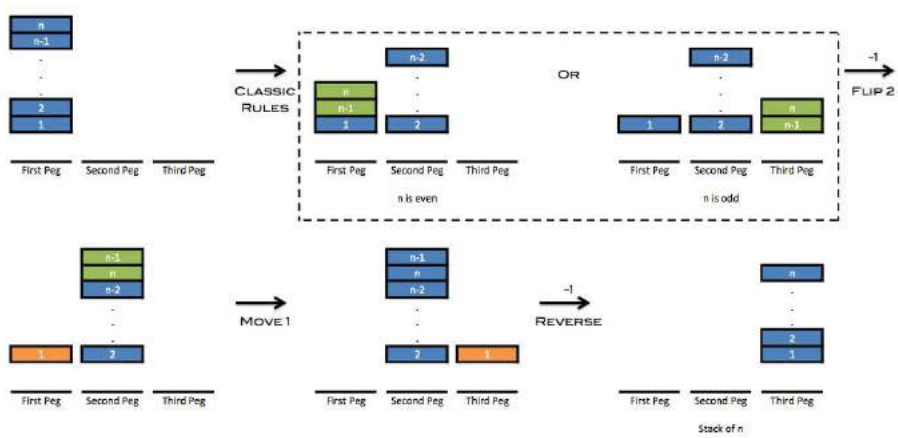

Figure 1: Upper Bound for $k = (n-3)$. Whenever $n = 4$, the disk labelled $n-2$ coincides with the disk labelled 2.

Thus, it takes

$$(2^n - 1) - 1 - 1 = 2^n - 3$$

moves to complete the puzzle.

Upper bound on the minimum number of moves to complete the k-Tower of Hanoi Puzzle whenever $k = (n-4)$

To obtain this bound, we will divide it into the following steps.

- Building a stack of 3
- Building stacks of $4, 5, \ldots, (n-3)$
- Building a stack of $(n-2)$
- Building a stack of $(n-1)$
- Building a stack of n

Note that this bound assumes a starting number of disks $n \geqslant 7$.

From the starting configuration, we build a stack of 3 disks using the classic rules. Thus, it takes $2^3 - 1 = 7$ moves. See Figure 2.

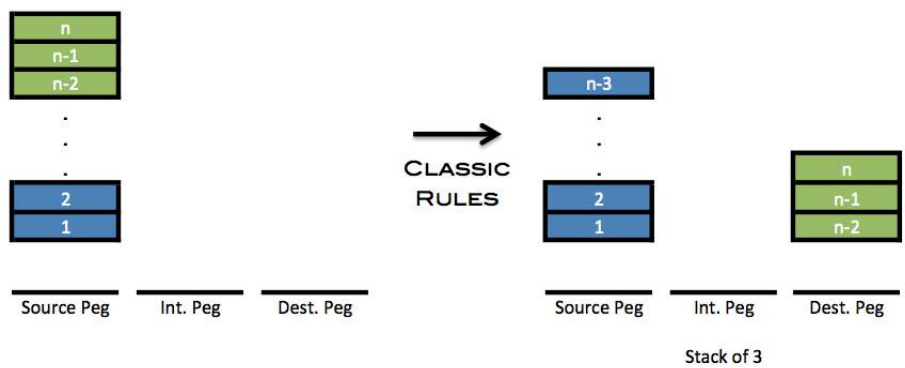

Figure 2: Building a Stack of 3 Disks

Next, we build stacks of 4, 5, …, $(n-3)$ using the classic rules until we reach a configuration where we can take advantage of the rule involving k. This configuration occurs, for each of these stacks, right before we are to place the 2nd largest disk of the stack we are building onto the largest disk of the stack we are building. Building a stack of 4 disks illustrates this concept immediately. We first move the $(n-3)^{\text{rd}}$ largest disk (this is the largest disk in the stack of 4 that we are building) from the source peg onto the intermediate peg. Notice that because $k = (n-4)$ and because we have $(n-4)$ disks on the source peg, we can flip 2 disks from the destination peg to the source peg, which saves 1 move. Next, we move the $(n-2)^{\text{nd}}$ largest disk (this is the 2nd largest disk in the stack of 4 that we are building) from the destination peg onto the intermediate peg. We then flip 2 disks from the source peg onto the intermediate peg, which saves 1 more move. We have now completed the stack of 4 disks and we have saved 2 moves. See Figure 3.

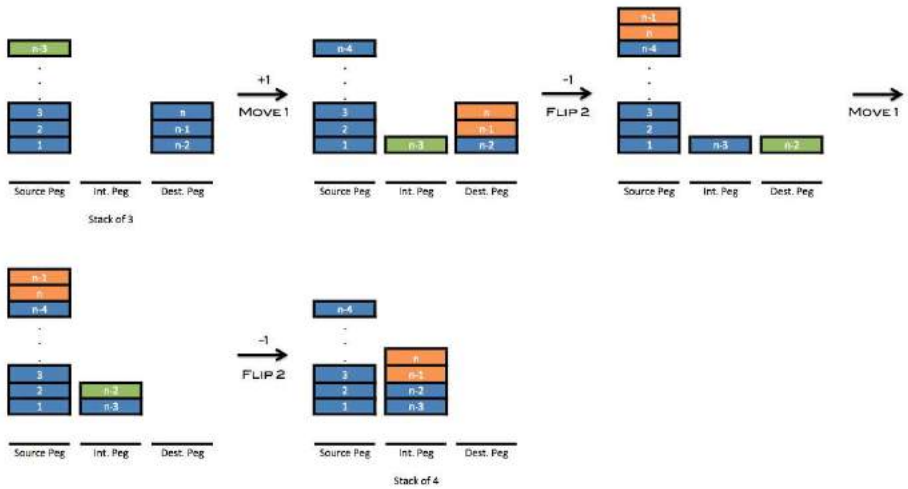

Figure 3: Building a Stack of 4 Disks. Whenever $n = 7$, the disk labelled $n - 4$ coincides with the disk labelled 3.

For the next stacks of $5, 6, \ldots, (n-3)$ disks we repeat this process.

- For a stack of an even number of disks, we make moves using the classic rules until we reach a configuration where the destination peg contains the nth and $(n-1)^{\text{st}}$ largest disks and the 2nd largest disk of the stack we are building; the intermediate peg contains only the largest disk of the stack we are building; and the source peg contains the remainder of the disks.

- For a stack of an odd number of disks, we make moves using the classic rules until we reach a configuration where the destination peg contains the nth and $(n-1)^{\text{st}}$ largest disks and the largest disk of the stack we are building; the intermediate peg contains only the 2nd largest disk of the stack we are building; and the source peg contains the remainder of the disks.

Next, we flip the nth and $(n-1)^{\text{st}}$ largest disks onto the source peg, which saves 1 move. Then we move the 2nd largest disk of the stack we are building onto the largest disk of the stack we are building. Next, we flip the nth and $(n-1)^{\text{st}}$ largest disks from the source peg onto the intermediate peg, which saves 1 more move. We then use the classic rules to finish building the stack. So, whenever we are building these stacks, we save 2 moves from the two times that we flip disks and the other moves are done according to the classic rules. See Figure 4.

Thus, it takes

$$(1+2^3-1-2)+(1+2^4-1-2)+\cdots+(1+2^{n-4}-1-2) = \sum_{i=3}^{n-4}(2^i-2) = \frac{1}{8}(32+2^n-16n)$$

moves to build stacks of 4, 5,..., $(n-3)$ disks.

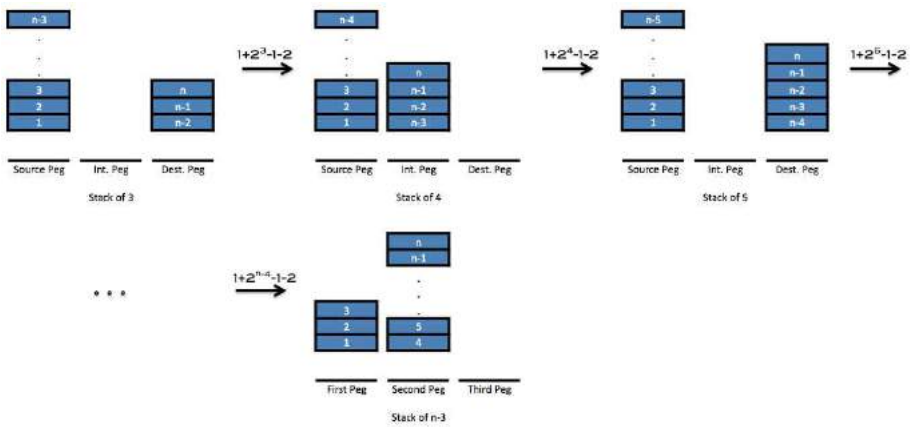

Figure 4: Overview of Building Stacks of 4, 5, $\cdots$, $(n-3)$ Disks. Notice that the pegs for the final configuration are fixed and are labelled accordingly.

Next, we wish to build a stack of $(n-2)$ disks.

- If n is even, we begin by making moves using the classic rules until we reach a configuration where the second peg contains the nth and $(n-1)^{\text{st}}$ largest disks on top of the 4th largest disk; the third peg contains only the 3rd largest disk; and the first peg contains the remainder of the disks.
- If n is odd, we begin by making moves using the classic rules until we reach a configuration where the second peg contains only the 4th largest disk; the third peg contains the nth and $(n-1)^{\text{st}}$ largest disks on top of the 3rd largest disk; and the first peg contains the remainder of the disks.

Then we flip the nth and $(n-1)^{\text{st}}$ largest disks onto the first peg, which saves 1 move. Next, we move the 4th largest disk onto the third peg.

- If n is even, we flip the nth and $(n-1)^{\text{st}}$ largest disks onto the third peg.
- If n is odd, we flip the nth and $(n-1)^{\text{st}}$ largest disks onto the second peg.

This flip saves 1 more move.

- If n is even, we make moves using the classic rules until we reach a configuration where the first peg contains the nth and $(n-1)^{\text{st}}$ largest, the 2nd largest, and the largest disks; the second peg contains no disks; and the third peg contains the remainder of the disks.
- If n is odd, we make moves using the classic rules until we reach a configuration where the first peg contains the 2nd largest and the largest disks; the second peg contains the nth and $(n-1)^{\text{st}}$ largest disks; and the third peg contains the remainder of the disks.

Then we flip the nth and $(n-1)^{\text{st}}$ largest disks onto the third peg. This flip saves 1 more move and completes the stack of $(n-2)$. See Figure 5.

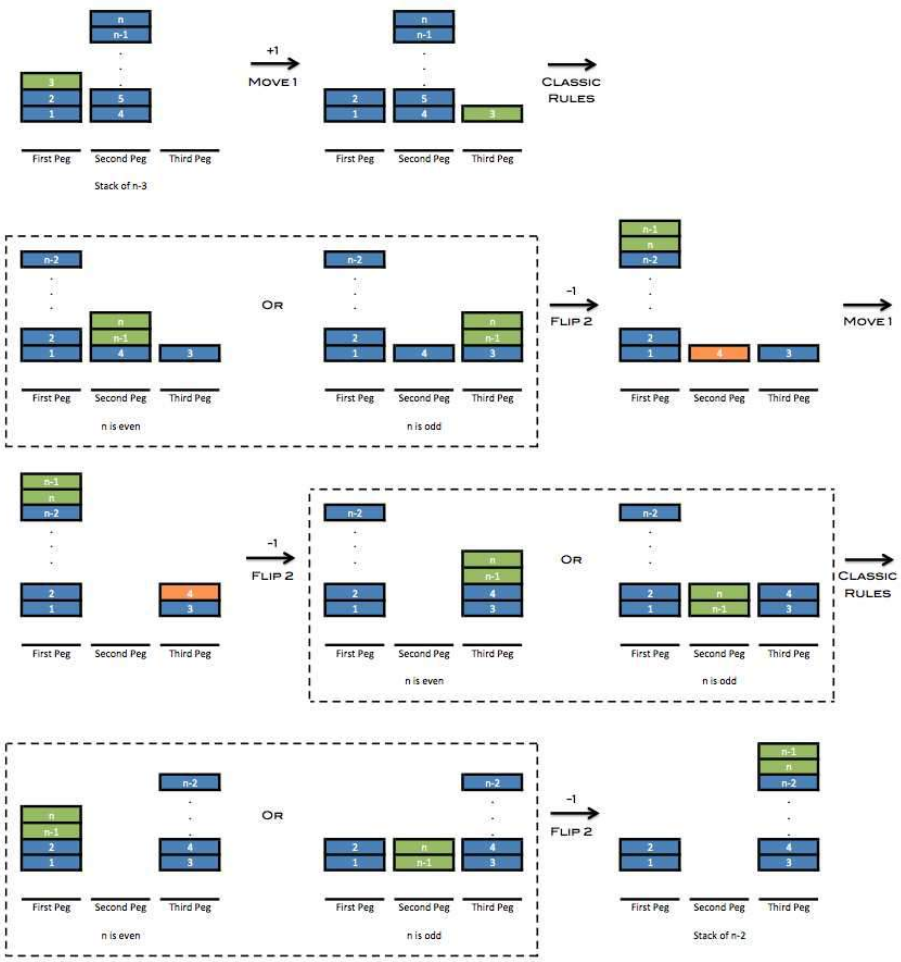

Figure 5: Overview of Building a Stack of $(n-2)$ Disks

Thus, we save 3 moves and it takes

$$1 + (2^{n-3} - 1) - 1 - 1 - 1 = 2^{n-3} - 3$$

moves to build this stack.

We now form a stack of $(n-1)$ disks. We start by moving the 2nd largest disk from the first peg onto the second peg.

- If n is even, we flip the nth and $(n-1)^{\text{st}}$ largest disks onto the second peg.
- If n is odd, we flip the nth and $(n-1)^{\text{st}}$ largest disks onto the first peg.

This flip saves 1 move.

- If n is even, we follow the classic rules until we reach a configuration where the second peg contains the nth and $(n-1)^{\text{st}}$ largest, and the 2nd largest disk; the third peg contains the 3rd largest disk; and the first peg contains the remaining disks.
- If n is odd, we follow the classic rules until we reach a configuration where the second peg contains only the 2nd largest disk; the third peg contains the nth and $(n-1)^{\text{st}}$ largest, and the 3rd largest disk; and the first peg contains the remaining disks.

We then flip the nth and $(n-1)^{\text{st}}$ largest disks onto the first peg. This saves 1 more move. Next, we move the 3rd largest disk onto the second peg.

- If n is even, we flip the nth and $(n-1)^{\text{st}}$ largest disks onto the third peg.
- If n is odd, we flip the nth and $(n-1)^{\text{st}}$ largest disks onto the second peg.

This saves us 1 more move.

- If n is even, we follow the classic rules until we reach the configuration where the first peg contains the $n^{\text{th}}, (n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks and the largest disk; the third peg contains no disks; and the second peg contains the remaining disks.
- If n is odd, we follow the classic rules until we reach the configuration where the first peg contains only the largest disk; the third peg contains the $n^{\text{th}}, (n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks; and the second peg contains the remaining disks.

We flip the $n^{\text{th}}, (n-1)^{\text{st}}$, and $(n-2)^{\text{nd}}$ largest disks onto the second peg. This saves 4 more moves and completes the stack of $(n-1)$. See Figure 6.

We have now reached the midpoint. So far we have made

$$\left(7 + \frac{1}{8}\left(32 + 2^n - 16n\right) + \left(2^{n-3} - 3\right) + \left(2^{n-2} - 7\right)\right) = \left(\frac{2^n}{2} - 2n + 1\right)$$

moves.

Now, we wish to form a stack of n. We do this by moving the largest disk onto the third peg and then reversing the previous steps for building stacks of $1, 2, \ldots, (n-1)$. See Figure 7.

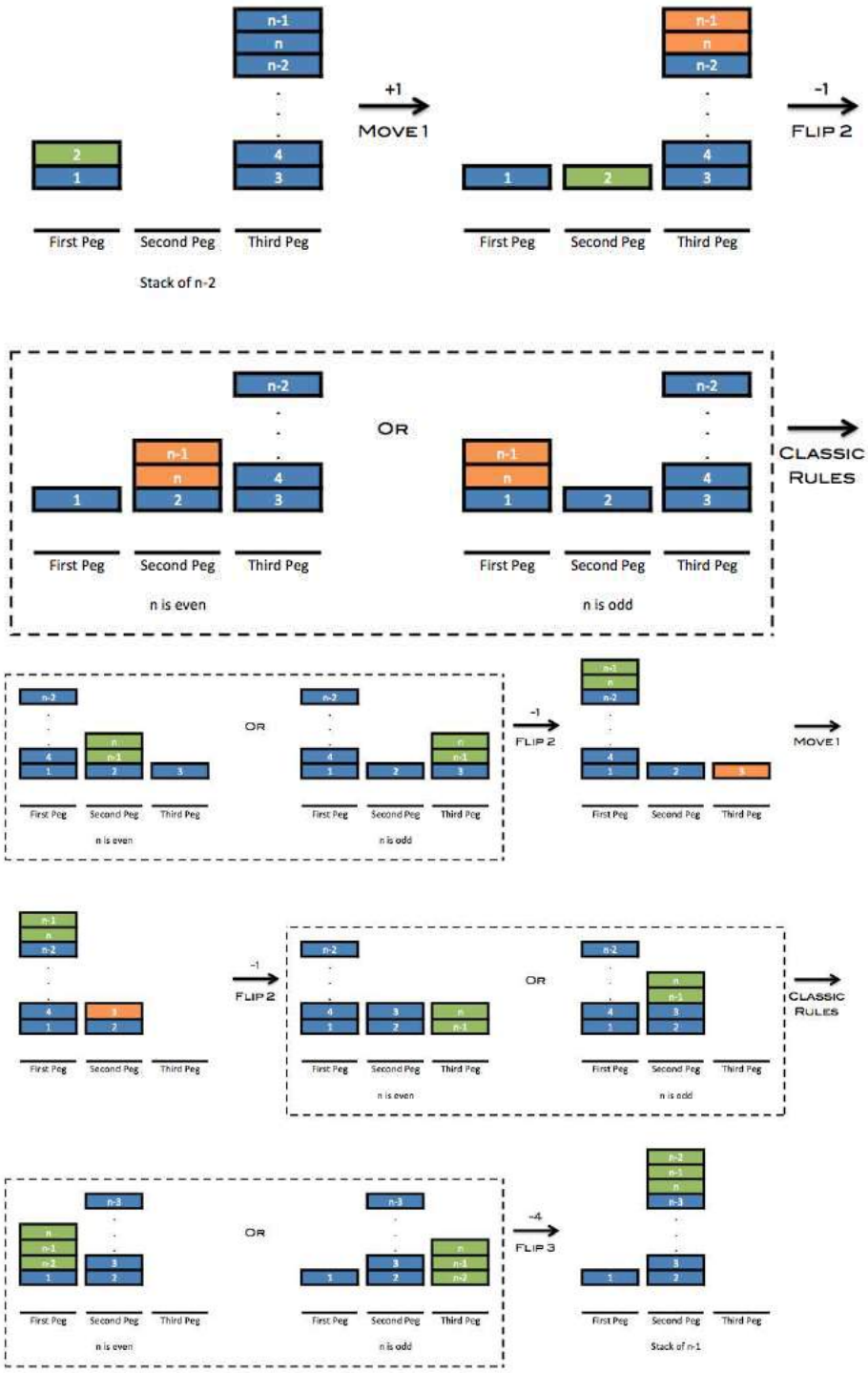

Figure 6: Overview of Building a Stack of $(n-1)$ Disks

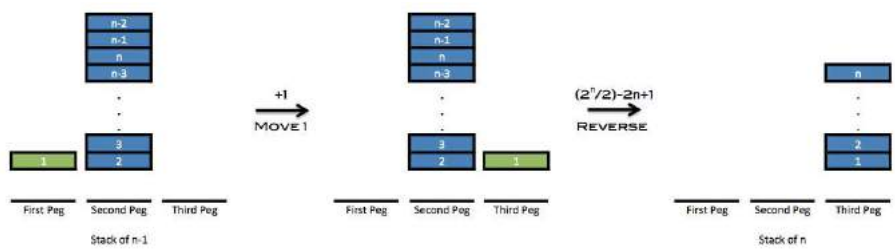

Figure 7: Overview of Building a Stack of n Disks

Thus, to build this stack it takes

$$1 + \left(\frac{2^n}{2} - 2n + 1\right)$$

moves.

Therefore, to complete the puzzle, it takes

$$\left(\frac{2^n}{2} - 2n + 1\right) + 1 + \left(\frac{2^n}{2} - 2n + 1\right) = 2^n - 4n + 3$$

moves.

Upper bound on the minimum number of moves to complete the k-Tower of Hanoi Puzzle whenever $k = (n - 5)$

To obtain this bound, we will divide it up into the following steps.

- Building a stack of 3
- Building a stack of 4
- Building stacks of $5, 6, \ldots, (n - 4)$
- Building a stack of $(n - 3)$
- Building a stack of $(n - 2)$
- Building a stack of $(n - 1)$
- Building a stack of n

Note that this bound assumes a starting number of disks $n \geqslant 9$.

From the starting configuration, we build a stack of 3 disks using the classic rules. Thus, it takes $2^3 - 1 = 7$ moves. See Figure 8.

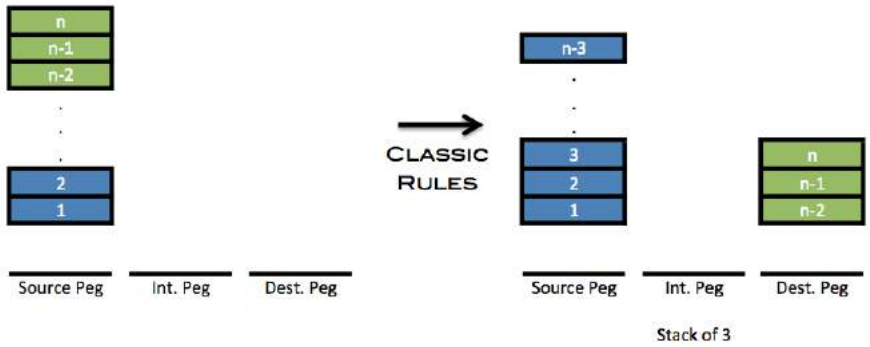

Figure 8: Building a Stack of 3 Disks

Next, we build a stack of 4 in the same manner that we built a stack of 4 in the bound for $k = (n-4)$. This takes 6 moves, as before. See Figure 9.

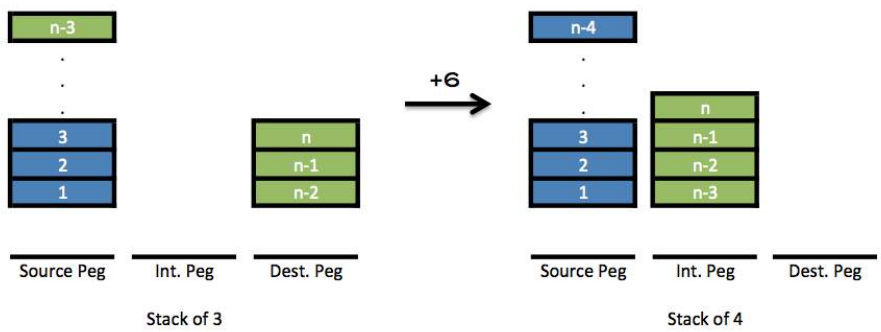

Figure 9: Building a Stack of 4 Disks

We now wish to build stacks of $5, 6, \ldots, (n-4)$. These stacks are built in a similar manner that stacks of $4, 5, \ldots, (n-3)$ for $k = (n-4)$ were built. Since $k = (n-5)$, we now only need a stack of $(n-5)$ in order to take advantage of the special rule involving k. As a result, we can flip 3 disks onto stacks of $(n-5)$ instead of flipping 2 disks onto stacks of $(n-4)$ (as we did whenever $k = (n-4)$). For a flip of 3 disks, we save 4 moves each time. So, we build stacks of 4, 5, …, $(n-3)$ by using the classic rules until we reach a configuration where we can take advantage of the rule involving k. This configuration occurs, for each of these stacks, right before we are to place the 2nd largest disk of the stack we are building onto the largest disk of the stack we are building. Building a stack of 5 illustrates this concept immediately. We first relocate the $(n-4)^{\text{th}}$ largest disk (this is the largest disk in the stack of 5 that we are building) from the source peg onto the destination peg. Notice that because $k = (n-5)$ and because we have $(n-5)$ disks on the source peg, we can flip 3 disks from the intermediate peg to the source peg, which saves 4 moves. Next, we move the $(n-3)^{\text{rd}}$ largest disk (this is the 2nd largest disk in the stack of 5 that we are

building) from the intermediate peg onto the destination peg. We then flip 3 disks from the source peg onto the destination peg, which saves 4 more moves. We have now completed the stack of 5 disks and we have saved 8 moves. See Figure 10.

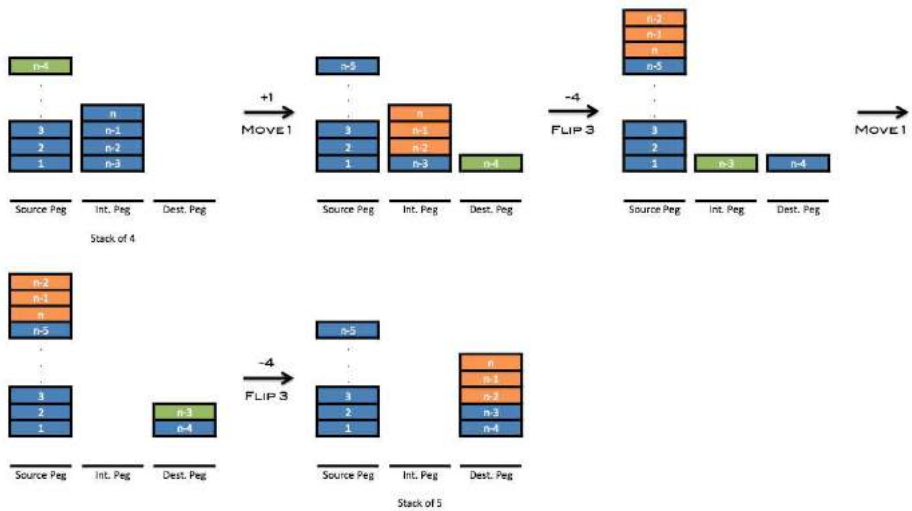

Figure 10: Building a Stack of 5 Disks

For the next stacks of $6, 7, \ldots, (n-4)$ disks we repeat this process. For each of these stacks, we begin by making moves using the classic rules until we reach a configuration where the source peg contains every disk except: the largest disk of the stack we are building, the 2nd largest disk of the stack we are building, and the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks. Moreover, the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks must be on a peg with only the largest disk of the stack we are building or the 2nd largest disk of the stack we are building. Note that where these disks are depends on whether the stack we are building contains an odd or an even number of disks. For example, when building a stack of 5 disks, the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks will be on the 2nd largest disk of the stack we were building. When building a stack of 6, they will be on largest disk of the stack we are building. Now, because we have $(n-5)$ disks on the source peg, we can take advantage of the special rule involving k. So, flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks onto the source peg, which saves 4 moves. Move the 2nd largest disk of the stack we are building onto the largest disk of the stack we are building, and then flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks from the source peg onto one of the other pegs (if we are building a stack of an even number of disks, we flip the disks onto the empty peg, otherwise we flip the disks onto the peg containing disks), which saves 4 more moves. We then use the classic rules to finish building the stack. So, whenever we are building these stacks, we save 4 moves from the two times that we flip disks and the other moves are done according to the classic rules. See Figure 11.

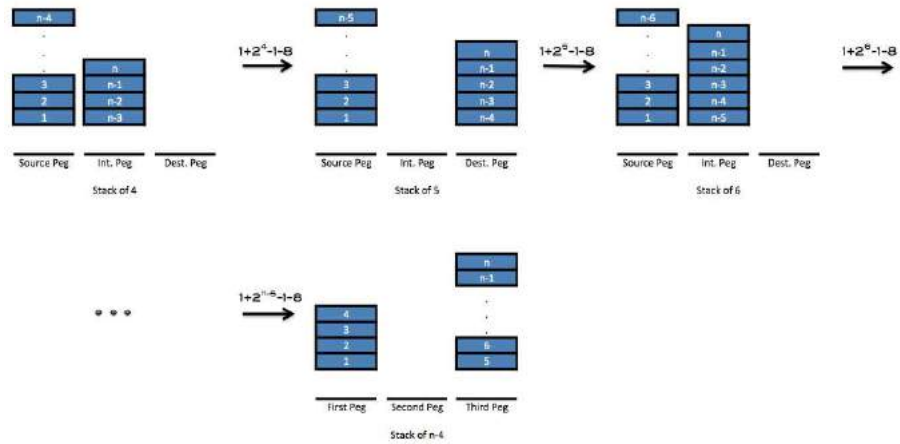

Figure 11: Overview of Building Stacks of $5, 6, \cdots, (n-4)$ Disks. Whenever $n = 9$, the disk labelled $n-6$ coincides with the disk labelled 3.

So, to build all of these stacks, it takes

$$(1 + 2^4 - 1 - 8) + \cdots + (1 + 2^{n-5} - 1 - 8) = \sum_{i=4}^{n-5} (2^i - 8) = \frac{(2^n - 128n + 768)}{16}$$

moves.

Now we want to build a stack of $(n-3)$ disks. We start by moving the 4th largest disk onto the second peg.

- If n is even, we follow the classic rules until we reach the configuration where the second peg contains only the 4th largest disk; the third peg contains the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks and the 5th largest disk; and the first peg contains the remainder of the disks.

- If n is odd, we follow the classic rules until we reach the configuration where the second peg contains the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks and 4th largest disk; the third peg contains only the 5th largest disk; and the first peg contains the remainder of the disks.

We then flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks onto the first peg, which saves 4 moves. Next, we move the 5th largest disk onto the second peg.

- If n is even, we flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks from the first peg onto the second peg.

- If n is odd, we flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks from the first peg onto the third peg.

This flip saves 4 more moves.

- If n is even, we make moves using the classic rules until we reach the configuration where the first peg contains the largest, 2nd largest and 3rd largest disks; the third peg contains the nth and $(n-1)^{\text{st}}$ largest disks; and the second peg contains the remainder of the disks.

- If n is odd, we make moves using the classic rules until we reach the configuration where the first peg contains the nth and $(n-1)^{\text{st}}$ largest, 3rd largest, 2nd largest and largest disks; the third peg contains no disks; and the second peg contains the remainder of the disks.

Then we flip the nth and $(n-1)^{\text{st}}$ largest disks onto the second peg, which saves 1 more move.See Figure 12.

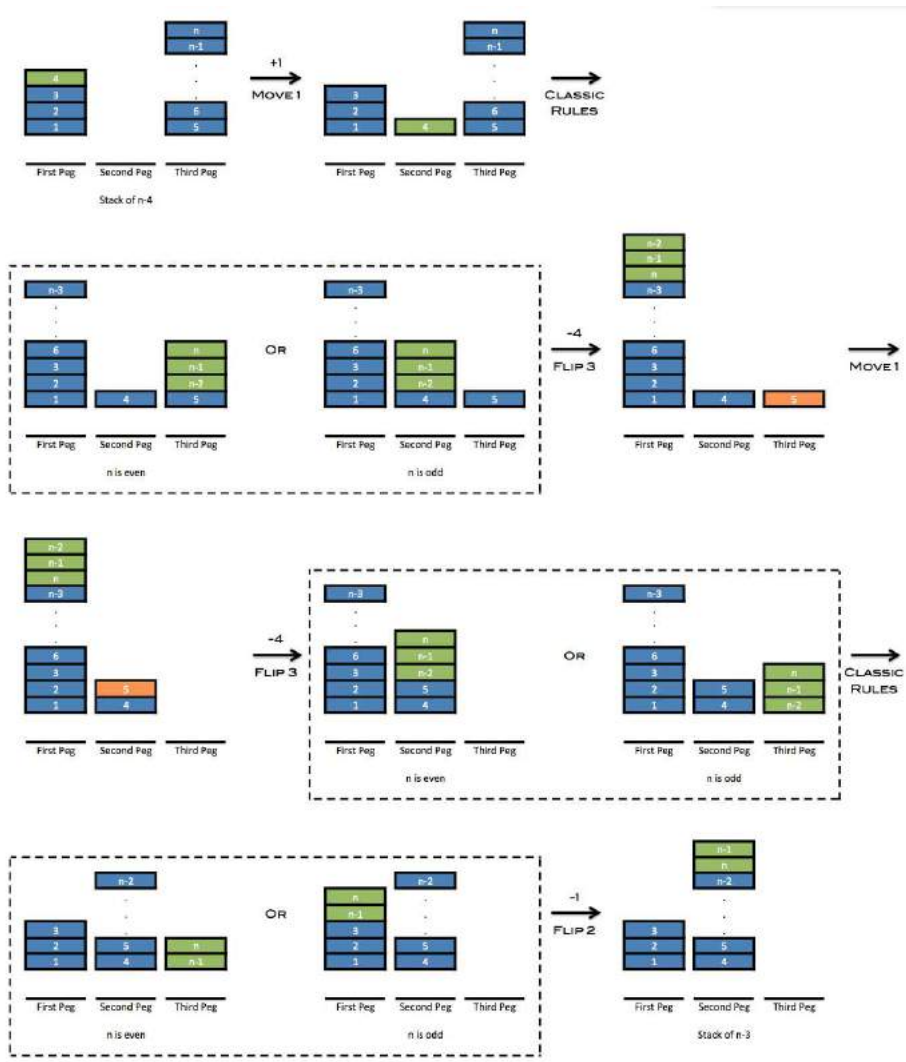

Figure 12: Overview of Building a Stack of $(n-3)$ Disks. Whenever $n = 9$, the disk labelled $n-6$ coincides with the disk labelled 3.

This completes the stack of $(n-3)$ disks and it takes $1+(2^{n-4}-1)-9$ moves.

Now we want to build a stack of $(n-2)$ disks. We begin by moving the 3rd largest disk onto the third peg.

- If n is even, we flip the nth and $(n-1)^{\text{st}}$ largest disks from the second peg onto the first peg.
- If n is odd, we flip the nth and $(n-1)^{\text{st}}$ largest disks from the second peg onto the third peg.

This flip saves 1 move.

- If n is even, we follow the classic rules until we reach the configuration where the second peg contains only the 4th largest disk; the third peg contains the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks and the 3rd largest disk; and the first peg contains the remainder of the disks.
- If n is odd, we follow the classic rules until we reach the configuration where the second peg contains the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks and 4th largest disk; the third peg contains only the 3rd largest disk; and the first peg contains the remainder of the disks.

We then flip the $n^{\text{th}}, (n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks onto the first peg and save 4 more moves. Now, we move the 4th largest disk onto the third peg.

- If n is even, we flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks from the first peg onto the second peg.
- If n is odd, we flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks from the first peg onto the third peg.

This flip saves 4 more moves.

- If n is even, we follow the classic rules until we reach the configuration where the first peg contains the 2nd largest and the 1st largest disks; the second peg contains the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks; and the third peg contains the remainder of the disks.
- If n is odd, we follow the classic rules until we reach the configuration where the first peg contains the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest, the 2nd largest and the 1st largest disks; the second peg contains no disks; and the third peg contains the remainder of the disks.

Next, we flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks onto the third peg, which saves 4 more moves. See Figure 13.

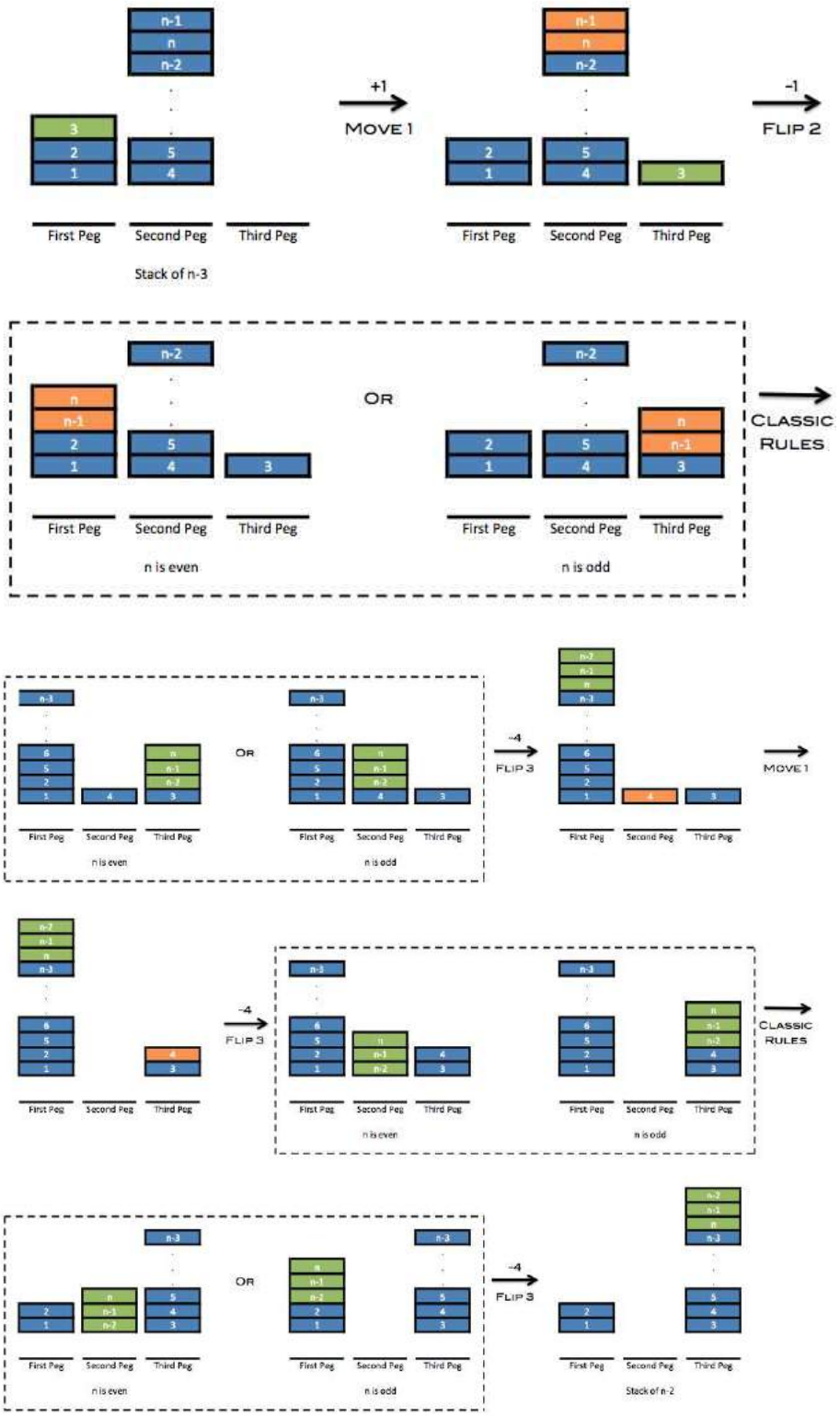

Figure 13: Overview of Building a Stack of $(n-2)$ Disks. Whenever $n = 9$, the disk labelled $n-6$ coincides with the disk labelled 3.

This completes the stack of $(n-2)$ disks and it takes

$$1 + (2^{n-3} - 1) - 1 - 4 - 4 - 4 = 2^{n-3} - 13$$

moves.

Next, to build a stack of $(n-1)$ disks, we will divide this step up into two parts. For the first part, we start by moving the 2nd largest disk onto the second peg.

- If n is even, we flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks from the third peg onto the first peg.
- If n is odd, we flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks from the third peg onto the second peg.

This flip saves 4 moves.

- If n is even, we follow the classic rules until we reach the configuration where the first peg contains only the largest disk; the third peg contains the nth and $(n-1)^{\text{st}}$ largest disks, the 4th largest disk and the 3rd largest disk; and the second peg contains the remainder of the disks.
- If n is odd, we follow the classic rules until we reach the configuration where the first peg contains the nth and $(n-1)^{\text{st}}$ largest disks and the largest disk; the third peg contains the 4th largest and the 3rd largest disks; and the second peg contains the remainder of the disks.

See Figure 14.

We flip the nth and $(n-1)^{\text{nd}}$ largest disks onto the second peg, which saves 1 more move. Next, we move the 4th largest disk from the third peg onto the first peg.

- If n is even, we flip the nth and $(n-1)^{\text{st}}$ largest disks from the second peg onto the first peg.
- If n is odd, we flip the nth and $(n-1)^{\text{st}}$ largest disks from the second peg onto the third peg.

This flip saves 1 more move.

- If n is even, we follow the classic rules until we reach the configuration where the second peg contains only the 2nd largest disk; the third peg contains the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks and the 3rd largest disk; and the first peg contains the remainder of the disks.
- If n is odd, we follow the classic rules until we reach the configuration where the second peg contains the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks and the 2nd largest disk; the third peg contains only the 3rd largest disk; and the first peg contains the remainder of the disks.

See Figure 15.

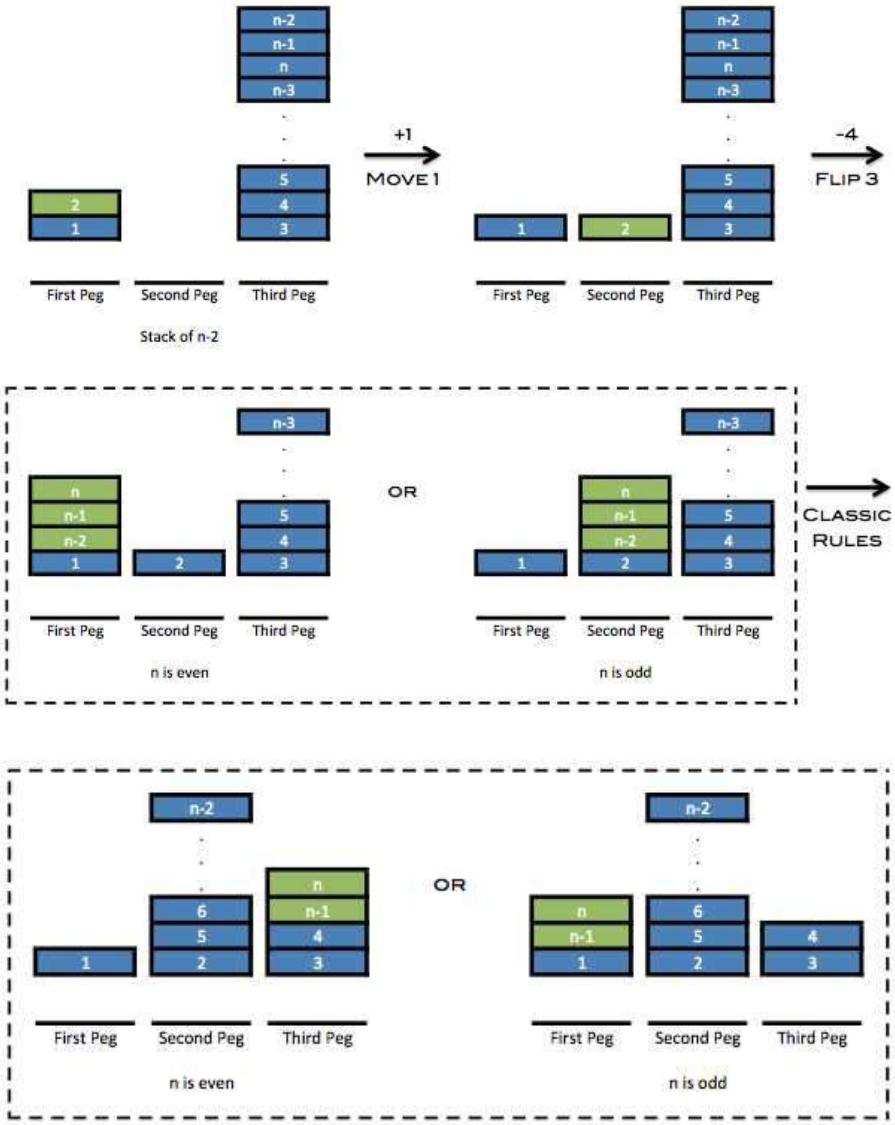
n-2
n-1
n
n-3
5
4
3
2
1
+1
MOVE 1
-4
FLIP 3
First Peg
Second Peg
Third Peg
Stack of n-2
OR
CLASSIC RULES
n is even
n is odd
6

Figure 14

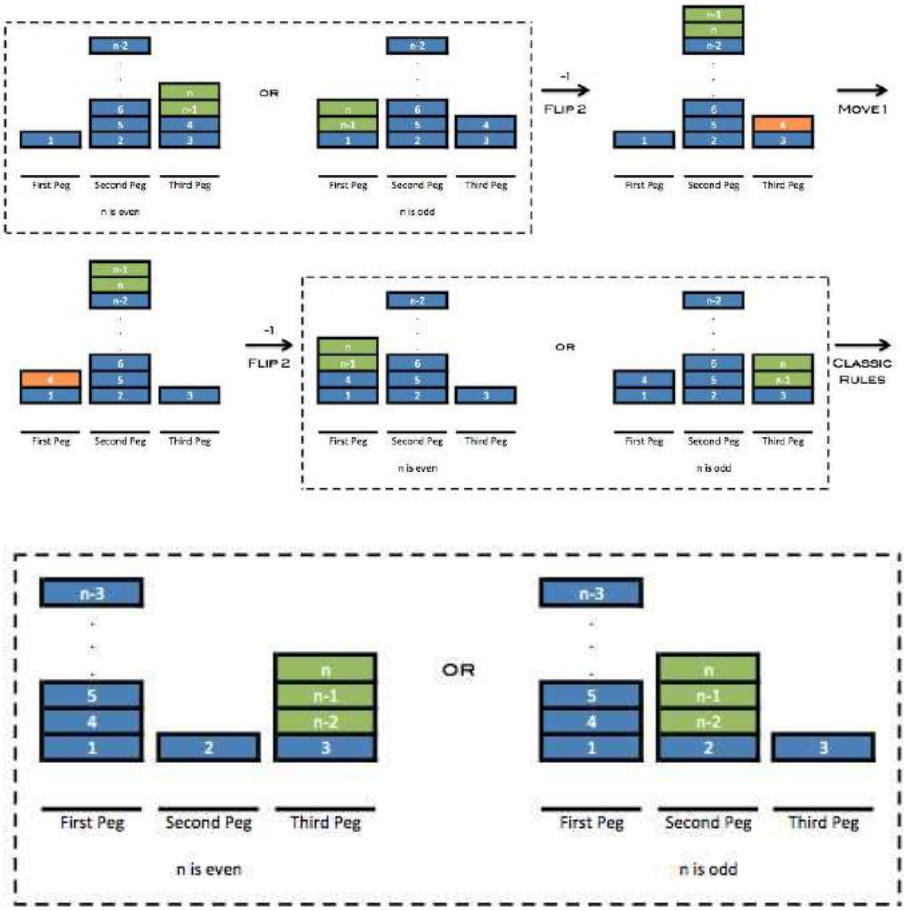

Figure 15

We then flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks onto the first peg and save 4 more moves. Then we move the 3rd largest disk from the third peg onto the second peg.

- If n is even, we flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks from the first peg onto the second peg.
- If n is odd, we flip the nth, $(n-1)^{\text{st}}$ and $(n-2)^{\text{nd}}$ largest disks from the first peg onto the third peg.

This flip saves 4 more moves. See Figure 16.

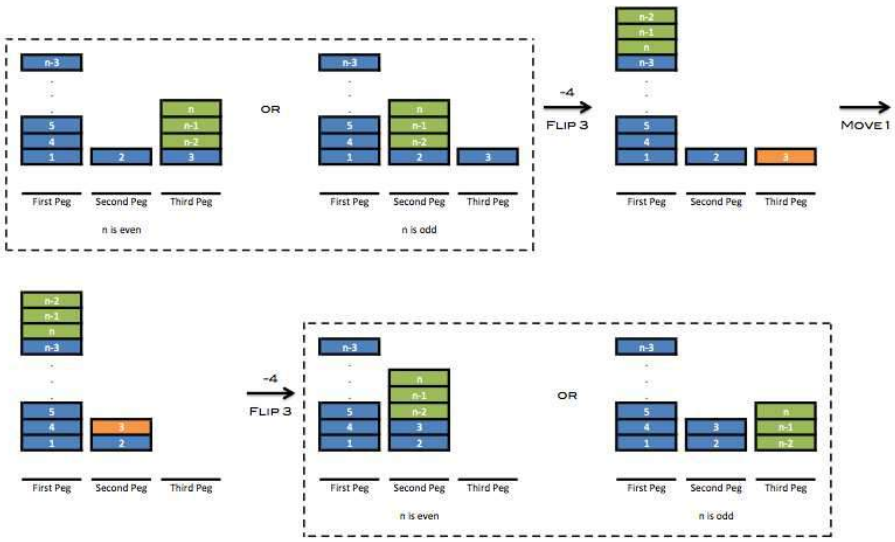

Figure 16

This concludes the first part of this step. Now we begin the second part of this step.

- If n is even, we make moves using the classic rules until we reach a configuration where the first peg contains the 5th largest, 4th largest, and largest disks; the third peg contains the nth largest and $(n-1)^{\text{st}}$ largest disks; and the second peg contains the remainder of the disks.
- If n is odd, we make moves using the classic rules until we reach a configuration where the first peg contains the nth largest, $(n-1)^{\text{st}}$ largest, 5th largest, 4th largest, and largest disks; the third peg contains no disks; and the second peg contains the remainder of the disks.

Then we flip the nth largest and $(n-1)^{\text{st}}$ largest disks onto the second peg, which saves 1 more move. We then move the 5th largest disk onto the third peg.

- If n is even, we flip the nth largest and the $(n-1)^{\text{st}}$ largest onto the first peg.
- If n is odd, we flip the nth largest and the $(n-1)^{\text{st}}$ largest onto the third peg.

This flip saves 1 more move. See Figure 17.

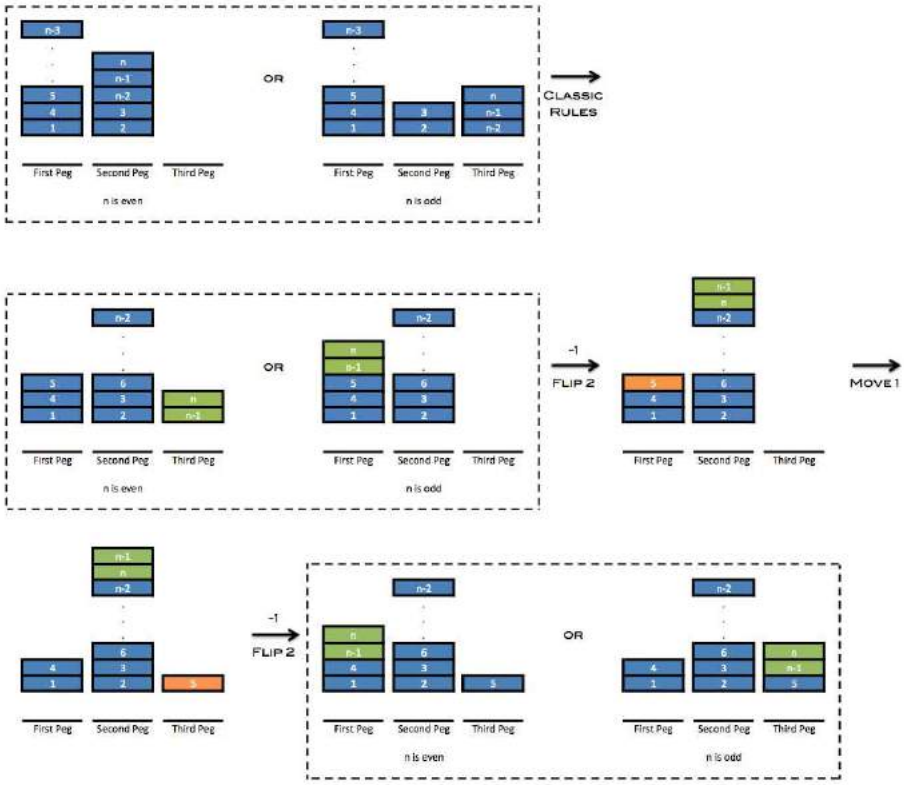

Figure 17

Next, we make a similar sequence of moves.

- If n is even, we make moves using the classic rules until we reach a configuration where the first peg contains only the largest disk; the third peg contains the nth largest, $(n-1)^{\text{st}}$ largest, 6th largest and 5th largest disks; and the second peg contains the remainder of the disks.
- If n is odd, we make moves using the classic rules until we reach a configuration where the first peg contains the nth largest, $(n-1)^{\text{st}}$ largest, and largest disks; the third peg contains the 6th largest and 5th largest disks; and the second peg contains the remainder of the disks.

Then we flip the nth largest and $(n-1)^{\text{st}}$ largest disks onto the second peg, which saves 1 more move. We then move the 6th largest disk onto the first peg.

- If n is even, we flip the nth largest and the $(n-1)^{\text{st}}$ largest onto the first peg.
- If n is odd, we flip the nth largest and the $(n-1)^{\text{st}}$ largest onto the third peg.

This flip saves 1 more move. See Figure 18.

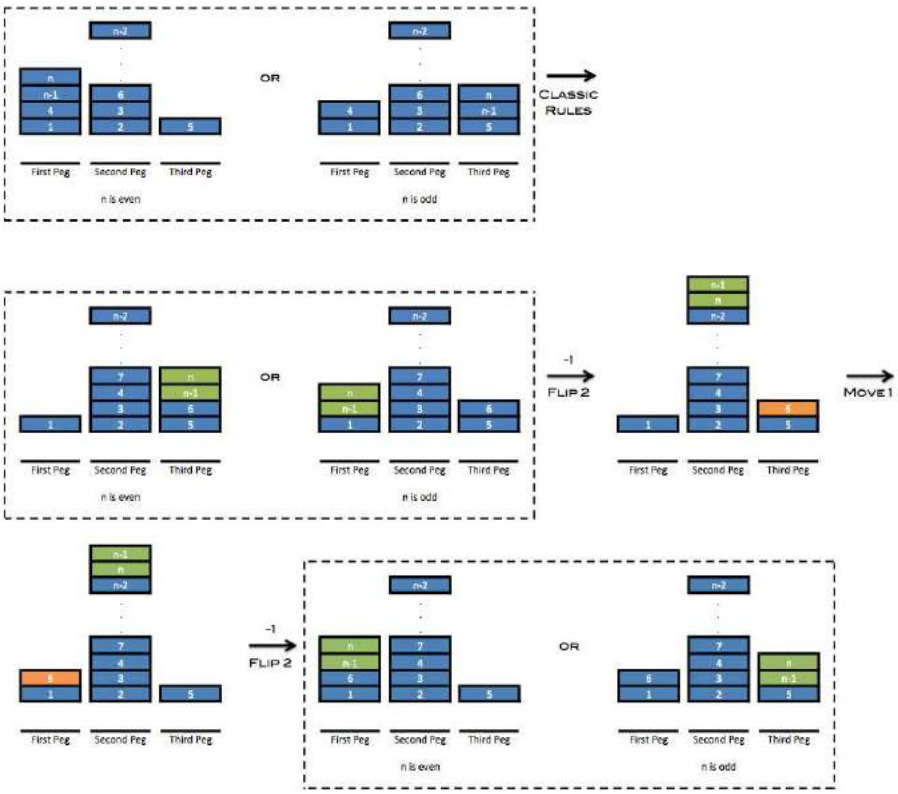

Figure 18

Notice that in Figure 17, we took advantage of the special rule involving k whenever we had the 4th largest and 5th largest disks together on a peg separate from a stack of $(n-5)$ disks. Also, in Figure 18, we took advantage of the special rule involving k whenever we had the 5th largest and 6th largest disks together on a peg separate from a stack of $(n-5)$ disks. In general, we can take advantage of the special rule involving k whenever we have a jth largest and $(j+1)^{\text{st}}$ largest disk together on a peg separate from a stack of $(n-5)$ disks, where $4 \leqslant j \leqslant (n-4)$. That is,

- If n is even and j is even, we make moves using the classic rules until we reach the configuration where the first peg contains the $(j+1)^{\text{st}}$ largest, j^{th} largest, and largest disks; the third peg contains the nth largest and the $(n-1)^{\text{st}}$ largest disks; and the second peg contains the remainder of the disks.

- If n is even and j is odd, we make moves using the classic rules until we reach the configuration where the first peg contains only the largest disk; the third peg contains the nth largest, $(n-1)^{\text{st}}$ largest, $(j+1)^{\text{st}}$ largest and j^{th} largest disks; and the second peg contains the remainder of the disks.

Then we flip the nth largest and the $(n-1)^{\text{st}}$ largest disks onto the second peg, which saves 1 more move.

- If j is even, we move the $(j+1)^{\text{st}}$ largest disk onto the third peg.
- If j is odd, we move the $(j+1)^{\text{st}}$ largest disk onto the first peg.

Next, we flip the nth largest and the $(n-1)^{\text{st}}$ largest disks onto the first peg, which saves 1 more move. See Figure 19.

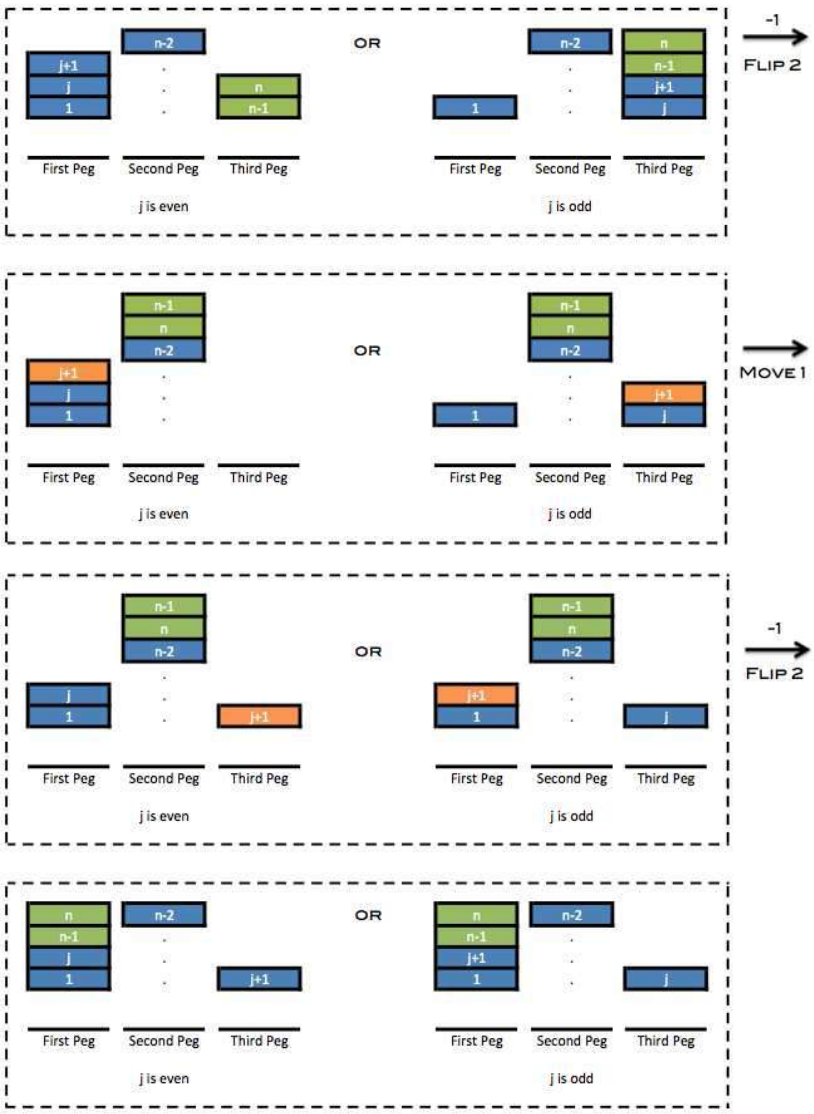

Figure 19: Moves made for $4 \leqslant j \leqslant (n-4)$ whenever n is even. The disks on the second peg are uniquely determined by the disks depicted on the other two pegs.

Otherwise,

- If n is odd and j is even, we make moves using the classic rules until we reach the configuration where the first peg contains the nth largest, $(n-1)^{\text{st}}$ largest, $(j+1)^{\text{st}}$ largest, j^{th} largest, and largest disks; the third peg contains no disks; and the second peg contains the remainder of the disks.
- If n is odd and j is odd, we make moves using the classic rules until we reach the configuration where the first peg contains the nth largest, $(n-1)^{\text{st}}$ largest, and the largest disk; the third peg contains the $(j+1)^{\text{st}}$ largest and j^{th} largest disks; and the second peg contains the remainder of the disks.

Then we flip the nth largest and the $(n-1)^{\text{st}}$ largest disks onto the second peg, which saves 1 more move.

- If j is even, we move the $(j+1)^{\text{st}}$ largest disk onto the third peg.
- If j is odd, we move the $(j+1)^{\text{st}}$ largest disk onto the first peg.

Next, we flip the nth largest and the $(n-1)^{\text{st}}$ largest disks onto the third peg, which saves 1 more move. See Figure 20.

Notice that because we use values of $j = 4, 5, 6, \ldots, (n-4)$, there are $(n-7)$ distinct values of j. Thus, we will have $(n-7)$ pairs of j and $(j+1)$ where we can take advantage of the special rule involving k. Also, for each of these pairs we save 2 moves. Thus, we will save $2(n-7)$ moves after we have made these sequences of moves, as depicted in Figure 21, we reach the following configuration:

- If n is even, we reach the configuration where the first peg contains the nth largest, $(n-1)^{\text{st}}$ largest, $(n-4)^{\text{th}}$ largest and the largest disks; the third peg contains only the $(n-3)^{\text{rd}}$ largest disk; and the second peg contains the remainder of the disks.
- If n is odd, we reach the configuration where the first peg contains the $(n-3)^{\text{rd}}$ largest and the largest disk; the third peg contains the nth, $(n-1)^{\text{st}}$, and $(n-4)^{\text{th}}$ largest disks; and the second peg contains the remainder of the disks.

Next,

- If n is even, we make moves using the classic rules until we reach the configuration where the first peg contains only the largest disk; the third peg contains the nth largest, $(n-1)^{\text{st}}$ largest, $(n-2)^{\text{nd}}$ largest and $(n-3)^{\text{rd}}$ largest disks; and the second peg contains the remainder of the disks.
- If n is odd, we make moves using the classic rules until we reach the configuration where the first peg contains the nth largest, $(n-1)^{\text{st}}$ largest, $(n-2)^{\text{nd}}$ largest and $(n-3)^{\text{rd}}$ largest disks and the largest disk; the third peg contains no disks; and the second peg contains the remainder of the disks.

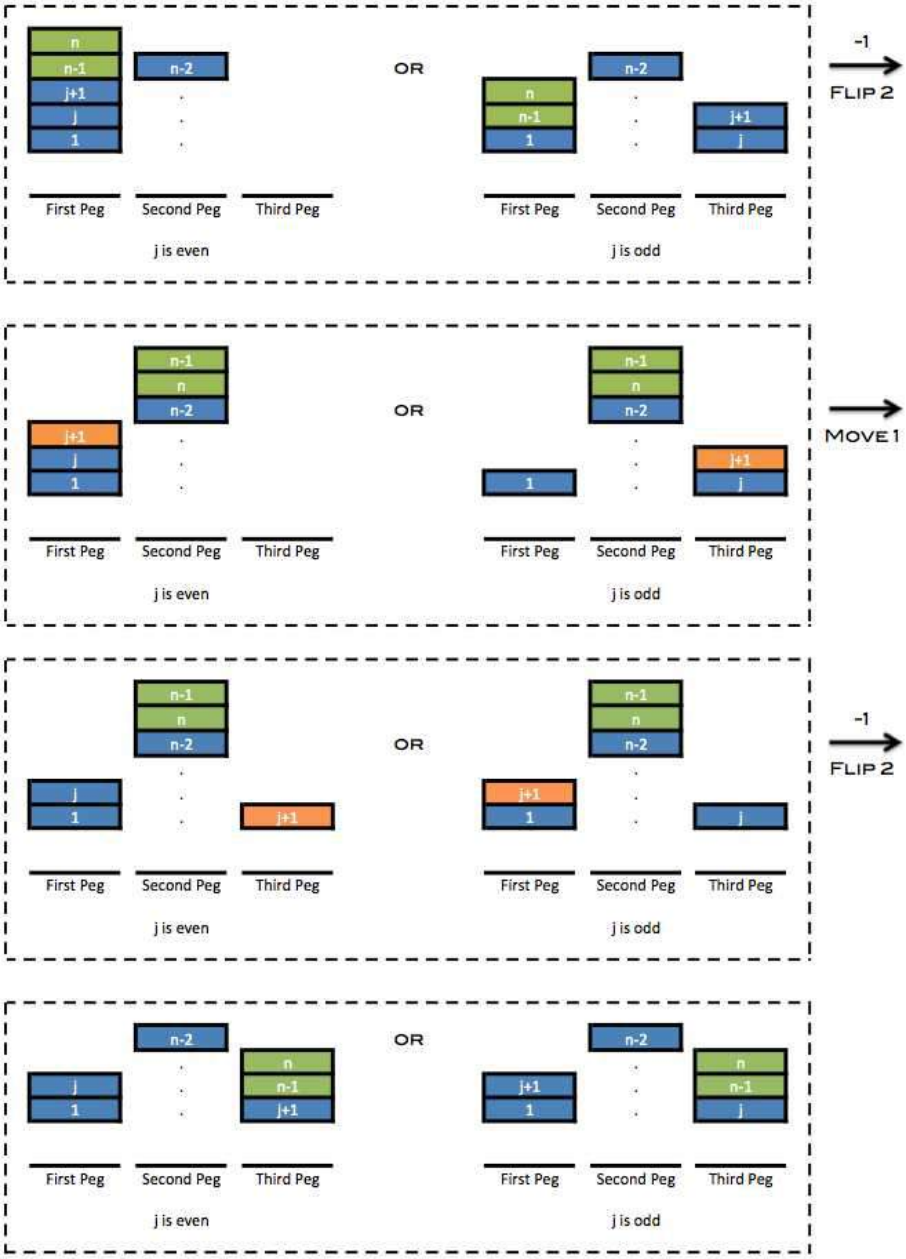

Figure 20: Moves made for $4 \leqslant j \leqslant (n-4)$ whenever n is odd. The disks on the second peg are uniquely determined by the disks depicted on the other two pegs.

Next, we flip the nth, $(n-1)^{\text{st}}$, $(n-2)^{\text{nd}}$ and $(n-3)^{\text{rd}}$ largest disks onto the second peg, which saves 11 more moves. See Figure 22.

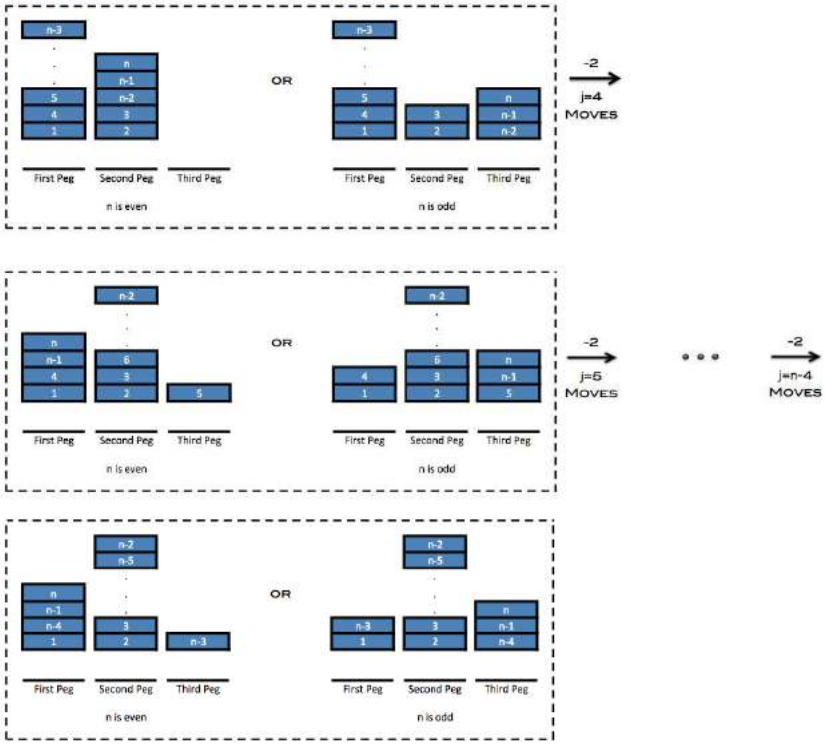

Figure 21

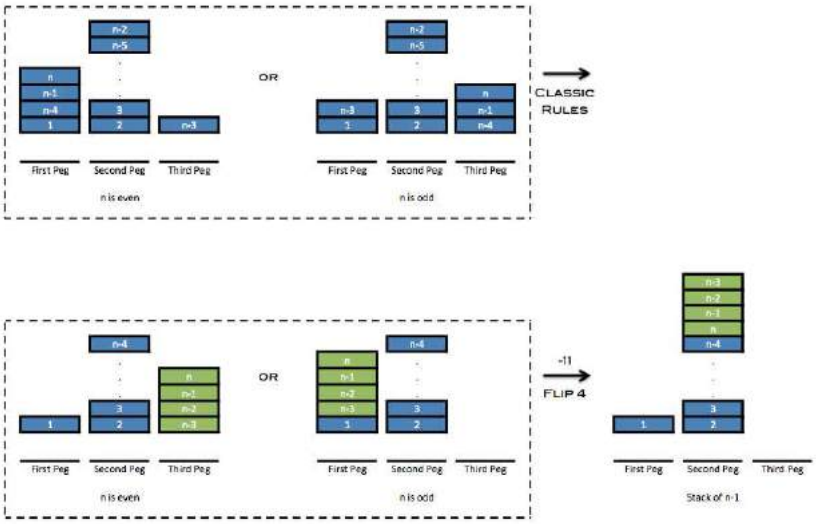

Figure 22

This completes the stack of $(n-1)$ disks and it takes

$$1 + (2^{n-2} - 1) - 4 - 1 - 1 - 4 - 4 - 2(n-7) - 11 = 2^{n-2} - 2n - 11$$

moves.

Now, we wish to form a stack of n. We do this by moving the largest disk onto the third peg and then reversing the previous steps for building stacks of $1, 2, \ldots, (n-1)$. Thus, it takes

$$1 + \left(7 + 6 + \tfrac{1}{16}(2^n - 128n + 768) + (2^{n-4} - 9) + (2^{n-3} - 13) + (2^{n-2} - 2n - 11)\right) = 1 + \left(\tfrac{2^n}{2} - 10n + 28\right)$$

moves to build this stack. See Figure 23.

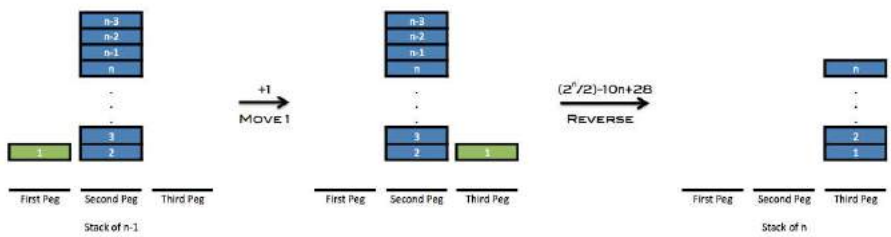

Figure 23: Overview of Building a Stack of n Disks

Therefore, it takes

$$\left(\frac{2^n}{2} - 10n + 28\right) + 1 + \left(\frac{2^n}{2} - 10n + 28\right) = 2^n - 20n + 57$$

moves to complete the puzzle.

Upper bound on the minimum number of moves to complete the k-Tower of Hanoi Puzzle whenever $k = 3$

We now look at the k-Tower of Hanoi puzzle whenever $k = 3$. We skip to $k = 3$ because upper bounds on the minimum number of moves required to complete the k-Tower of Hanoi puzzle whenever $k = 1$ and $k = 2$ are known. An upper bound of $n^2 - n + 1$ is shown in [2] whenever $k = 1$, and an upper bound of $2n^2 - 4n + 1$ whenever $k = 2$ is shown in [1].

Rules for the k-Tower of Hanoi Puzzle Whenever $k = 3$

- Disks can only be moved one at a time.
- Only the top disk on a stack can be moved from that peg.
- A disk can be moved on top of a smaller one provided that the *three* disks on the bottom of that stack are the largest on that peg (with the largest disk on the bottom, the second largest disk on top of that one, and the third largest disk on top of that one).

The upper bound, $T_3(n)$, for the minimum number of moves required to complete the k-Tower of Hanoi puzzle with n disks whenever $k = 3$ is

$$T_3(n) = \begin{cases} 2^n - 1 & \text{if } n = 3, 4, 5 \\ 61 & \text{if } n = 6 \\ 103 & \text{if } n = 7 \\ 161 & \text{if } n = 8 \\ 2n^2 + 24n - 153 & \text{if } n \geqslant 9. \end{cases}$$

Notice that for the cases of $n = 3, 4, 5, 6, 7$, we just refer back to the bounds from Section 3 for $k = n - x$ for $x = 0, 1, 2, 3, 4$ respectively, to obtain bounds of 7, 15, 31, 61 and 103 moves, respectively. We begin with the case of $n \geqslant 9$.

Theorem 1. *The optimal solution for the k-Tower of Hanoi puzzle for $n \geqslant 9$ disks whenever $k = 3$ has an upper bound of $T_3(n) = 2n^2 + 24n - 153$.*

Proof. To obtain this bound, we will divide it up into the following steps.

- Building a stack of 3
- Building stacks of $4, 5, \cdots, (n - 4)$
- Building a stack of $(n - 3)$
- Building a stack of $(n - 2)$
- Building a stack of $(n - 1)$
- Building a stack of n

Note that this bound assumes a starting number of disks $n \geqslant 9$. Also, for the previous bounds, we focused on totaling the number of saves at various steps in completing those puzzles and then subtracting these saves from the number of moves that would have been made when using only the classic rules. However, to obtain the bound for $k = 3$, we will simply total the number of moves that we make as we complete the puzzle.

Building a Stack of 2 Disks:

We build a stack of 3 disks on the destination peg using the classic rules. This takes 7 moves.

Building Stacks of 4, 5, $\cdots$, $(n-4)$ Disks:

To build a stack of 4 disks, we move the $(n-3)^{\text{rd}}$ largest disk from the source peg onto the intermediate peg. This takes 1 move. Then we flip the nth and $(n-1)^{\text{st}}$ largest disks from the destination peg onto the source peg. This takes 2 moves. We then move the $(n-2)^{\text{nd}}$ largest disk from the destination peg onto the intermediate peg. This takes 1 move. Next, we flip the nth and $(n-1)^{\text{st}}$ largest disks from the source peg onto the intermediate peg. This takes 2 moves. Thus, it takes

$$1+2+1+2=2(3)$$

moves to build a stack of 4 disks. See Figure 24.

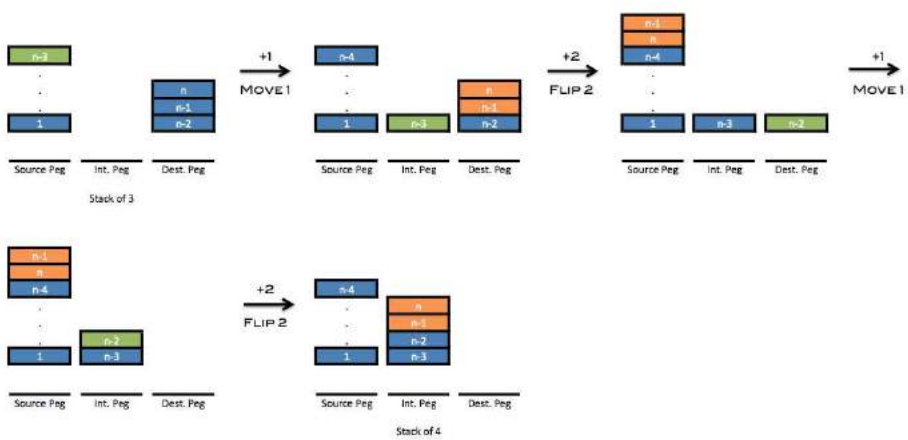

Figure 24: Overview of Building a Stack of 4 Disks

We repeat this same process to build the next stack of 5 disks. We begin by moving the $(n-4)^{\text{th}}$ largest disk from the source peg onto the destination peg. We then flip all disks, except the $(n-3)^{\text{rd}}$ largest disk, from the intermediate peg onto the source peg. This takes 3 moves. We then move the $(n-3)^{\text{rd}}$ largest disk onto the destination peg. This takes 1 move. Next, we flip the same disks that were flipped onto the source peg, onto the destination peg. This takes another 3 moves, and gives us a stack of 5. Thus, it takes

$$1+3+1+3=2(4)$$

moves to build a stack of 5 disks. See Figure 25.

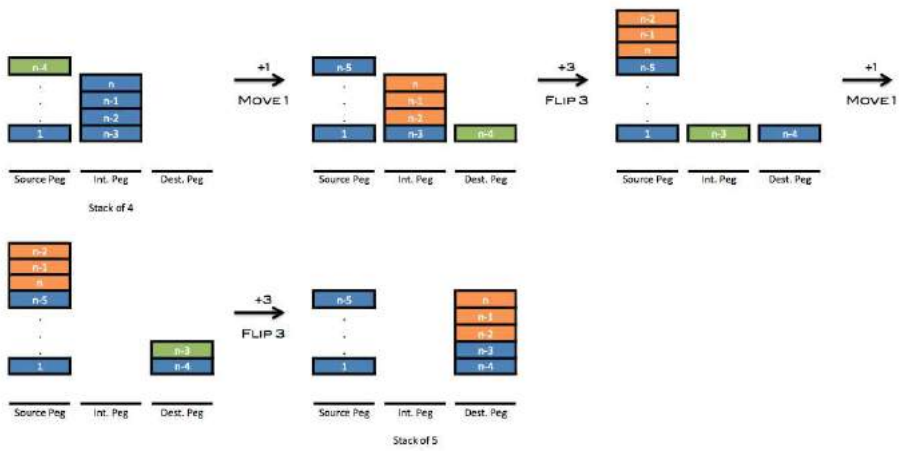

Figure 25: Overview of Building a Stack of 5 Disks

In general, to build a stack of x disks, for $4 \leqslant x \leqslant (n-4)$, it takes 1 move to relocate the largest disk of the stack we are building, $(x-2)$ moves to flip disks, 1 move to put the 2nd largest disk of the stack we are building onto the largest disk of the stack we are building, and $(x-2)$ more moves to flip the remaining disks onto the stack we are building. That is, it takes

$$1 + (x-2) + 1 + (x-2) = 2(x-1)$$

moves to build a stack of x disks. See Figure 26.

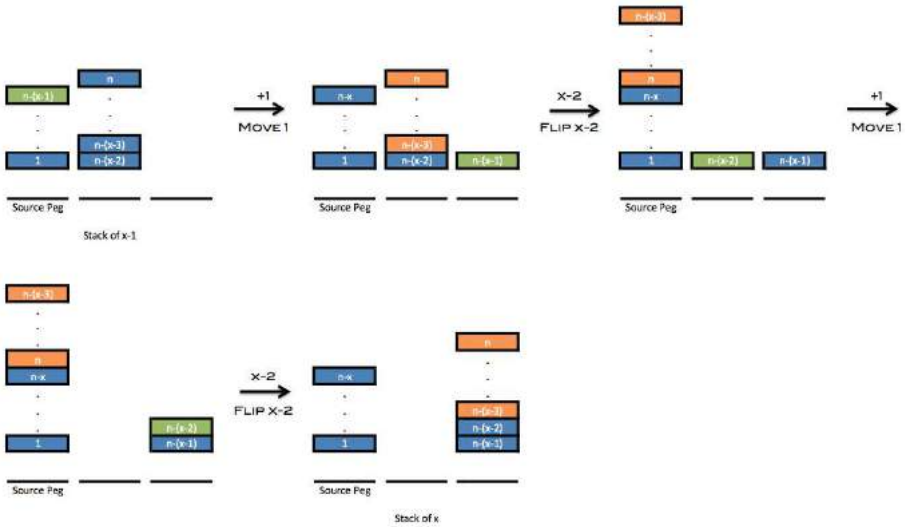

Figure 26: Overview of Building a Stack of x Disks. Stacks of an even number of disks will be on the intermediate peg and stacks of an odd number of disks will be on the destination peg. The final stack of $(n-4)$ disks will be on the third peg.

So, it takes

$$\sum_{i=4}^{n-4}(2(i-1)) = n^2 - 9n + 14$$

moves to build stacks of $4, 5, \ldots, (n-4)$.

Building a Stack of $(n-3)$ Disks:

After building a stack of $(n-4)$ disks, we reach the configuration where the first peg contains the 4th, 3rd, 2nd, and largest disks; the second peg contains no disks; and the third peg contains the remainder of the disks. We now need to build a stack of $(n-3)$ disks, and we start by moving the 4th largest disk onto the second peg. This takes 1 move. Next, we flip the top $(n-5)$ disks from the third peg onto the first peg. This takes $(n-5)$ moves. Then we move the 5th largest disk from the third peg onto the second peg. This takes 1 move. Next, we move the 6th largest disk from the first peg onto the second peg. This takes 1 move. Then we move the 7th largest disk onto the third peg. This takes 1 move. We then flip $(n-7)$ disks from the first peg onto the second peg, which takes $(n-7)$ moves. Lastly, we move the 7th largest disk from the third peg onto the second peg, giving us a stack of $(n-3)$ disks. See Figure 27.

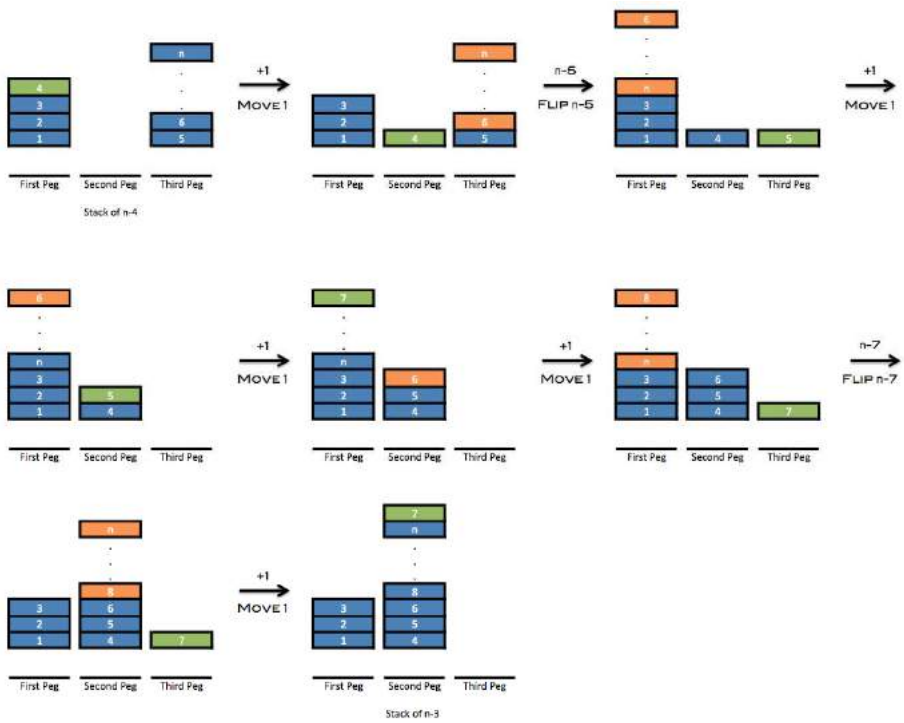

Figure 27: Overview of Building a Stack of $(n-3)$ Disks. Notice that we make a counterintuitive move by taking out the 7th largest disk. Doing this will save many moves in the long term.

Thus, it takes

$$1 + (n-5) + 1 + 1 + 1 + (n-7) + 1 = 2n - 7$$

moves to build a stack of $(n-3)$ disks.

Building a Stack of $(n-2)$ Disks:

We now build a stack of $(n-2)$ disks. First, we move the 3rd largest disk onto the third peg. This takes 1 move. Next, move the 7th largest disk from the second peg onto the first peg. This takes 1 move. Flip $(n-7)$ disks from the second peg onto the first peg. This takes $(n-7)$ moves. Move the 6th largest disk from the second peg onto the third peg. This takes 1 move. Flip $(n-7)$ disks from the first peg onto the second peg. This takes $(n-7)$ moves. Move the 7th largest disk from the first peg onto the third peg. This takes 1 move. Flip $(n-7)$ disks from the second peg onto the third peg. This takes $(n-7)$ moves. Move the 5th largest disk from the second peg onto the first peg. This takes 1 move. Move the 8th largest disk from the third peg onto the second peg. This takes 1 move. Flip $(n-8)$ disks from the third peg onto the first peg. This takes $(n-8)$ moves. Move the 8th largest disk from the second peg onto the first peg. This takes 1 move. Flip the 7th largest and 6th largest disks from the third peg onto the first peg. This takes 2 moves. Move the 4th largest disk from the second peg onto the third peg. This takes 1 move. Flip $(n-5)$ disks from the first peg onto the second peg. This takes $(n-5)$ moves. Move the 5th largest disk from first peg onto the third peg. This takes 1 move. Flip $(n-5)$ disks from the second peg onto the third peg. This takes $(n-5)$ moves and completes the stack of $(n-2)$ disks. See Figure 28.

Thus, it takes

$$1+1+(n-7)+1+(n-7)+1+(n-7)+1+1+(n-8)+1+2+1+(n-5)+1+(n-5) = 6n - 28$$

moves to build stack of $(n-2)$ disks.

Building a Stack of $(n-1)$ Disks:

We now build a stack of $(n-1)$ disks. Due to the length of this step, we break the sequence of moves up into three sections. We start by moving the 2nd largest disk from the first peg onto the second peg. This takes 1 move. Then we flip the 6th largest and 7th largest disks from the third peg to the first peg. This takes 2 moves. Move the 8th largest disk from the third peg onto the second peg. This takes 1 move. Flip $(n-8)$ disks from the third peg onto the first peg. This takes $(n-8)$ moves. Move the 8th largest disk from the second peg onto the first peg. This takes 1 move. Move the 5th largest disk from the third peg onto the second peg. This takes 1 move. Flip $(n-7)$ disks from the first peg onto the second peg. This takes $(n-7)$ moves. Move the 7th largest disk from the first peg onto the third peg. This takes 1 move. Flip $(n-7)$ disks

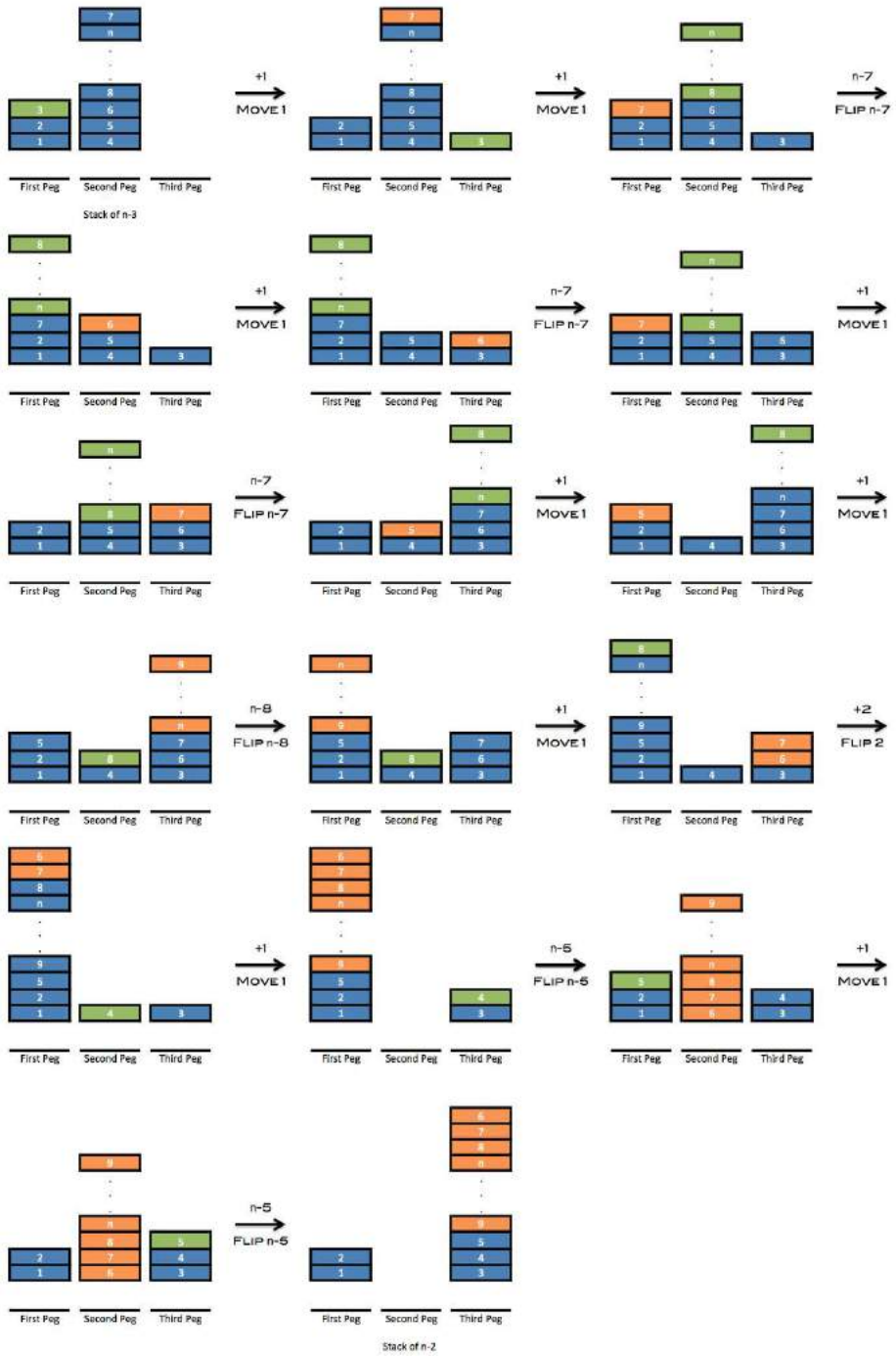

Figure 28: Overview of Building a Stack of $(n-2)$ Disks. Whenever $n = 9$, the disk labelled n coincides with the disk labelled 9.

from the second peg onto the third peg. This takes $(n-7)$ moves. Move the

6th largest disk from the first peg onto the second peg. This takes 1 move. Flip $(n-7)$ disks from the third peg onto the second peg. This takes $(n-7)$ moves. Move the 7th largest disk from the third peg onto the second peg. This takes 1 move. Move the 4th largest disk from the third peg onto the first peg. This takes 1 move. Flip $(n-6)$ disks from the second peg onto the first peg. This takes $(n-6)$ moves. See Figure 29.

Move the 6th largest disk from the second peg onto the third peg. This takes 1 move. Figure 30 will begin after this move is made.

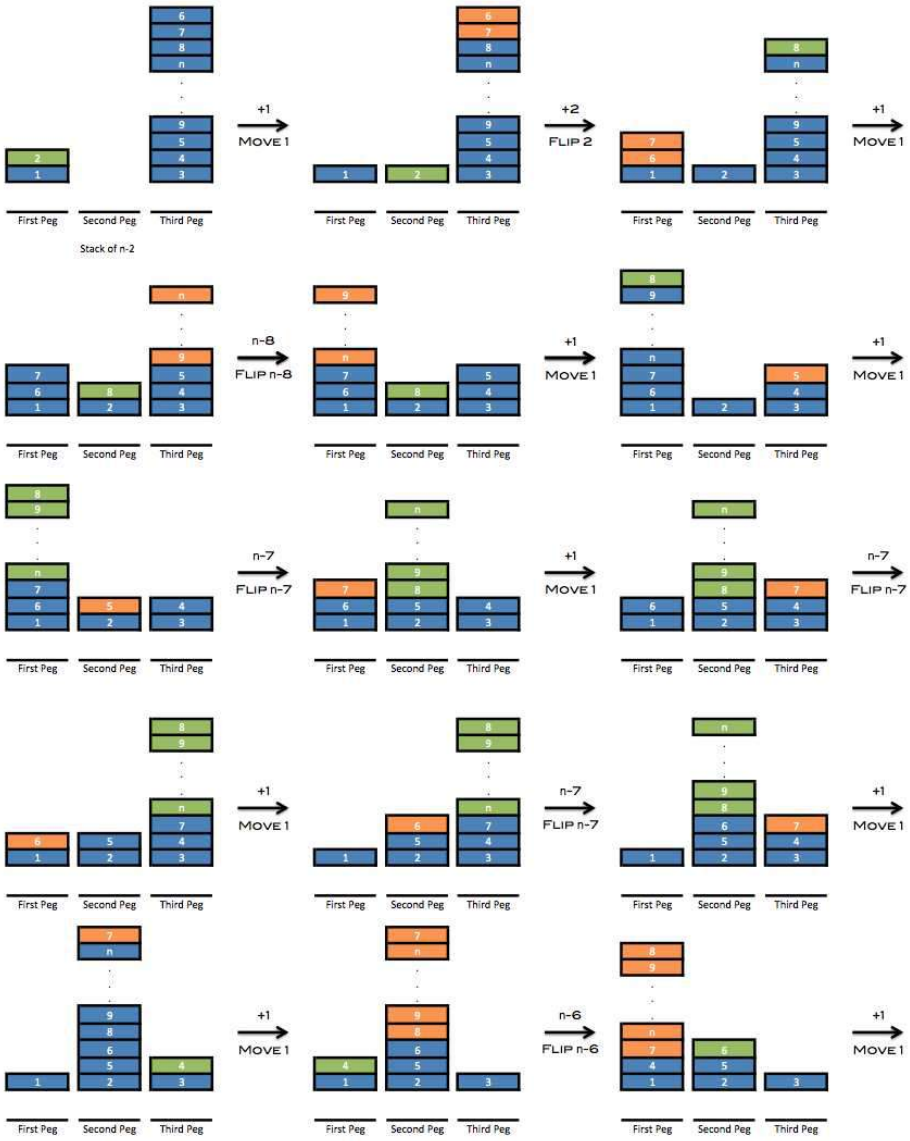

Figure 29

Flip $(n-7)$ disks from the first peg onto the second peg. This takes $(n-7)$ moves. Move the 7th largest disk from the first peg onto the third peg. This takes 1 move. Flip $(n-7)$ disks from the second peg onto the third peg. This takes $(n-7)$ moves. Move the 5th largest disk from the second peg onto the first peg. This takes 1 move. Flip $(n-7)$ disks from the third peg onto the first peg. This takes $(n-7)$ moves. Flip the 7th largest and 6th largest disks from the third peg onto the first peg. This takes 2 move. Move the 3rd largest disk from the third peg onto the second peg. This takes 1 move. Move the 6th largest disk from the first peg onto the second peg. This takes 1 move. Move the 7th largest disk from the first peg onto the third peg. This takes 1 move. Flip $(n-7)$ disks from the first peg onto the second peg. This takes $(n-7)$ moves. Move the 7th largest disk from the third peg onto the second peg. This takes 1 move. Move the 5th largest disk from the first peg onto the third peg. This takes 1 move. Move the 7th largest disk from the second peg onto the first peg. This takes 1 move. Move the 8th largest disk from the second peg onto the third peg. This takes 1 move. See Figure 30.

Flip $(n-8)$ disks from the second peg onto the first peg. This takes $(n-8)$ moves. Figure 31 will begin after this move is made.

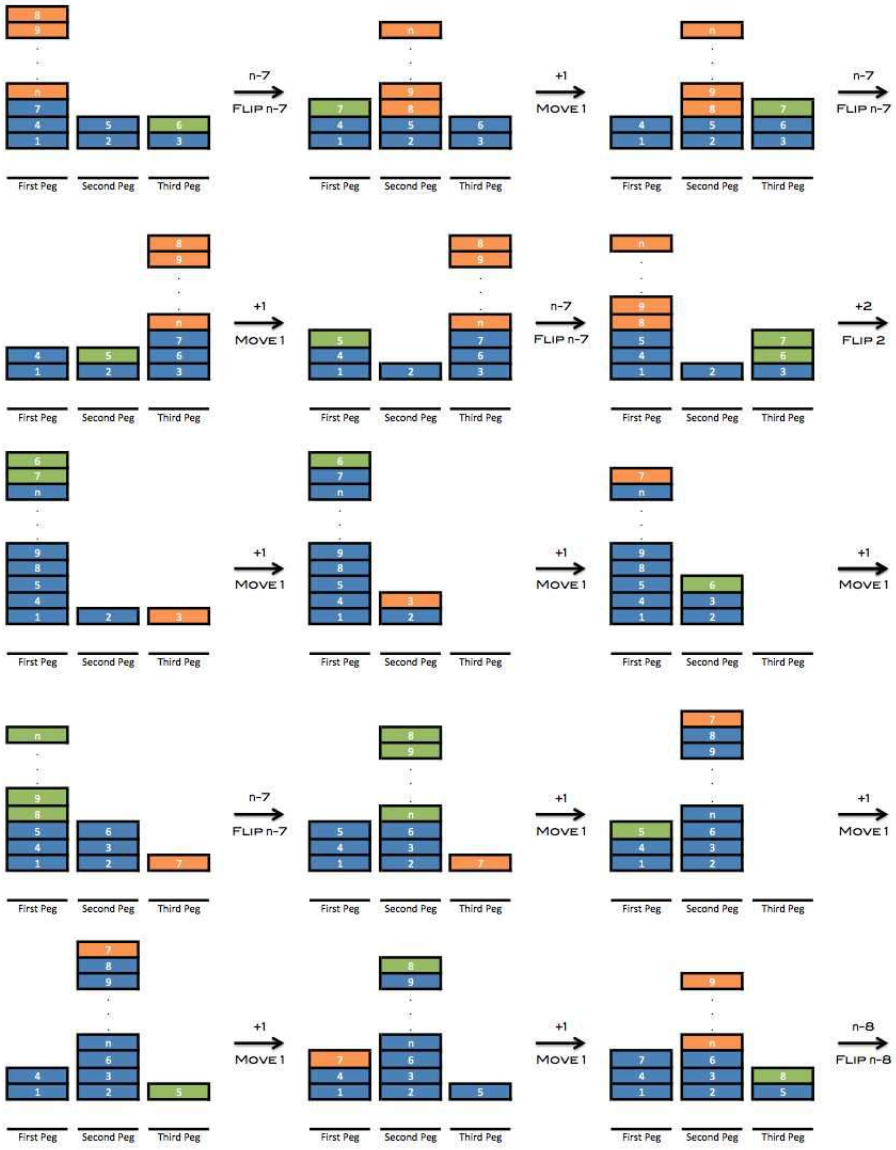
n-7
FLIP n-7
+1
MOVE 1
n-7
FLIP n-7
+1
MOVE 1
n-7
FLIP n-7
+2
FLIP 2
+1
MOVE 1
+1
MOVE 1
+1
MOVE 1
n-7
FLIP n-7
+1
MOVE 1
+1
MOVE 1
+1
MOVE 1
+1
MOVE 1
n-8
FLIP n-8
First Peg
Second Peg
Third Peg

Figure 30

Move the 8th largest disk from the third peg onto the first peg. This takes 1 move. Move the 6th largest disk from the second peg onto the third peg. This takes 1 move. Flip $(n-7)$ disks from the first peg onto the second peg. This takes $(n-7)$ moves. Move the 7th largest disk from the first peg onto the third peg. This takes 1 move. Flip $(n-7)$ disks from the second peg onto the third peg. This takes $(n-7)$ moves. Move the 4th largest disk from the first peg onto the second peg. This takes 1 move. Flip $(n-4)$ disks from the third peg onto the second peg. This takes $(n-4)$ moves. See Figure 31.

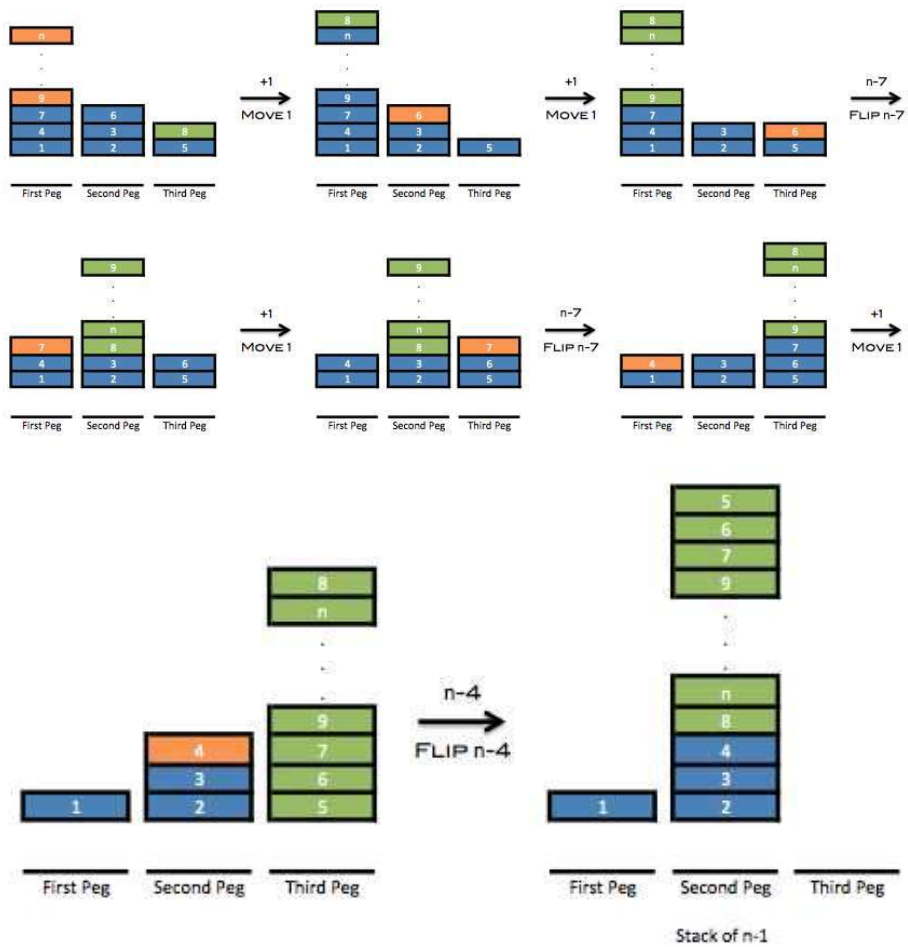

Figure 31

Thus, it takes

1+2+1+(n-8)+1+1+(n-7)+1+(n-7)+1+(n-7)+1+1+(n-6)+1+(n-7)+1+(n-7)+1
+(n-7)+2+1+1+1+(n-7)+1+1+1+1+(n-8)+1+1+(n-7)+1+(n-7)+1+(n-4)=13n-63

moves to build a stack of $(n-1)$ disks.

Building a Stack of n Disks:

Now, we wish to form a stack of n. We do this by moving the largest disk onto the third peg and then reversing the previous steps for building stacks of $1, 2, \cdots, (n-1)$. Thus, it takes

$$1+\big(7+(n^2-9n+14)+(2n-7)+(6n-28)+(13n-63)\big) = 1+(n^2+12n-77)$$

to build a stack of n disks. See Figure 32.

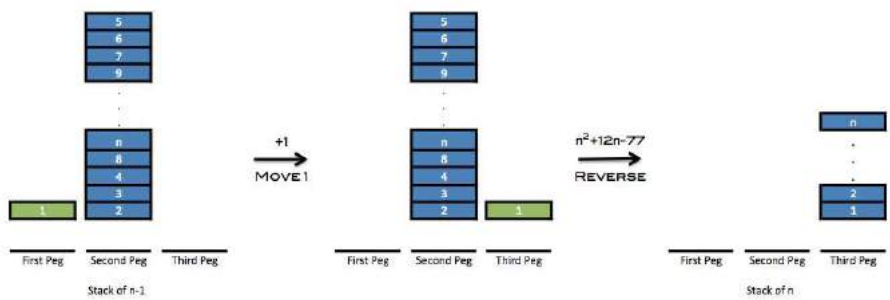

Figure 32: Overview of Building a Stack of n Disks

Therefore, it takes

$$(n^2+12n-77)+1+(n^2+12n-77) = 2n^2+24n-153$$

moves to complete the puzzle.

□

The optimal solution for the k-Tower of Hanoi puzzle for $n = 8$ disks whenever $k = 3$ has an upper bound of 161.

This can be shown by disregarding two steps for the bound of $n \geqslant 9$, which gives a bound of 161.

Further Work

As the value of k increases, the number of steps needed to complete the puzzle also increases. We have noticed a relationship between the sequences of moves for various values of k. We can explore these relationships between the bounds to find a general solution to the k-Tower of Hanoi puzzle for an arbitrary number of disks and an arbitrary value of k.

References

[1] Gregg, S.B., Orozco, J. "The Broncho Tower a modification of the Tower of Hanoi puzzle", *Rose-Hulman Undergrad. Math. J.*, 4, no. 2, 47–58, 2013.

[2] "The Tower of Stanford", Problems and Solutions, *American Mathematical Monthly*, 111, no. 4, 364–365, April 2004.