Mathematics for Computer Science
CSE 401

Shah Murtaza Rashid Al Masud, PhD
Department of CSE
University of Asia Pacific

Recursion: recursion is a method of solving a problem where the solution depends on solutions to smaller instances of the same problem.

What is a recurrent problem?

It is a problem that keeps occurring on a too-frequent basis.

If a problem happens once and there is a reason to suspect that it might happen again, decisive action should be taken to prevent recurrence.

Recurrent Problems

Tower of Hanoi

Lines in the plane

Josephus Problem

Tower of Hanoi

The Tower of Hanoi puzzle was invented by the French mathematician Édouard Lucas in 1883.

The board has **three** pegs/rod.

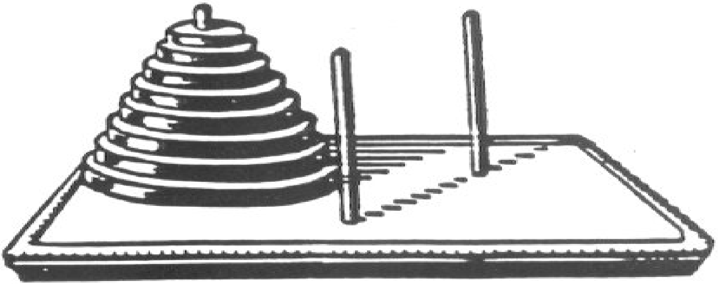
The **tiles** are ***n* disks**, all of different sizes, with a hole in the middle so that they can be put on the pegs.

At the beginning of the game, the disks are all on the first peg, in decreasing order from bottom to top (larger at the bottom, smaller at the top)..

The aim of the game is to transfer all the disks/entire Tower to one of the other pegs, moving only one disk at a time and never moving a larger one onto a smaller disk.

The Tower of Hanoi

Edouard Lucas formulation involved three pegs and eight distinctly-sized disks stacked on one of the pegs from the biggest on the bottom to the smallest on the top, like s



The GOAL

The puzzle goal is to move the stack of disks to one of the other pegs with the following rule:

L - rule:

must move one disk at a time;

a larger disk cannot be on top of any smaller disks at any time do it in as few moves as possible

Lucas furnished his puzzle with a romantic legend about Tower of Brahma (64 disks) with monks, gold, diamond needles etc...

The Tower of Hanoi GENERALIZED

Tower has now *n* disks, all stacked in decreasing order from bottom to top on one of three pegs,

Question:

what is the minimum number of (legal) moves needed to move the stack to one of the other pegs?

Plan:

We start by expressing the minimum number of moves required to move a stack of *n* disks as a recurrence relation,

i.e. find and prove a recursive (recurrent) formula

The Tower of Hanoi GENERALIZED to n disks

We denote by

T_n - the minimum number of moves that will transfer n disks from one peg to another under the

L - rule:

must move one disk at a time;

a larger disk cannot be on top of any smaller disks at any time do it in as few moves as possible

$n = 1$, we have 1 disk and 1 move, i.e. $T_1 = 1$

$n = 2$, we have 2 disks and 3 moves: top (smaller) disk from peg 1 to peg 2, remaining (larger) disk from peg 1 to peg 3, the disk from peg 2 (smaller) on the top of the disk (larger) on peg 3 so L - rule holds and hence $T_2 = 3$

Recursive Formula for T_n - end of the proof

These two inequalities $T_n \leq 2T_{n-1} + 1$ and $T_n \geq 2T_{n-1} + 1$, together with the trivial solution for n = 0, yield

$$T_0 = 0;$$
$$T_n = 2T_{n-1} + 1, \qquad \text{for } n > 0.$$

So how do we solve a recurrence? One way is to guess the correct solution, then to prove that our guess is correct. And our best hope for guessing the solution is to look (again) at small cases. So we compute, successively, $T_3 = 2\cdot 3 + 1 = 7$; $T_4 = 2\cdot 7 + 1 = 15$; $T_5 = 2\cdot 15 + 1 = 31$; $T_6 = 2\cdot 31 + 1 = 63$. Aha! It certainly looks as if

$$T_n = 2^n - 1, \qquad \text{for } n \geqslant 0.$$

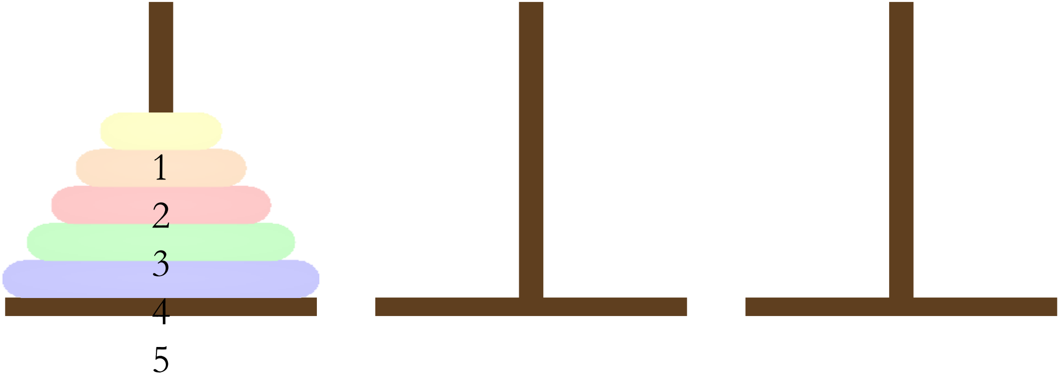
The Tower of Hanoi: Solution

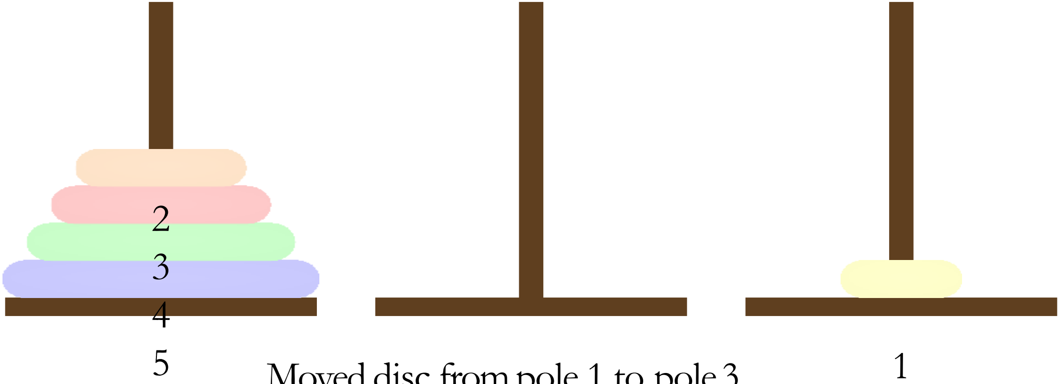
Recurrences are ideally set up for mathematical induction. Using mathematical induction the following can be proved:

For the Tower of Hanoi puzzle with $n \geqslant 0$, the minimum number of moves needed is:

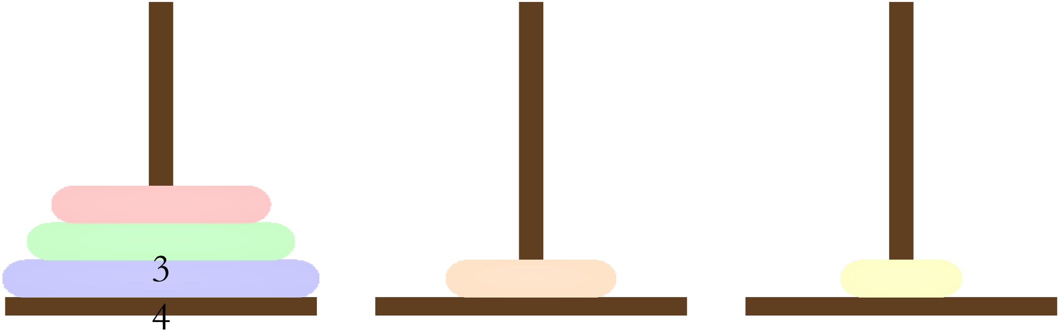
$$T_n = 2^n - 1\,.$$

Let's look at the example borrowed from Martin Hofmann and Berteun Damman.

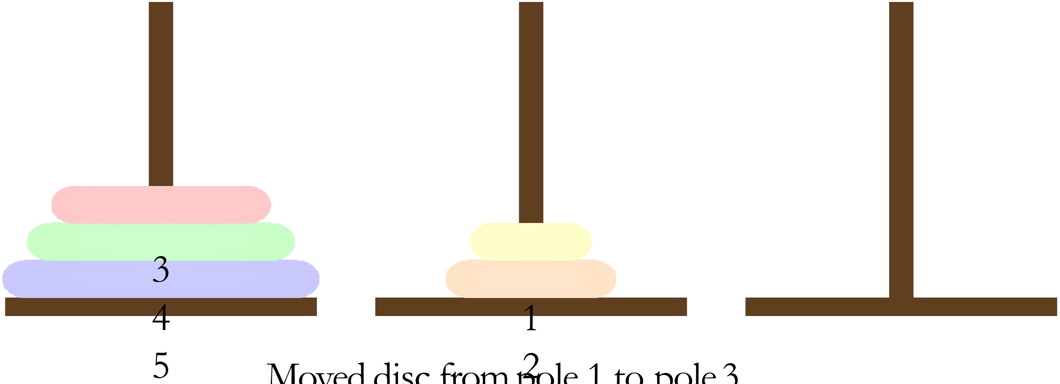

1
2
3
4
5



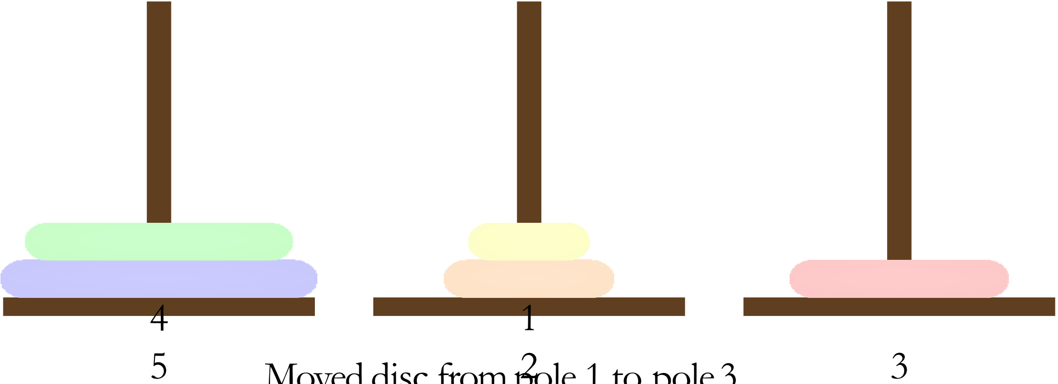
Moved disc from pole 1 to pole 3.



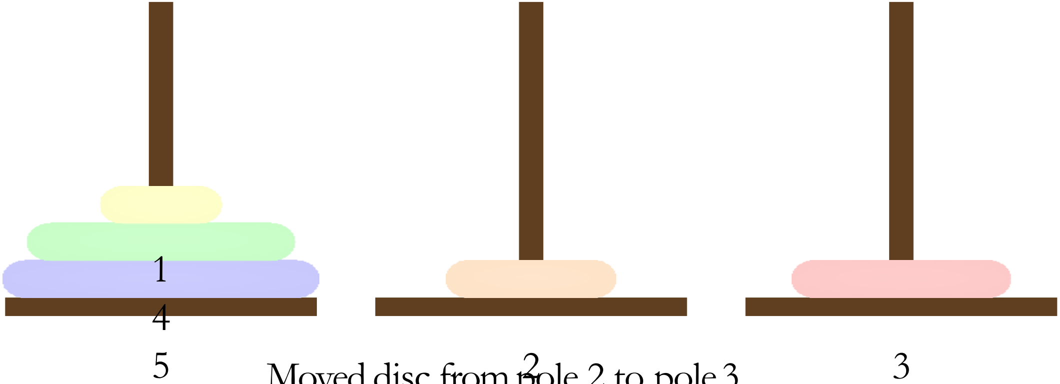
Moved disc from pole 1 to pole 2.



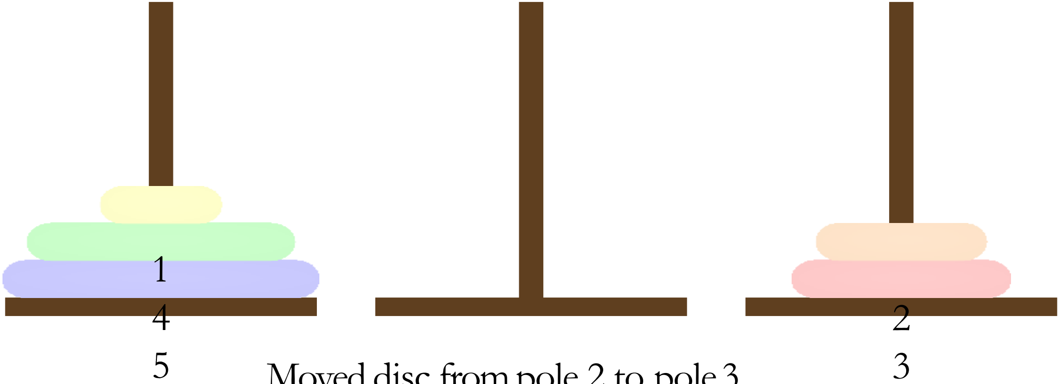
Moved disc from pole 1 to pole 3.



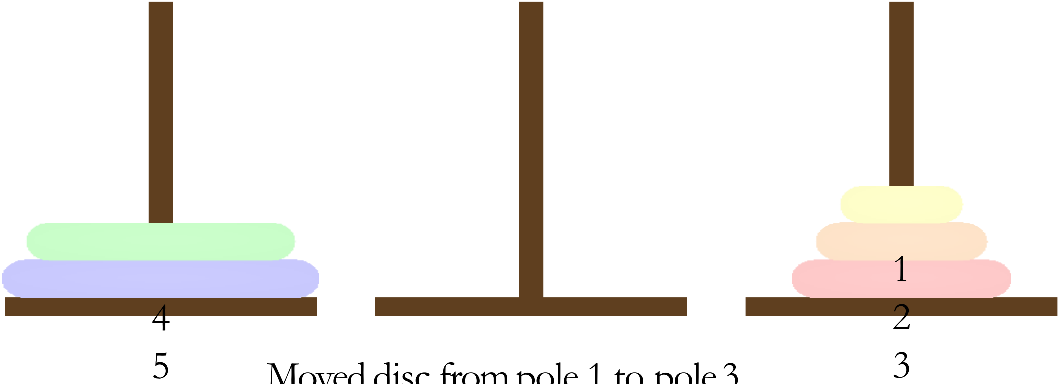
Moved disc from pole 1 to pole 3.



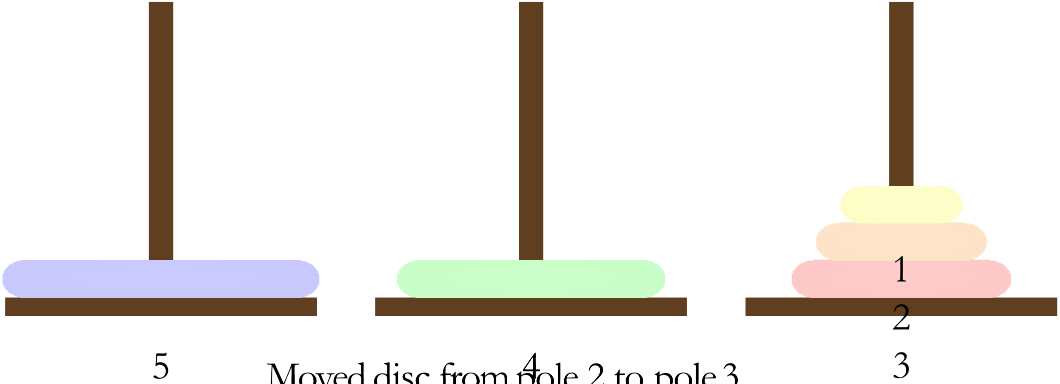
Moved disc from pole 2 to pole 3.



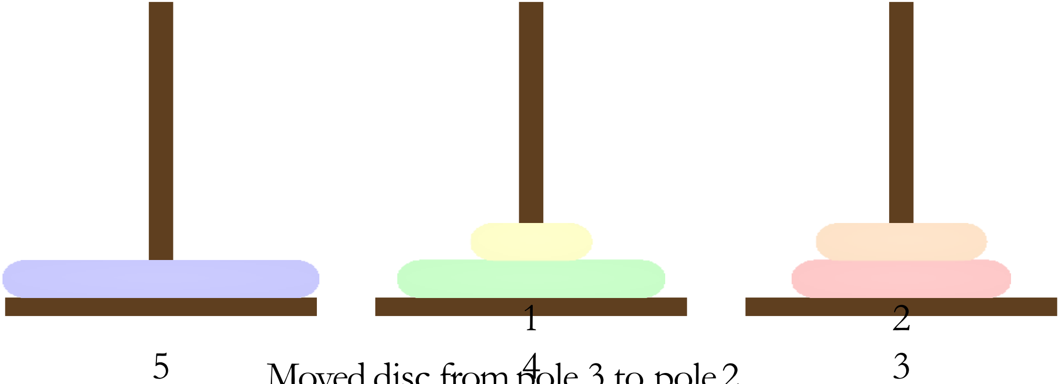
Moved disc from pole 2 to pole 3.



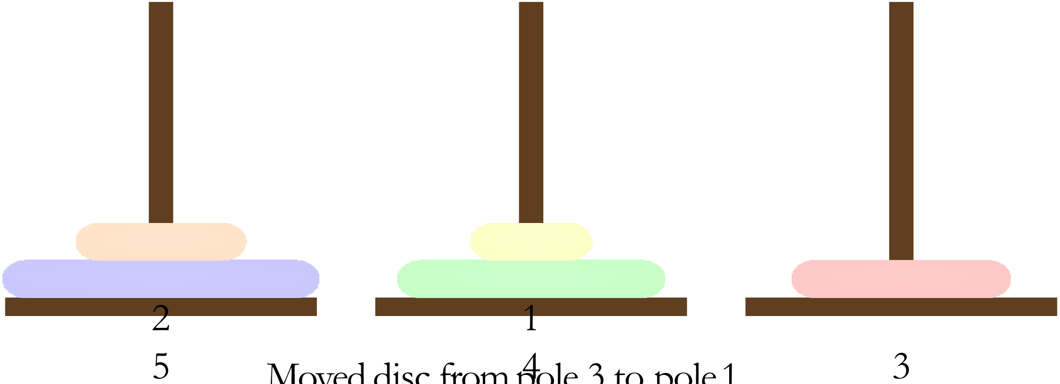
Moved disc from pole 1 to pole 3.



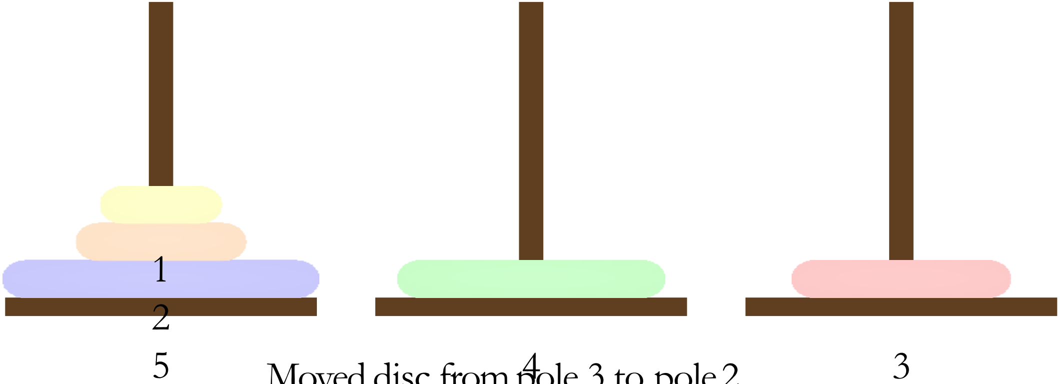
Moved disc from pole 2 to pole 3.



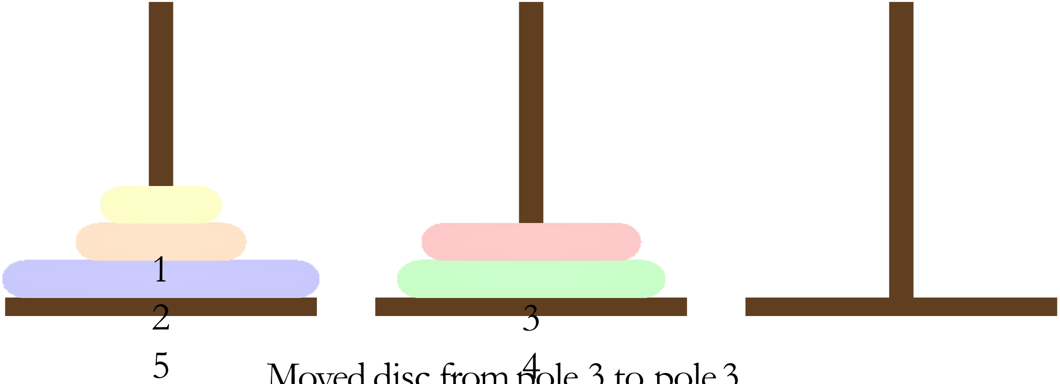
Moved disc from pole 3 to pole 2.



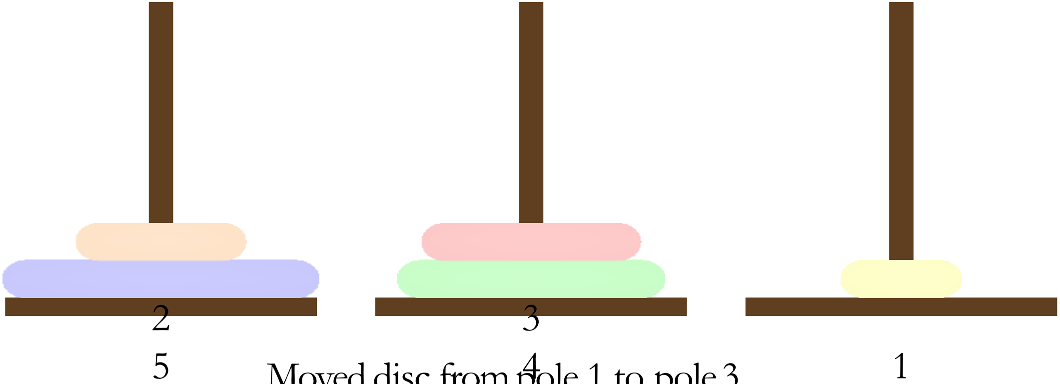
Moved disc from pole 3 to pole 1.



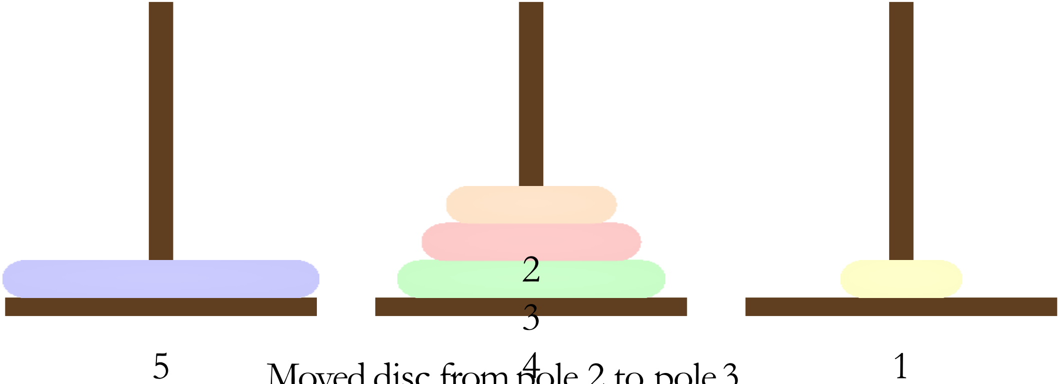
Moved disc from pole 3 to pole 2.



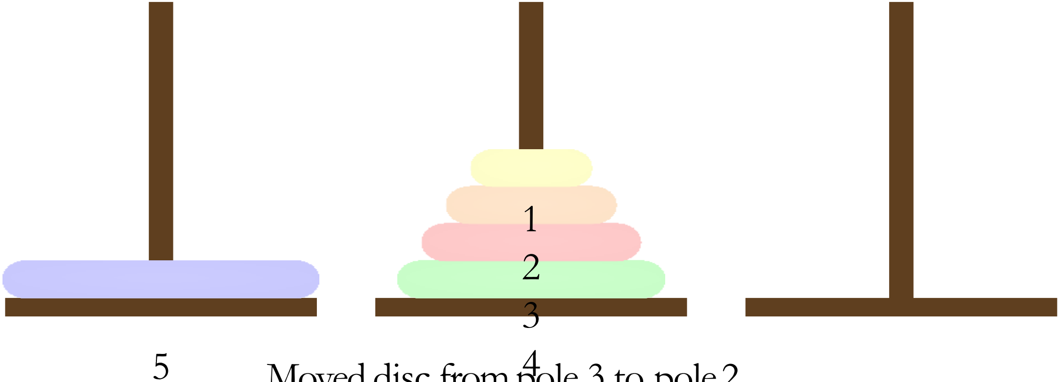
Moved disc from pole 3 to pole 3.



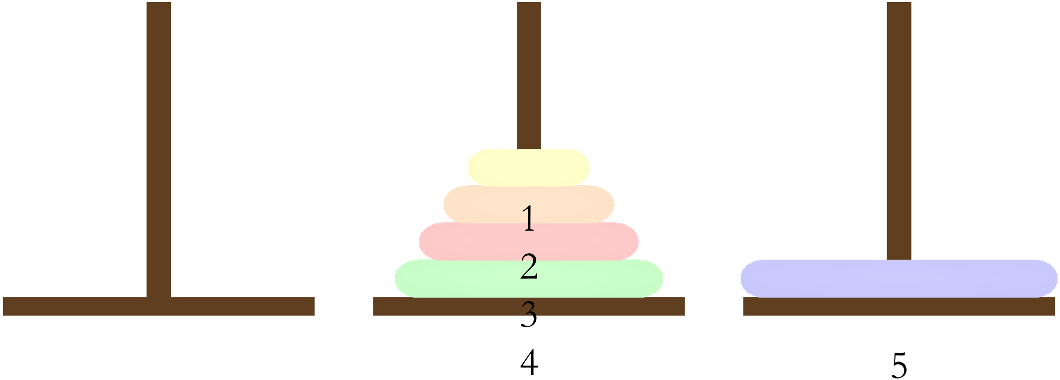
Moved disc from pole 1 to pole 3.

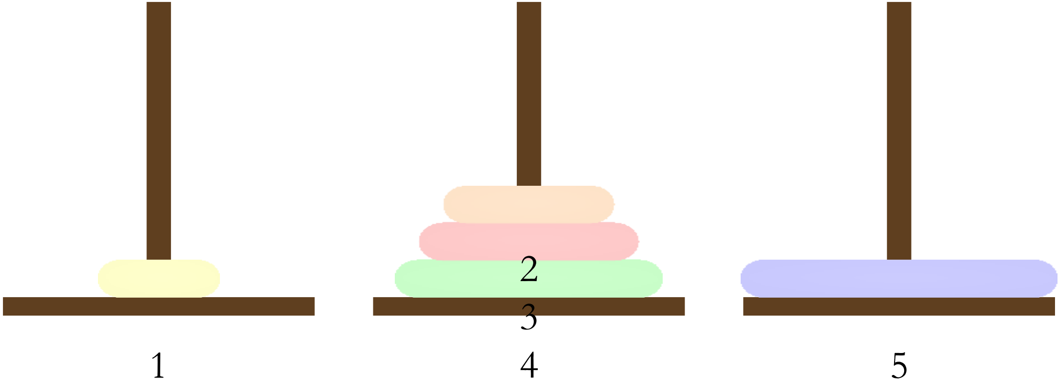


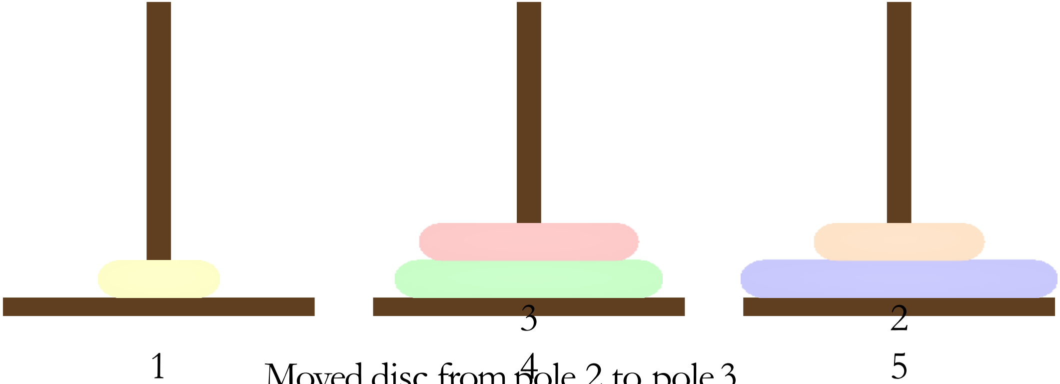
Moved disc from pole 2 to pole 3.



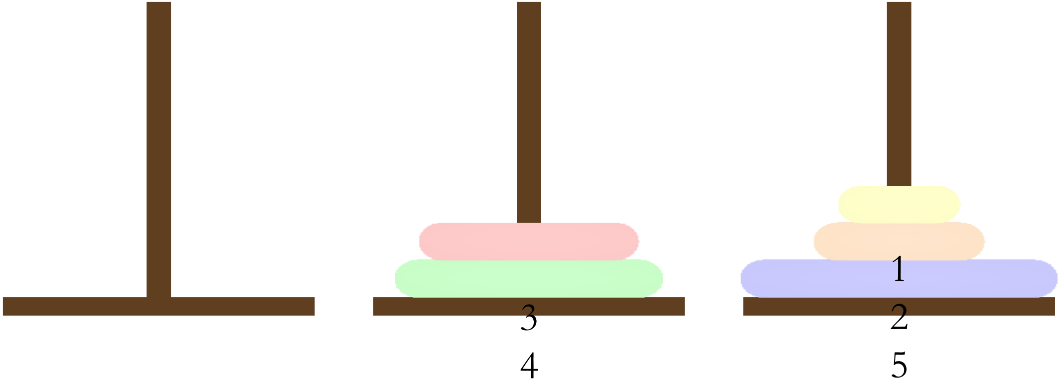
Moved disc from pole 3 to pole 2.

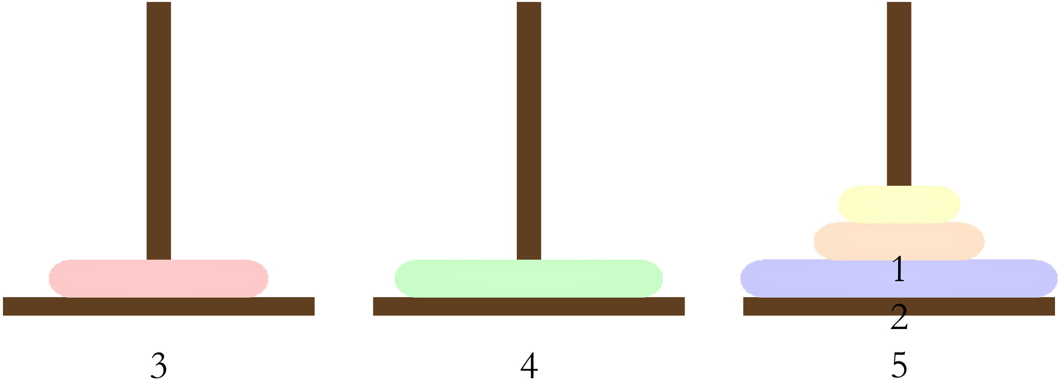

1
2
3
4
5

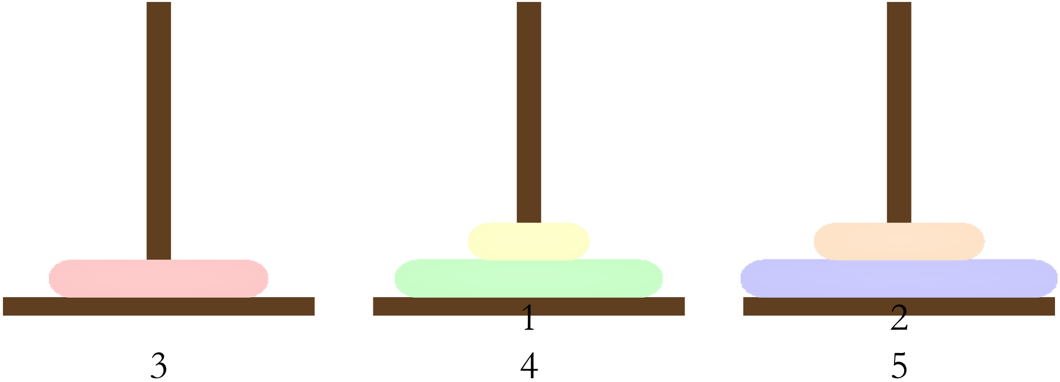

2
3
1
4
5

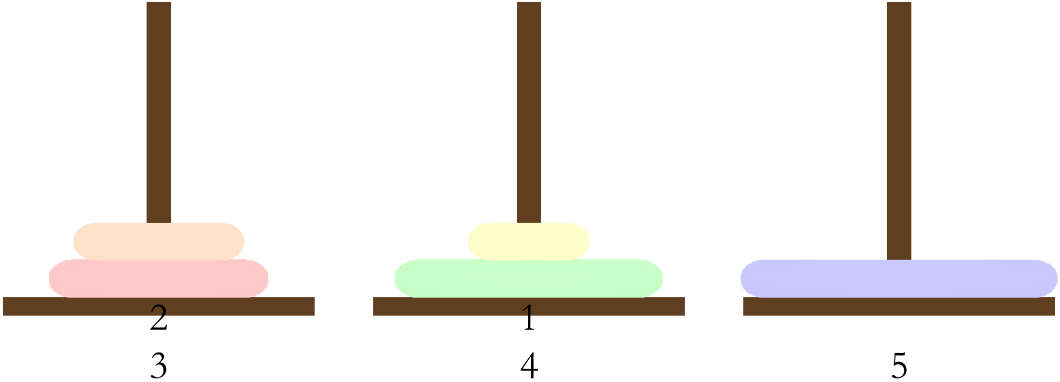


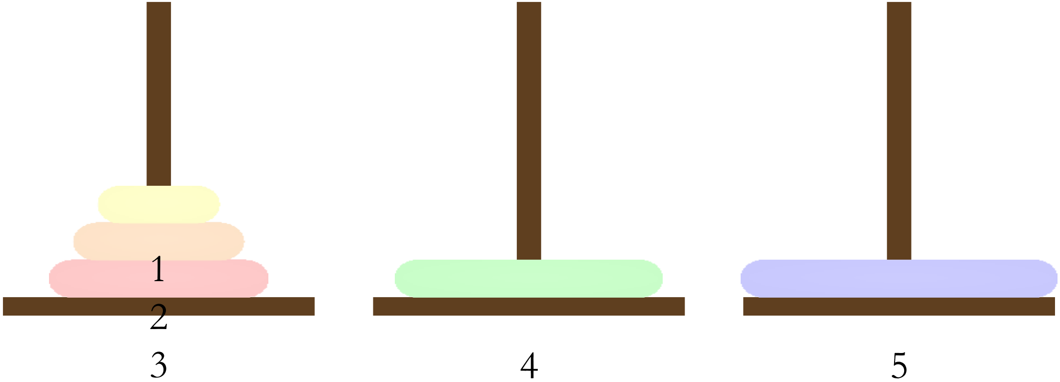
Moved disc from pole 2 to pole 3.

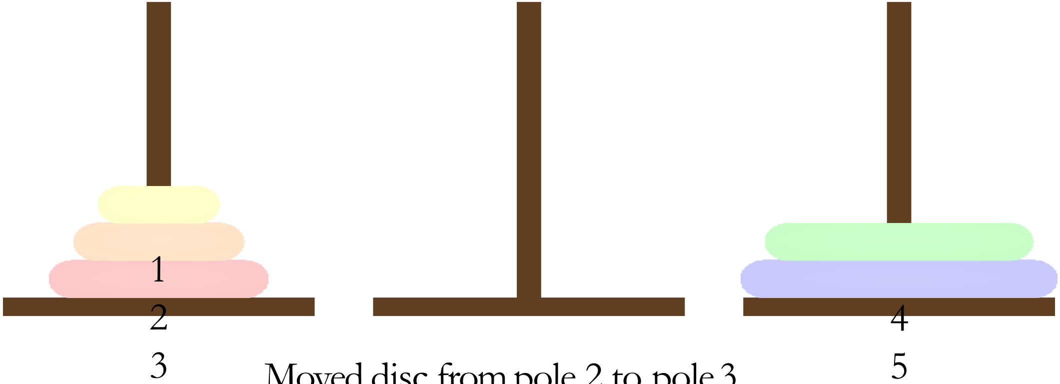

1
3
2
4
5


1
2
3
4
5


1
2
3
4
5

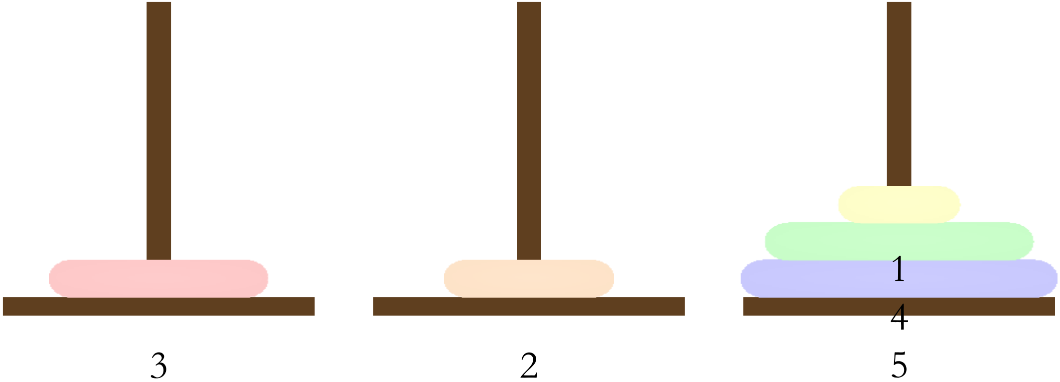

2
3
1
4
5

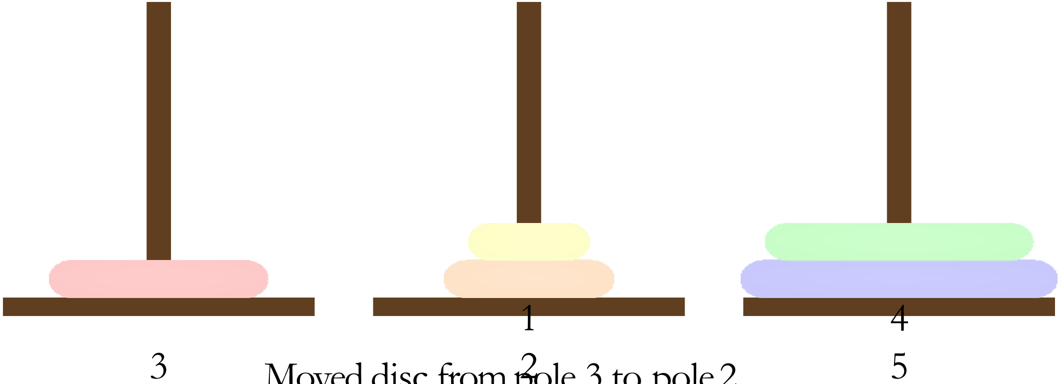

1
2
3
4
5



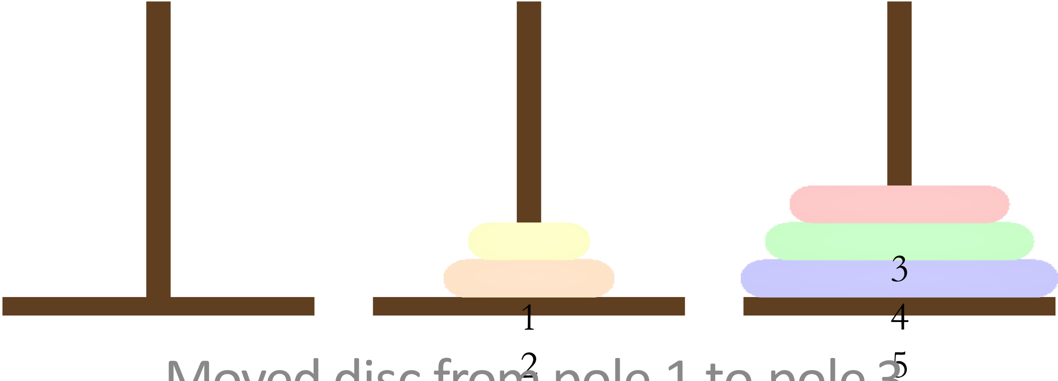
Moved disc from pole 2 to pole 3.


2
3
1
4
5

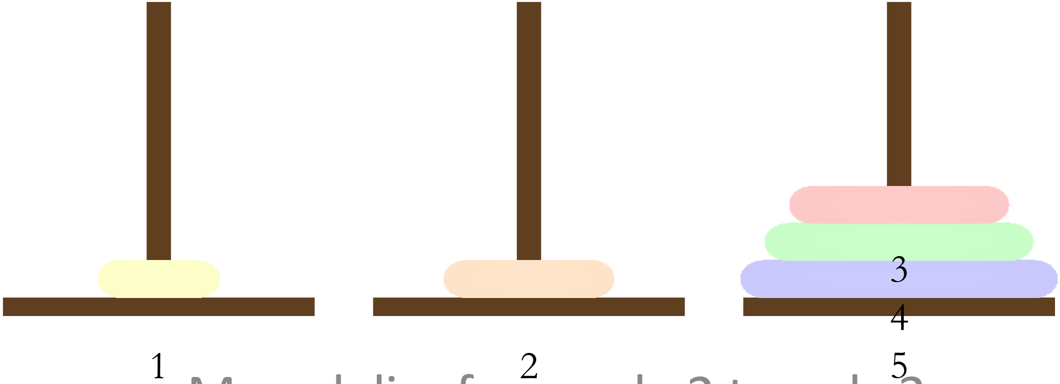

1
4
3
2
5



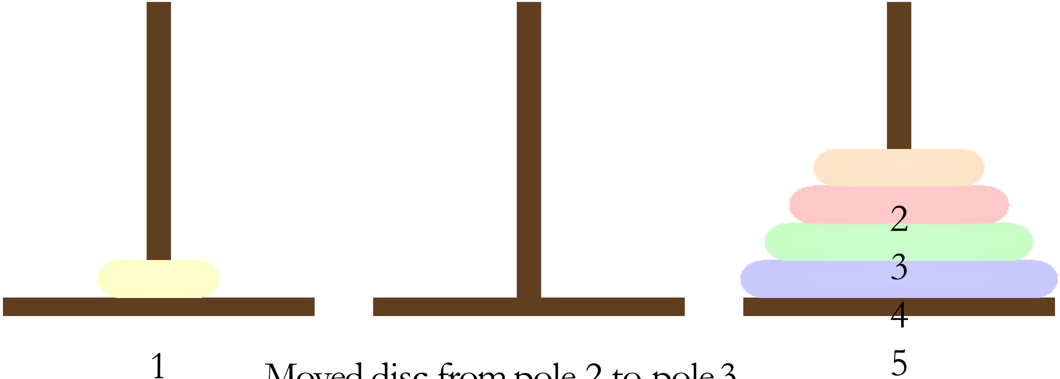
Moved disc from pole 3 to pole 2.



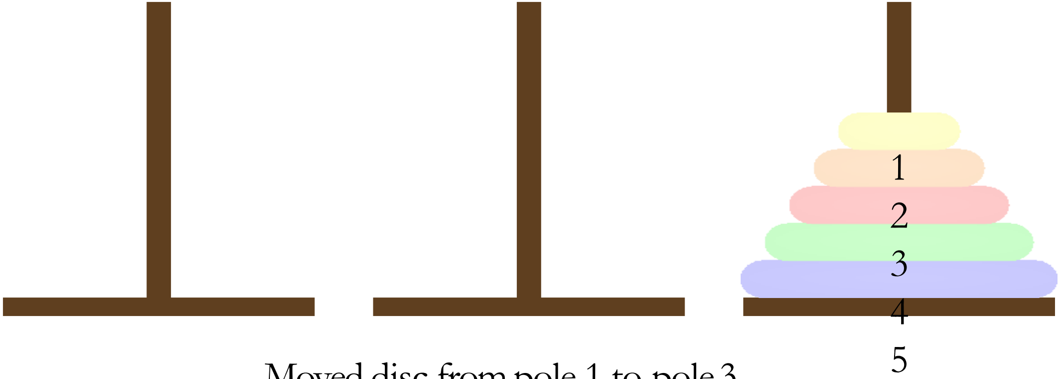
Moved disc from pole 1 to pole 3.



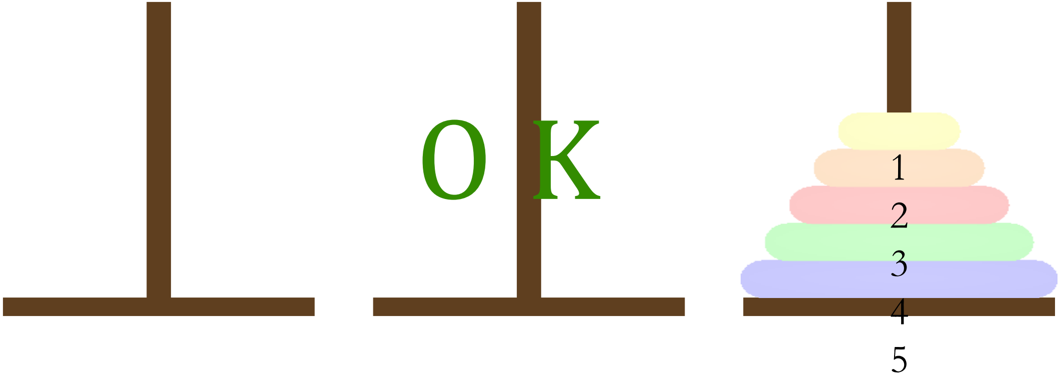
Moved disc from pole 2 to pole 3.



Moved disc from pole 2 to pole 3.



Moved disc from pole 1 to pole 3.


OK
1
2
3
4
5

By inductive hypothesis, $(n+1)$ disks require:

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

moves.

Find the shortest sequence of moves that transfers a tower of n disks from the left peg A to the right peg B, if direct moves between A and B are disallowed. Here, A is the start peg, B the stop peg, and C the spool peg.

Solution. For $n = 1$ the shortest sequence is $A \to C$, $C \to B$. For $n = 2$ it is:

1. $A \to C$.
2. $C \to B$.
3. $A \to C$.
4. $B \to C$. Note that the whole tower is on peg C now.
5. $C \to A$.
6. $C \to B$.
7. $A \to C$.
8. $C \to B$.

For the general case, observe that the strategy that solves the problem for n disks works as follows:

1. Move the upper tower of $n - 1$ disks on peg B
2. Move the n-th disk to peg C.
3. Move the upper tower of $n - 1$ disks on peg A.
4. Move the n-th disk to peg B.
5. Move the upper tower of $n - 1$ disks on peg B.

Then the number X_n of moves needed by the strategy to solve the problem with n disks satisfies $X_0 = 0$ and $X_n = 3X_{n-1} + 2$ for every $n > 0$. It is easy to see that the only solution is $X_n = 3^n - 1$.

Lines in Plane

Maximum number of regions on a plane formed due to non-parallel lines

Lines in the Plane

The problem of Lines in the Plane was proposed by JACOB STEINER, Swiss mathematician in 1826

PROBLEM: what's the **maximum** number of regions L_n that can be defined in the plane by n lines?

For $n = 0$, it's easy to see that there's only one region i.e. $L_0 = 1$.

For $n = 1$ there're two regions no matter how the line's oriented $L_1 = 2$.

Lines in the Plane

If $n = 2$, then the maximum number of regions we can define is $L_2 = 4$

Four regions is the best we can do with two lines because the lines must either cross or not cross; if they cross, then the lines define four regions, and if they don’t cross they define three.

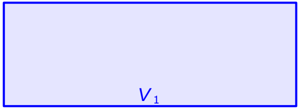
Lines in the Plane

Since we have $L_0 = 1$, $L_1 = 2$, and $L_2 = 4$, one might be led to conjecture that $L_n = 2^n$.

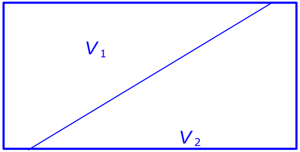
This immediately breaks down when we consider 3 lines - $n = 3$.

No matter how the third line is placed, we can only split at most three pre-existing regions, i.e. we can add at most three new regions using the third line and $L_3 = 7$

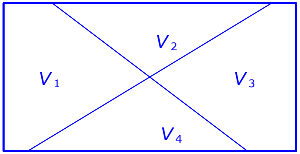
Lines in the Plane small cases



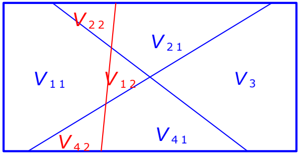
$$L_0 = 1$$



$$L_1 = 2$$



$$L_2 = 4$$



$$L_3 = L_2 + 3 = 7$$

Lines in the Plane

The above image shows the maximum number of regions a line can divide a plane. One line can divide a plane into two regions, two non-parallel lines can divide a plane into 4 regions and three non-parallel lines can divide into 7 regions, and so on.

Lines in the Plane

Now solve the recursion as follows:

$L(2) - L(1) = 2 \; \ldots \; (i)$

$L(3) - L(2) = 3 \; \ldots \; (ii)$

$L(4) - L(3) = 4 \; \ldots \; (iii)$

. . .

. . .

$L(n) - L(n\text{-}1) = n \; ; \; \ldots \; (n)$

Adding all the above equation we get,

$L(n) - L(1) = 2 + 3 + 4 + 5 + 6 + 7 + \ldots\ldots + n \; ;$

$L(n) = L(1) + 2 + 3 + 4 + 5 + 6 + 7 + \ldots\ldots + n \; ;$

$L(n) = 2 + 2 + 3 + 4 + 5 + 6 + 7 + \ldots\ldots + n \; ;$

$L(n) = 1 + 2 + 3 + 4 + 5 + 6 + 7 + \ldots\ldots + n + 1 \; ;$

$L(n) = n\,(\,n + 1\,)\,/\,2 + 1 \; ;$

Lines in the Plane

OR

The recurrence is therefore:

$$
\begin{aligned}
L_0 &= 1; \\
L_n &= L_{n-1} + n, \qquad \text{for } n > 0.
\end{aligned} \tag{1.4}
$$

The known values of L_1, L_2, and L_3 check perfectly here, so we'll buy this.

Now we need a closed-form solution. We could play the guessing game again, but 1, 2, 4, 7, 11, 16, . . . doesn't look familiar; so let's try another tack. We can often understand a recurrence by "unfolding" or "unwinding" it all the way to the end, as follows:

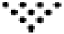
$$
\begin{aligned}
L_n &= L_{n-1} + n \\
&= L_{n-2} + (n-1) + n \\
&= L_{n-3} + (n-2) + (n-1) + n \\
&= L_0 + 1 + 2 + \cdots + (n-2) + (n-1) + n \\
&= 1 + S_n, \qquad \text{where } S_n = 1 + 2 + 3 + \ldots + (n-1) + n.
\end{aligned}
$$

Lines in the Plane

In other words, L_n is one more than the sum S_n of the first n positive integers.

The quantity S_n pops up now and again, so it's worth making a table of small values. Then we might recognize such numbers more easily when we see them the next time:

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14
S_n	1	3	6	10	15	21	28	36	45	55	66	78	91	105

These values are also called the triangular numbers, because S_n is the number of bowling pins in an n-row triangular array. For example, the usual four-row array  has $S_4 = 10$ pins.

To evaluate S_n we can use a trick that Gauss reportedly came up with in 1786, when he was nine years old [73] (see also Euler [92, part 1, §415]):

$$\begin{array}{rcccccccccccc}
S_n & = & 1 & + & 2 & + & 3 & + \cdots + & (n-1) & + & n \\
+\,S_n & = & n & + & (n-1) & + & (n-2) & + \cdots + & 2 & + & 1 \\
\hline
2S_n & = & (n+1) & + & (n+1) & + & (n+1) & + \cdots + & (n+1) & + & (n+1)
\end{array}$$

$$S_n = \frac{n(n+1)}{2}, \qquad \text{for } n \geqslant 0.$$

Lines in the Plane

OK, we have our solution:

$$L_n = \frac{n(n+1)}{2} + 1, \qquad \text{for } n \geqslant 0.$$

The key induction step is:

$$L(n) = L_n = L_{n-1} + n$$

we evaluate now its first few terms:

$$L_0 = 1,\ L_1 = 2,\ L_2 = 4,\ L_3 = 7,\ L_4 = 11,\ L_5 = 16,$$

and so on....

PART THREE: The Josephus Problem

Flavius Josephus - historian of 1st century

Josephus Story: During Jewish-Roman war he was among 41 Jewish rebels captured by the Romans. They preferred suicide to the capture and decided to form a circle and to kill every third person until no one was left.

Josephus with with one friend wanted none of this suicide nonsense.

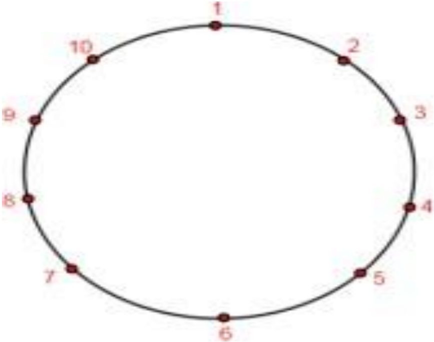
He calculated where he and his friend should stand to avoid being killed and they were saved.

The Josephus Problem - Our variation

n people around the circle

We **eliminate** each second remaining person until one survives. We denote by $J(n)$ the **position** of a surviver

Example $n = 10$



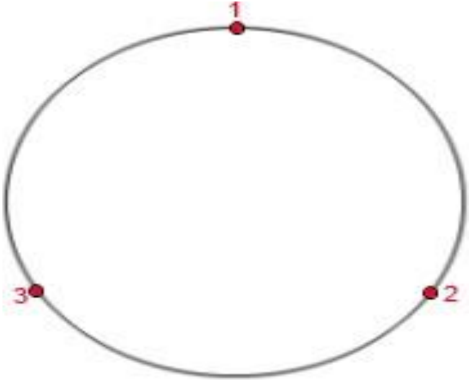
Elimination order: 2, 4, 6, 8, 10, 3, 7, 1, 9. As a result, number 5 survives, i.e. $J(n) = 5$

Problem: Determine survivor number J(n)

We know that J(n) = 5

We **evaluate** now J(n) for n=1,2,6

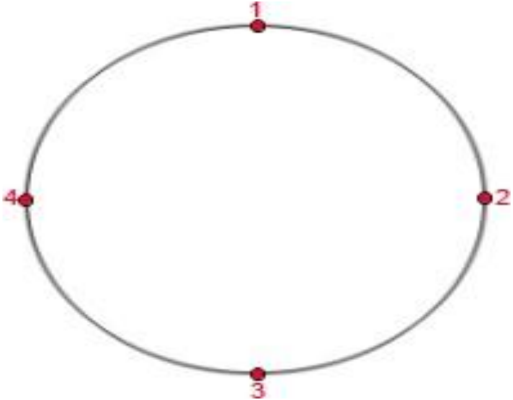
J(1)=1, J(2) = 1, J(3):



We get that J(3)=3

Determine survivor number J(n)

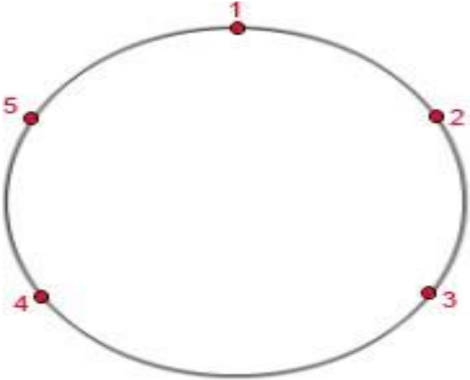
Picture for J(4):



We get J(4)=1

Determine survivor number J(n)

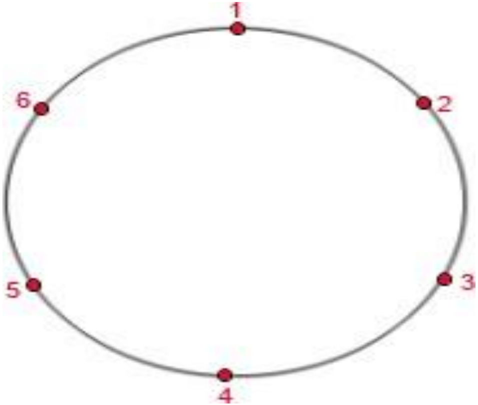
Picture for J(5):



We get J(5)=3

Problem: Determine survivor number J(n)

Picture for J(6):



We get J(6)=5

Determine survivor number J(n)

We put our results in a table:

n	1	2	3	4	5	6
J(n)	1	1	3	1	3	5

Observation

All our J(n) after the first run are odd numbers

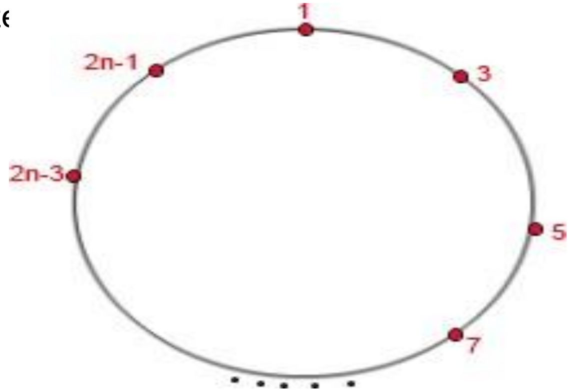
Fact

First trip eliminates all even numbers

Determine survivor number J(n)

ASSUME that we START with 2n people

Afte



3 goes out next

This is like starting with n except **each** person has been doubled and decreased by 1

Determine survivor number $J(n)$

Case $n=2n$

We get **$J(2n)=2J(n)-1$** (each person has been doubled and decreased by 1)

We know that $J(10)=5$, so $J(20) = 2J(10)-1 = 2*5-1 = 9$

$J(20)=J(2*10)= 2J(10)-1 = 2*5-1=10-1=9$

$J(20)=9$

$J(50)= 37$

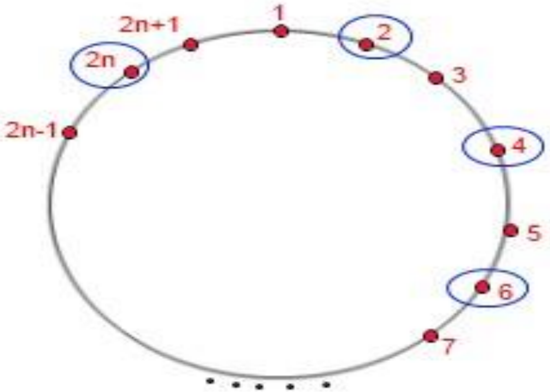
$J(30)=2J(15)-1= 2*15-1=29$

Determine survivor number J(n)

Case n=2n+1

ASSUME that we start with 2n+1 people:

First looks like that

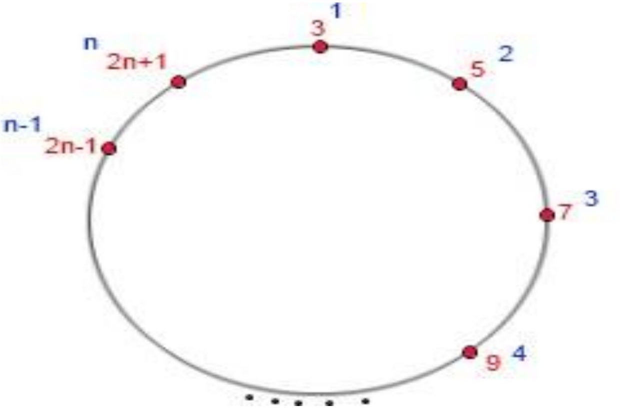


1 is wipped out after 2n

We want to have n-elements after **first** round

Determine survivor number J(n)

After the first trip we have



This is like starting with n except that now each person is doubled and increased by 1

Determine survivor number $J(n)$

CASE $n=2n + 1$

$$J(2n+1)=2J(n)+1$$

Recurrence Formula for J(n)

The Recurrence Formula RF for J(n) is:

$$J(1) = 1$$

$$J(2n) = 2J(n) - 1$$

$$J(2n+1) = 2J(n) + 1$$

Remember that J(n) is a position of the survivor

J(21)=J(20+1)=2J(10)+1=2*5+1=11

From Recursive Formula to Closed Form Formula

In order to find a **Closed Form Formula** (CF) equivalent to given **Recursive Formula** RF we ALWAYS follow the the Steps 1 - 4 listed below.

Step 1 Compute from recurrence RF a TABLE for some initial values. In our case RF is:

$$J(1) = 1, \;\; J(2n) = 2J(n) - 1, \; J(2n+1) = 2J(n) + 1$$

Step 2 **Look** for a pattern formed by the values in the TABLE

Step 3 **Find** - guess a closed form formula CF for the pattern

Step 4 **Prove** by Mathematical Induction that RF = CF

TABLE FOR J(n)

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
J(n)	1	1	3	1	3	5	7	1	3	5	7	9	11	13	15	1
	G1	G2		G3				G4								G5

Observation: $J(n) = 1$ for $n = 2^k$, $k = 0, 1, ..$

Next step: we form groups of J(n) for n consecutive powers of 2 and observe that

J(n)	G1	G2	G3	G4	G5	...
n	2^0	$2^1 + l$	$2^2 + l$	$2^3 + l$	$2^4 + l$	...

for $0 \leq l < 2^{(k-1)}$ and $k = 1, 2, ...5,$

Computation of J(n)

Observe that for each group G_k the corresponding ns are
$n = 2^{k-1} + l$ for all $0 \le l < 2^{(k-1)}$, k=1, 2, ...

and the value of $J(n)$ for $n = 2^k + l$ i.e. $J(n) = J(2^k + l)$, k=0,1..

increases by 2 within the group

J(n)	1	3	5	7 = 2l+1
n	2^2	$2^2 + l$	$2^2 + 2$	$2^2 + 3$
	l=0	l=1	l=2	l=3

l=0 l=1 l=2 l=3

Guess for CF formula for J(n)

Given $n = 2^{k-1} + l$ we **observed** that $J(n) = 2l + 1$

We **guess** that our CF formula is

$$J(2^k + l) = 2l + 1,$$

for any $k \geq 0$, $0 \leq l < 2^k$

Representation of *n*

$n = 2^k + l$ is called a representation of n when

l is a **remainder** by dividing n by 2^k and

k is the largest power of 2 not exceeding n

*J(50) = J(2^5+18=l)= 2l+1=2*18+1=36+1=37*

*J(50)= 2J(25)-1=2*19-1=37*

Proof RF = CF

RF: $J(1) = 1, J(2n) = 2J(n) - 1, J(2n+1) = 2J(n) + 1$

CF: $J(2^k + l) = 2l + 1$, for $n = 2^k + l$, $k \geq 0, 0 \leq l < 2^k$

Proof: by Mathematical Induction on k

Base Case: k=0.

and l = 0.

$J(2^0) = 1$

$n = 1$.

We evaluate $J(1) = 1$,

Some Facts

Fact 1 $\forall_m J(2^m) = 1$

Proof by induction over m

Observe that $2^m \in Even$, so we use the formula $J(2n) = 2J(n) - 1$, and get

$$J(2^m) = J(2 * 2^{m-1}) =^{def} 2J(2^{m-1}) - 1 =^{ind} 2 * 1 - 1 = 1$$

Hence we also have

Fact 2 First person will always survive whenever n is a power of 2

General Case

Fact 3 Let $n = 2^m + l$. Observe that the number of people is reduced to power of 2 after there have been l executions.

The first remaining person, the survivor is number $2l + 1$

Our solution

$$J(2^m + l) = 2l + l$$

where $n = 2^m + l$ and $0 \leq l < 2^m$ depends heavily on powers of 2

The Josephus Problem small numbers

Evaluate $J(n)$ for small n:

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	$\cdots$
$J(n)$	1	1	3	1	3	5	7	1	3	5	7	9	11	13	15	1	$\cdots$

1. $J(n)$ is always odd;
2. Recurrence equation:

$$J(1) = 1\ ;$$
$$J(2n) = 2J(n) - 1 \quad \text{for } n >= 1\ ;$$
$$J(2n+1) = 2J(n) + 1 \quad \text{for } n >= 1\ .$$

n	1	2	3	4	5	6
J(n)	1	1	3	1	3	5

The Josephus Problem small numbers

Evaluate $J(n)$ for small n:

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	$\cdots$
$J(n)$	1	1	3	1	3	5	7	1	3	5	7	9	11	13	15	1	$\cdots$

For example

$$
\begin{aligned}
J(86) &= 2J(43) - 1 = 45 \\
J(43) &= 2J(21) + 1 = 23 \\
J(21) &= 2J(10) + 1 = 11 \\
J(10) &= 5
\end{aligned}
$$

The Josephus Problem small numbers

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$J(n)$	1	1	3	1	3	5	7	1	3	5	7	9	11	13	15	1

Voilà! It seems we can group by powers of 2 (marked by vertical lines in the table); $J(n)$ is always 1 at the beginning of a group and it increases by 2 within a group. So if we write n in the form $n = 2^m + l$, where 2^m is the largest power of 2 not exceeding n and where l is what's left, the solution to our recurrence seems to be

$$J(2^m + l) = 2l + 1, \qquad \text{for } m \geqslant 0 \text{ and } 0 \leqslant l < 2^m. \tag{1.9}$$

(Notice that if $2^m \leqslant n < 2^{m+1}$, the remainder $l = n - 2^m$ satisfies $0 \leqslant l < 2^{m+1} - 2^m = 2^m$.)

We must now prove (1.9). As in the past we use induction, but this time the induction is on m. When $m = 0$ we must have $l = 0$; thus the basis of

The Josephus Problem small numbers

(1.9) reduces to $J(1) = 1$, which is true. The induction step has two parts, depending on whether l is even or odd. If $m > 0$ and $2^m + l = 2n$, then l is even and

$$J(2^m + l) \;=\; 2J(2^{m-1} + l/2) - 1 \;=\; 2(2l/2 + 1) - 1 \;=\; 2l + 1\,,$$

by (1.8) and the induction hypothesis; this is exactly what we want. A similar proof works in the odd case, when $2^m + l = 2n + 1$. We might also note that (1.8) implies the relation

$$J(2n+1) - J(2n) \;=\; 2.$$

Either way, the induction is complete and (1.9) is established.

To illustrate solution (1.9), let's compute $J(100)$. In this case we have $100 = 2^6 + 36$, so $J(100) = 2 \cdot 36 + 1 = 73$.

The Josephus Problem small numbers

Closed formula can be used for computing function $J(n)$:

Example

We have $1030 = 2^{10} + 6$, so $J(1030) = 2 \cdot 6 + 1 = 13$.

Recap:

We studied recurrence function in a way to solve various problems

Reference: Concrete mathematics by Ronald, Knuth