Advanced Counting Techniques
8.1 Applications of Recurrence Relations

Credit

Ming-Hsuan Yang, Cheng-Chia Chen,

Siti Zanariah Satari, McGraw-Hill,

Kirack Sohn, Marc Pomplun, Yun Peng,

Husni Al-Muhtaseb

8.1 Recurrence relations

Many counting problems can be solved with recurrence relations

Example: The number of bacteria doubles every hour. If a colony begins with 5 bacteria, how many will be present in n hours?

- Let $a_n = 2a_{n-1}$ where n is a positive integer with $a_0 = 5$

Recurrence relations

A **recurrence relation** for the sequence $\{a_n\}$ is an equation that expresses in terms of one or more of the previous terms of the sequence,

- i.e., $a_0, a_1, \ldots, a_{n-1}$, for all integers n with $n \geq n_0$ where n_0 is a nonnegative integer

A sequence is called a **solution** of a recurrence relation if its terms satisfy the recurrence relation

Recursion and recurrence

A recursive algorithm provides the solution of a problem of size n in terms of the solutions of one or more instances of the same problem of smaller size.

When we analyze the complexity of a recursive algorithm, we obtain a recurrence relation that expresses the number of operations required to solve a problem of size n in terms of the number of operations required to solve the problem for one or more instance of smaller size

Example

Let $\{a_n\}$ be a sequence that satisfies the recurrence relation $a_n = a_{n-1} - a_{n-2}$ for $n = 2, 3, 4, \ldots$ and suppose that $a_0 = 3$ and $a_1 = 5$,

what are a_2 and a_3?

Using the recurrence relation,

$a_2 = a_1 - a_0 = 5 - 3 = 2$ and

$a_3 = a_2 - a_1 = 2 - 5 = -3$

Example

Determine whether the sequence $\{a_n\}$, where $a_n = 3n$ for every nonnegative integer n, is a solution of the recurrence relation

$$a_n = 2a_{n-1} - a_{n-2} \text{ for } n = 2, 3, 4, \ldots$$

Suppose $a_n = 3n$ for every $n \geq 0$.

Then for $n \geq 2$, we have

$$2a_{n-1} - a_{n-2} = 2(3(n-1)) - 3(n-2)$$

$$= 6n - 6 - 3n + 6 = 3n = a_n.$$

Thus, $\{a_n\}$ where $a_n = 3n$ is a solution for the recurrence relation

Modeling with recurrence relations

Compound interest: Suppose that a person deposits \$10,000 in a savings account at a bank yielding 11% per year with interest compounded annually. How much will it be in the account after 30 years?

Let P_n denote the amount in the account after n years. The amount after n years equals the amount in the account after n - 1 years plus interest for the n-th year, we see the sequence $\{P_n\}$ has the recurrence relation

$$P_n = P_{n-1} + 0.11P_{n-1} = (1.11)P_{n-1}$$

Modeling with recurrence relations

The initial condition $P_0 = 10{,}000$, thus

$P_1 = (1.11)P_0$

$P_2 = (1.11)P_1 = (1.11)\ (1.11)P_0 = (1.11)^2 P_0$

$P_3 = (1.11)P_2 = (1.11)\ (1.11)^2 P_0 = (1.11)^3 P_0$

...

$P_n = (1.11)P_{n-1} = (1.11)^n P_0$

We can use mathematical induction to establish its validity

Modeling with recurrence relations

Assume $P_n = (1.11)^n\, 10{,}000$.

$$\begin{aligned} P_{n+1} &= (1.11)P_n \\ &= (1.11)(1.11)^n\, 10{,}000 \\ &= (1.11)^{n+1}\, 10{,}000 \end{aligned}$$

$n = 30$, $P_{30} = (1.11)^{30}\, 10{,}000$

$= 228{,}922.97$

Towers of Hanoi

- Game description
  - Start with three pegs numbered 1, 2 and 3 mounted on a board,
  - *n* disks of different sizes with holes in their centers,
  - placed in order of increasing size from top to bottom.
- Objectives of the game
  - Find the minimum number of moves needed to have all *n* disks stacked in the same order in peg number 3.

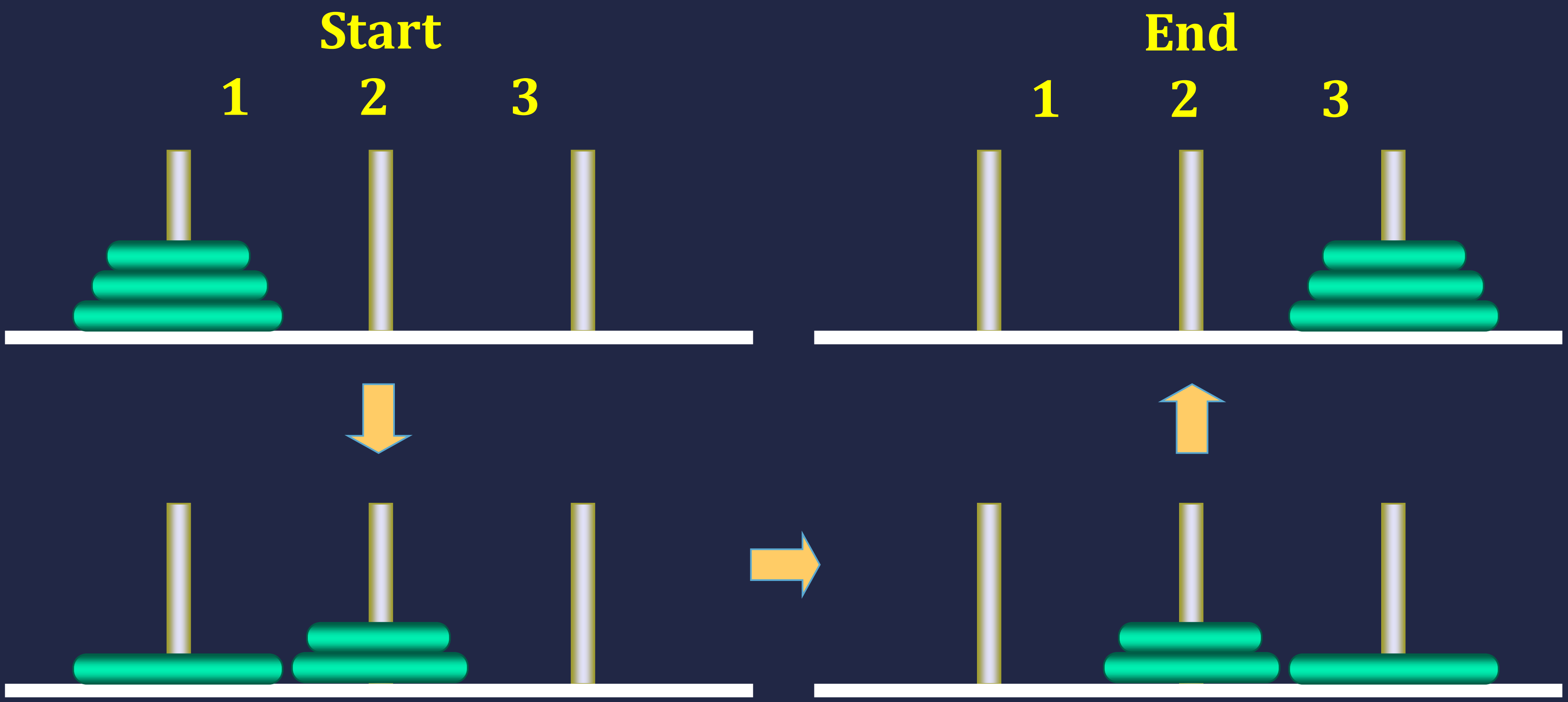
Rules of the game: Hanoi towers

Start with all disks stacked in peg 1 with the smallest at the top and the largest at the bottom

- Use peg number 2 for intermediate steps
- Only a disk of smaller diameter can be placed on top of another disk

End of game: Hanoi towers

Game ends when all disks are stacked in peg number 3 in the same order they were stored at the start in peg number 1.



Example: Hanoi towers

Number of disks = 3

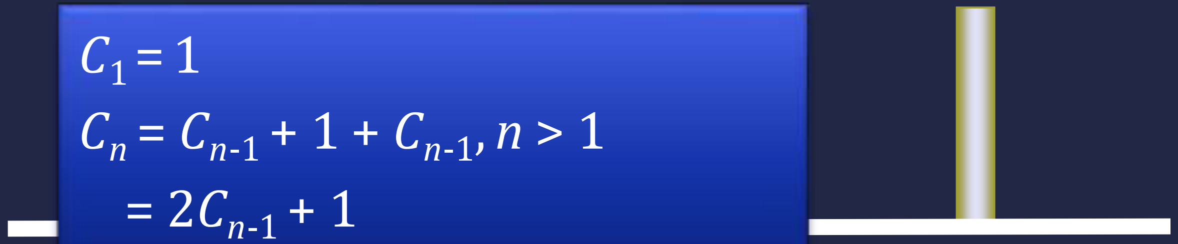
1. $[d_1, d_2] : p_1 \rightarrow p_2$ (using p_3)
    1. $d_1 : p_1 \rightarrow p_3$
    2. $d_2 : p_1 \rightarrow p_2$
    3. $d_1 : p_3 \rightarrow p_2$
2. $d_3 : p_1 \rightarrow p_3$
3. $[d_1, d_2] : p_2 \rightarrow p_3$ (using p_1)
    1. $d_1 : p_2 \rightarrow p_1$
    2. $d_2 : p_2 \rightarrow p_3$
    3. $d_1 : p_1 \rightarrow p_3$

$C_1 = 1$

$C_n = C_{n-1} + 1 + C_{n-1}, n > 1$

$= 2C_{n-1} + 1$

$= 2^n - 1$



Number of moves with n disks

Disks	Moves
3	7
4	15
5	31
6	63
7	127
15	32767
26	67108863
35	3.44 E+10
64	1.84 E+19
84	1.93 E+25
85	3.87 E+25
95	3.96 E+28

Example: Bit Strings

Find a recurrence relation and give initial conditions for the number of bit strings of length n that do not have two consecutive 0s.

Solution:

Let a_n : number of bit strings of length n that do not have two consecutive 0s. Then;

a_n = (number of bit strings of length n - 1 that do not have two consecutive 0s)
+ (number of bit strings of length n - 2 that do not have two consecutive 0s)

$$a_n = a_{n-1} + a_{n-2} \; ; \qquad n \geq 3$$

Have 3 minutes to grasp

With initial condition;

$a_1 = 2$, both strings of length 1 do not have consecutive 0s (0 & 1)

$a_2 = 3$, the valid strings only 01, 10 and 11

Details in next slides

Modeling with Recurrence Relations

Question:

Let a_n denote the number of bit strings of length n that do not have two consecutive 0s ("valid strings"). Find a recurrence relation and give initial conditions for the sequence $\{a_n\}$.

Solution:

Idea: The number of valid strings equals the number of valid strings ending with a 0 plus the number of valid strings ending with a 1. (Sum Rule)

$$\boxed{1}\ \boxed{2}\ \boxed{\ldots}\ \boxed{n-3}\ \boxed{n-2}\ \boxed{n-1}\ \boxed{n}$$

Any bit string of length n-1 with no two consecutive 0s

$$\underbrace{\boxed{1}\ \boxed{2}\ \boxed{\ldots}\ \boxed{n-3}\ \boxed{n-2}\ \boxed{n-1}}_{a_{n-1}}\ \boxed{\mathbf{1}}$$

End with a 1

Any bit string of length n-2 with no two consecutive 0s

$$\underbrace{\boxed{1}\ \boxed{2}\ \boxed{\ldots}\ \boxed{n-3}\ \boxed{n-2}}_{a_{n-2}}\ \boxed{\mathbf{1}}\ \boxed{\mathbf{0}}$$

End with a 0

$$\therefore a_n = a_{n-1} + a_{n-2},\quad n \geq 3$$

$$a_1 = 2 \text{ (strings : 0,1)}$$

$$a_2 = 3 \text{ (strings : 01,10,11)}$$

$$\therefore a_3 = a_2 + a_1 = 5,\ a_4 = 8,\ a_5 = 13$$

Modeling with Recurrence Relations

Let us assume that $n \geq 3$, so that the string contains at least 3 bits.

Let us further assume that we know the number $a_{n\text{-}1}$ of valid strings of length $(n – 1)$.

Then how many valid strings of length n are there, if the string ends with a 1?

There are $a_{n\text{-}1}$ such strings, namely the set of valid strings of length $(n – 1)$ with a 1 appended to them.

Note: Whenever we append a 1 to a valid string, that string remains valid.

Modeling with Recurrence Relations

Now we need to know: How many valid strings of length n are there, if the string ends with a 0?

Valid strings of length n ending with a 0 must have a 1 as their $(n - 1)^{st}$ bit (otherwise they would end with 00 and would not be valid).

And what is the number of valid strings of length $(n - 1)$ that end with a 1?

We already know that there are a_{n-1} strings of length n that end with a 1.

Therefore, there are a_{n-2} strings of length $(n - 1)$ that end with a 1.

Modeling with Recurrence Relations

So there are a_{n-2} valid strings of length n that end with a 0 (all valid strings of length $(n - 2)$ with 10 appended to them).

The number of valid strings is the number of valid strings ending with a 0 plus the number of valid strings ending with a 1.

That gives us the following **recurrence relation**:

$$a_n = a_{n-1} + a_{n-2}$$

Modeling with Recurrence Relations

What are the **initial conditions**?

$a_1 = 2$ (0 and 1)

$a_2 = 3$ (01, 10, and 11)

Some values:

$a_3 = a_2 + a_1 = 3 + 2 = 5$

$a_4 = a_3 + a_2 = 5 + 3 = 8$

$a_5 = a_4 + a_3 = 8 + 5 = 13$

…

This sequence satisfies the same recurrence relation as the **Fibonacci sequence**.

Since $a_1 = f_3$ and $a_2 = f_4$, we have $a_n = f_{n+2}$.