Middlebury

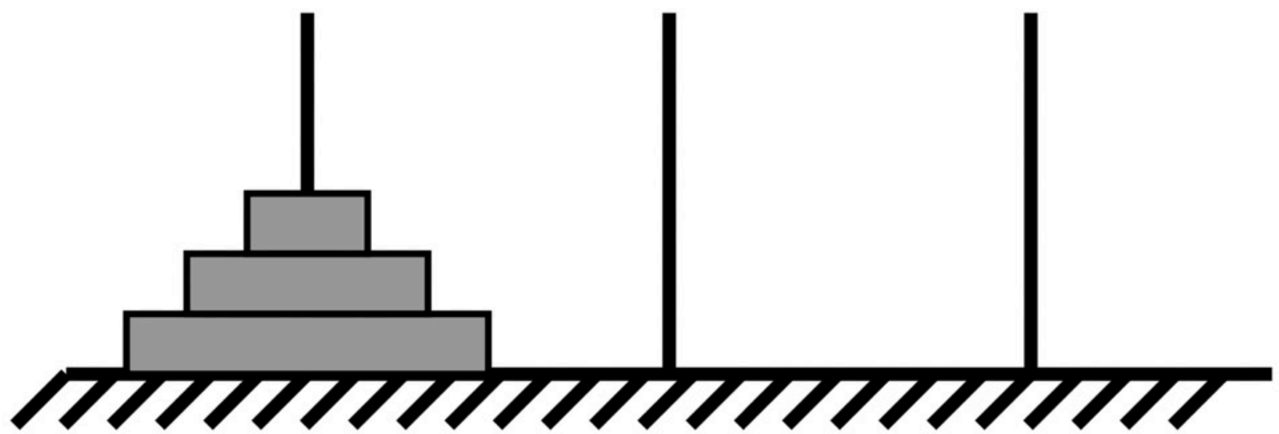
CSCI 200: Math Foundations of Computing

Spring 2026

Lecture 4M: Recurrence Relations I

Goals for today:

- Develop recurrence relations to analyze the performance of recursive algorithms.
- Solve recurrence relations using the *expand and pray* method.
- Solve homogeneous linear recurrence relations.

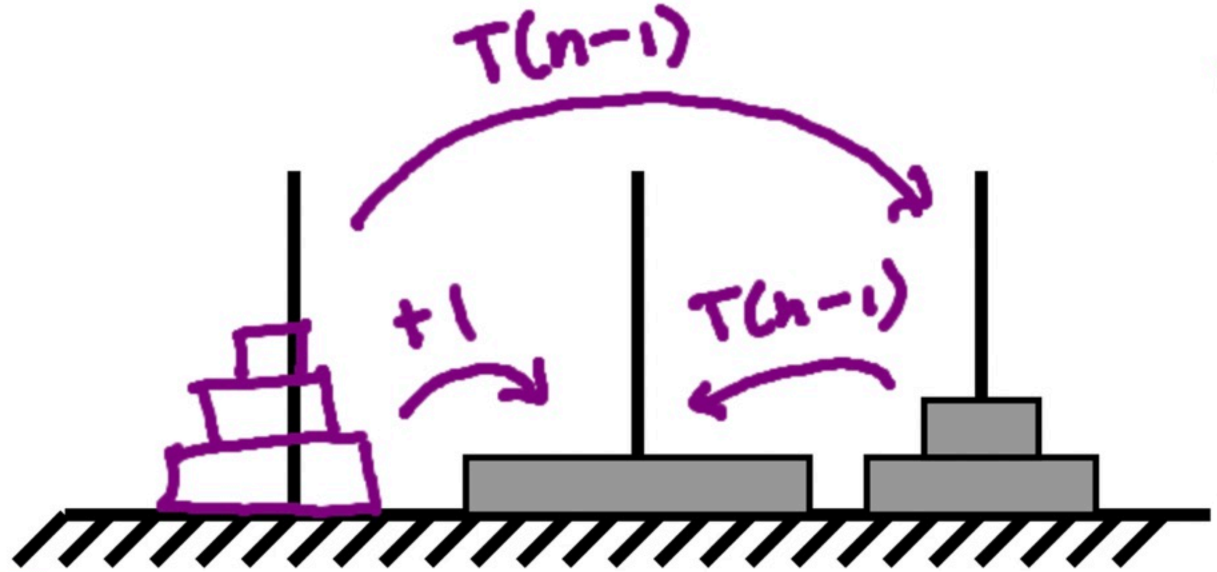


	# moves	
$n=1$	1	
$n=2$	3	
$n=3$	7	
$n=4$	17	15
	?	

Rules:

1. Move one disk at a time.
2. Every move displaces a disk from the top of one stack to the top of another (or empty rod).
3. No larger disk can ever be placed on top of a smaller one.

Counting the number of moves in the Tower of Hanoi problem.



let $T(n)$ be # moves to displace a stack of n disks

$$\boxed{T(n) = 2T(n-1) + 1}$$

recurrence relation

base case $T(1) = 1$

How to get a closed-form expression? Let's try expanding the recurrence relation, and hope we see a pattern.

$\square = n-1$

$T(\boxed{n}) = 2T(\boxed{n}-1) + 1$ → plug in $T(\boxed{n-1}) = 1 + 2T(n-2)$

$= 1 + 2T(n-1)$

$= 1 + 2(1 + 2T(n-2))$

$= 1 + 2 + 2^2 T(n-2)$ → $T(n-2) = 1 + 2T(n-3)$

$= 1 + 2 + 2^2 [1 + 2T(n-3)]$

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

$= 1 + 2 + 2^2 + \dots + 2^i \overbrace{T(n-i)}^{\text{eventually going to reach } \boxed{T(1) = 1}}$

when is $n - i = 1$? ($n=1$)

last term $i = n-1$

geometric series

$= \sum_{i=0}^{\boxed{n-1}} 2^i = \frac{1 - r^{\square + 1}}{1 - r} = \frac{1 - 2^n}{1-2} = \boxed{2^n - 1}$

$r = 2$

Types of recurrence relations we will see.

```
def fib(n):
    if n <= 1:
      return n
    return fib(n - 1) + fib(n - 2)
```

linear

$$f(n) = f(n-1) + f(n-2)$$

$$f(0) = 0$$

$$f(1) = 1$$

```
def binary_search(lst, val, lo, hi):
    if lo > hi:
        return -1
    m = (lo + hi) // 2
    if lst[m] == val:
        return m
    elif lst[m] > val:
        return binary_search(lst, val, lo, m - 1)
    else:
        return binary_search(lst, val, m + 1, hi)
```

divide-and-conquer

Developing recurrence relations from recursive programs.

```python
def fib(n):
    if n <= 1:
      return n
    return fib(n - 1) + fib(n - 2)
```

$$f(n) = f(n-1) + f(n-2)$$

$$f(0) = 0$$
$$f(1) = 1$$

homogeneous recurrence relation

Solving homogeneous linear recurrence relations: Determining a closed-form expression for Fibonacci numbers.

$f(0) = 0$
$f(1) = 1$

$$f(n) = f(n-1) + f(n-2)$$

guess: always r^n $\quad r \in \mathbb{R}$

$f(n) = r^n$
$f(n-1) = r^{n-1}$
$f(n-2) = r^{n-2}$

$$r^n = r^{n-1} + r^{n-2}$$

$$r^n - r^{n-1} - r^{n-2} = 0$$

$$\underbrace{r^{n-2}}_{\neq 0}\underbrace{\left[\overset{(1)}{\underset{a}{r^2}} - \underset{b}{r} - \underset{c}{1}\right]}_{=0} = 0$$

$$r = \frac{1 \pm \sqrt{1+4}}{2}$$

$$r_1 = \frac{1+\sqrt{5}}{2}, \quad r_2 = \frac{1-\sqrt{5}}{2}$$

thm: both solutions $\rightarrow$ $f_1(n) = r_1^n$, $f_2(n) = r_2^n$

then $c_1 f_1(n) + c_2 f_2(n)$ is also a solution.
$c_1, c_2 \in \mathbb{R}$

$$f(n) = c_1\left(\frac{1+\sqrt{5}}{2}\right)^n + c_2\left(\frac{1-\sqrt{5}}{2}\right)^n$$

Solving homogeneous linear recurrence relations:
Determining a closed-form expression for Fibonacci numbers.

$$f(n) = c_1\left(\frac{1+\sqrt{5}}{2}\right)^n + c_2\left(\frac{1-\sqrt{5}}{2}\right)^n \quad \text{with } f(0)=0,\ f(1)=1$$

$$f(0) = 0 = c_1 + c_2$$

$$f(1) = 1 = c_1\left(\frac{1+\sqrt{5}}{2}\right) + c_2\left(\frac{1-\sqrt{5}}{2}\right)$$

$$c_1 = \frac{1}{\sqrt{5}} \quad c_2 = \frac{-1}{\sqrt{5}}$$

0, 1, 1, 2, 3, 5, 8, ...

$$\boxed{f(n) = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n}$$

What if we have this recurrence relation?

$$f(n) = 4f(n-1) - 4f(n-2), \quad \text{with} \quad f(0) = 1, \quad \text{and} \quad f(1) = 0$$

What are the roots of the $\underbrace{\text{characteristic equation}}_{r^2 + \cdots}$ for this recurrence?

A. $r_1 = 2, r_1 = -2$

B. $r_1 = 2, r_1 = 4$

C. $r_1 = 2, r_2 = 2$

D. $r_1 = 2, r_1 = 1$

$r^n = 4r^{n-1} - 4r^{n-2}$

$r^n - 4r^{n-1} + 4r^{n-2} = 0$

$\underbrace{r^{n-2}}_{\neq 0} \underbrace{[r^2 - 4r + 4]}_{=0} = 0$

$(r-2)^2 = 0$

$r_1 = 2$

$r_2 = 2$

thm: repeated roots

if $f(n) = r^n$ with r is a repeated root then $f(n) = n r^n$ is also a solution.

$f(n) = 2^n$

$f(n) = c_1 2^n + c_2 \cdot n \cdot 2^n$

$f(0) = \boxed{1 = c_1}$

$f(1) = 0 = 2 + c_2(1)(2) \rightarrow c_2 = -1$

$f(n) = 2^n - n 2^n = \boxed{(1-n)2^n}$