

# Recursion

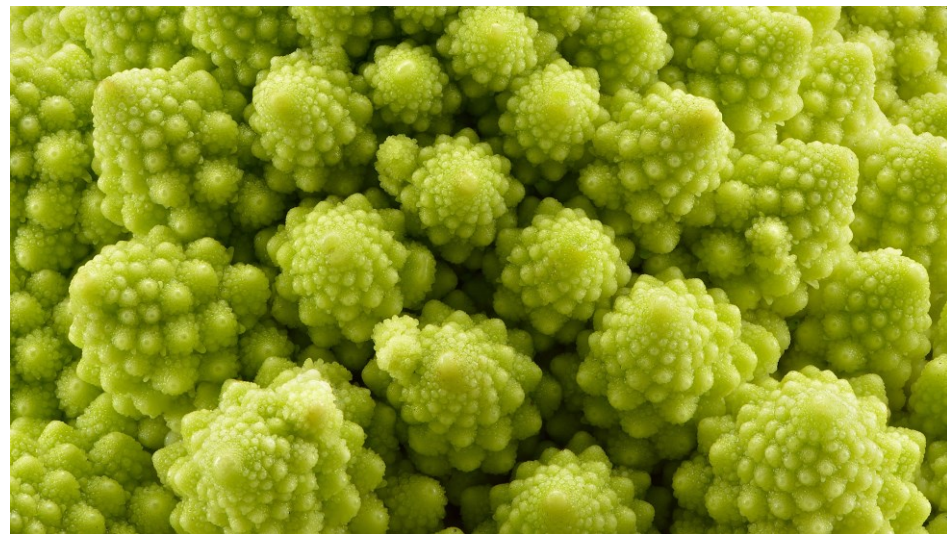
## Recursive Algorithms

C1

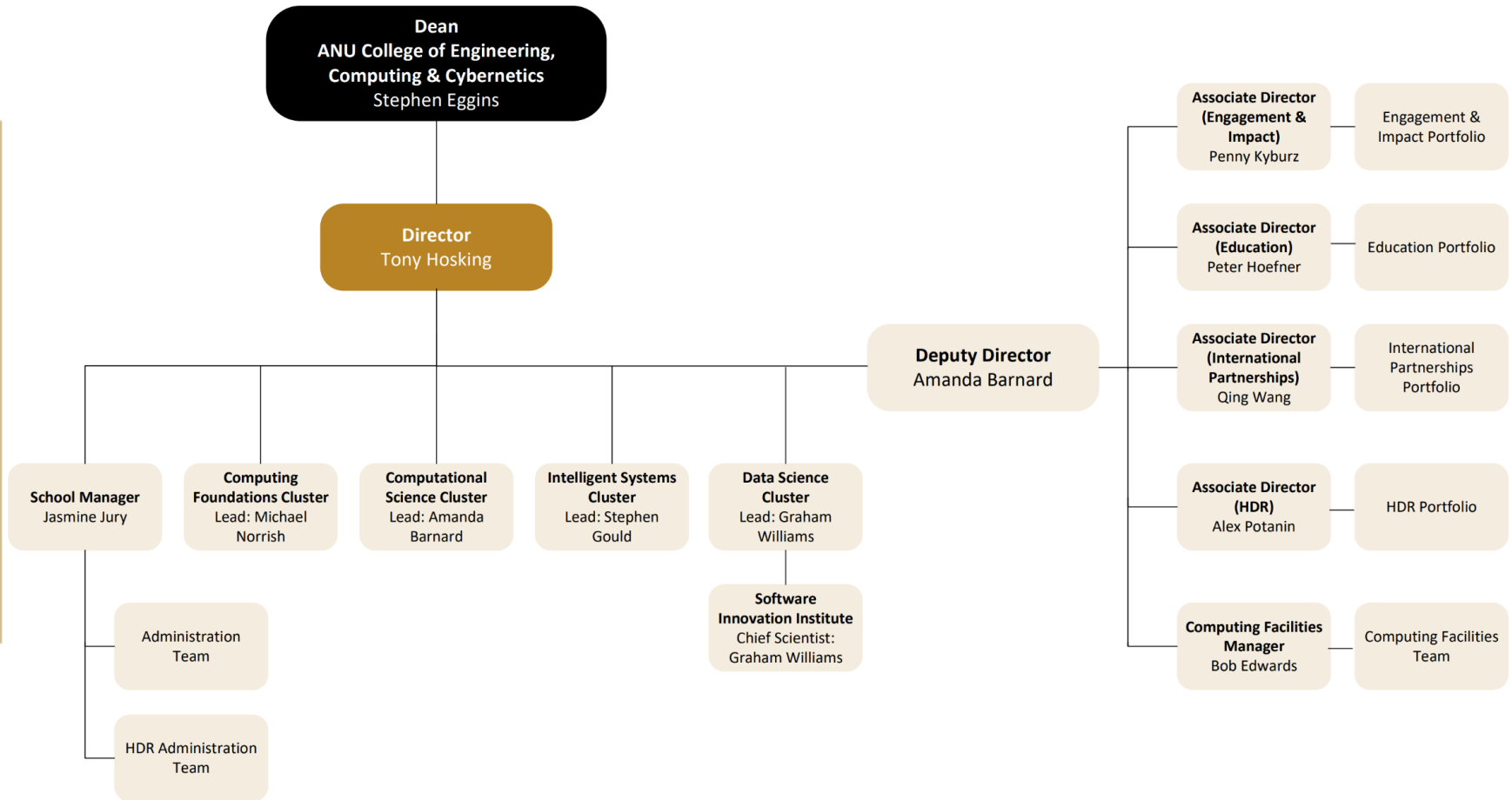




Alf van Beem, CC0, via Wikimedia Commons



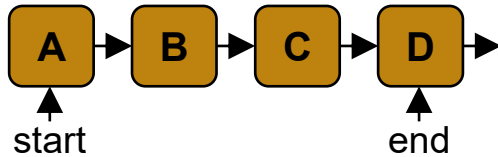
Ivar Leidus - Own work, CC BY-SA 4.0, via Wikimedia Commons



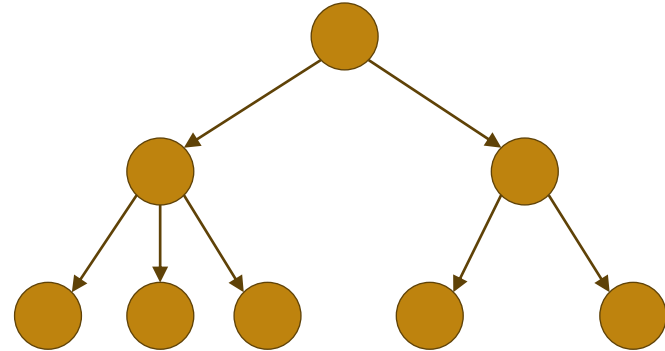
# Recursive Data Structures

A recursive data structure is comprised of components that reference other components of the same type.

- One or more base components
- One or more composite components



Linked List



Tree

# Recursive Algorithms

A recursive algorithm references itself.

It is comprised of:

- One or more base cases
- One or more recursive cases that reduce towards the base cases

# Example: Fibonacci Sequence

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, ...

$$\text{fib}(0) = 0$$

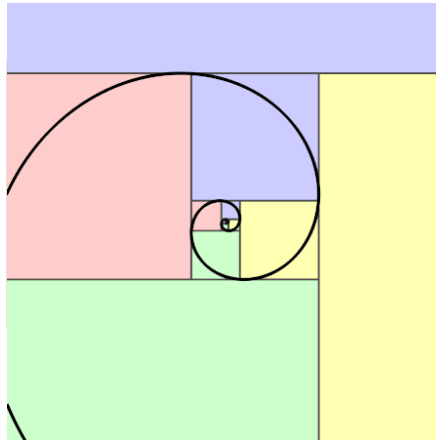
(base case)

$$\text{fib}(1) = 1$$

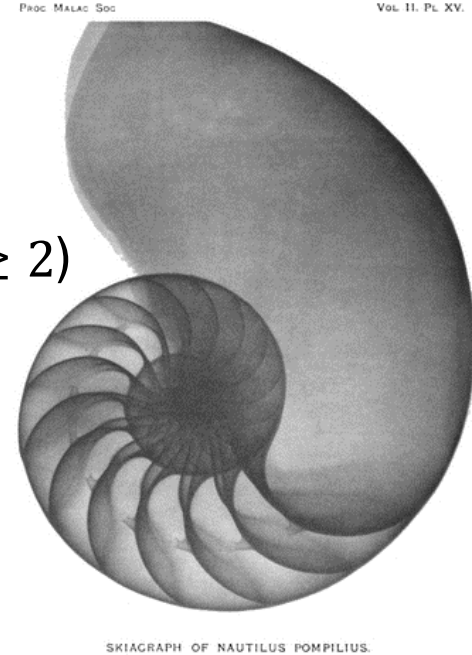
(base case)

$$\text{fib}(n) = \text{fib}(n-1) + \text{fib}(n-2)$$

(recursive case for  $n \geq 2$ )



Jahobr, CCO, via Wikimedia Commons



James Green, Public domain, via Wikimedia Commons

# Example: Mergesort (von Neumann, 1945)

## Sort a list

- List of size 0 (base case)
  - Already sorted
- List of size 1 (base case)
  - Already sorted
- List of size  $> 1$  (recursive case)
  - Split into two sub-lists
  - Sort each sub-list (recursion)
  - Merge the two sorted sublists into one sorted list (by iteratively picking the lower of the two least elements)

