# Recursion

Sunday, February 4, 2018 2:32 PM

## Recursion

- Divide (same sub-problems) and Conquer (base case + recursive call)
    - Be careful of infinite recursion (must be able to reach base case)
- Can be visualised with stacks
- Before recursive call = FIFO, after call = LIFO
- Comparison with iterative solution? Efficiency?
    - Recursion = slower but easier to write
- Head Recursion vs. Tail Recursion (recursive call is last function to perform)

## Application

- Countdown, gcd, base conversion, factorial
- Linked Lists: printing (forwards/ backwards), insertion into sorted LL
- Tower of Hanoi (vs. iterative?)
    - Recursive = O(2^n)
- nCk, binary search, find kth smallest #, fibonacci #, permutations
- Duplicated calls (e.g. fib #)

Slide 37: repeated work
Slide 39: B
Slide 44: 8 queens problem

## Key Concepts

- Backtracking
- Systematic and exhaustive search
- Future: search spaces

```
String.subString(i);
String.charAt();
```