Admin: HW: Don't forget to tell Gradescope your teammates!
HW1 is graded — regrade requests due in 2 weeks
HW2 swap two to three or quack to steque
HW3 out tonight due next Monday
No more late registration

**Concatenation for degrees with random access**

- Doubly-linked lists
- In an array list with resizing

**HEAD** **ACHE** $\xrightarrow{\text{concat}}$ **HEAD ACHE**

**IMPOSSIBLE** **KINGS MEN**

**Concat(AB):**

- if $A$.num < $B$.num
  - while $A$ is not empty
    - $B$.push($A$.pull($i$))
  - else
    - while $B$ is not empty
      - $A$.shove($B$.pop($i$))

**Total # items in all deques**

$\Theta(n)$ time worst case

Start with $n$ degrees of size 1 and perform $n-1$ concatenations

How long?
- Crude: $\Theta(n^2)$
- Balanced: $\Theta(n \log n)$

$[\log n] = \# \text{times we divide by 2 to go from } n \text{ to } \leq 1$

$log_b n = \frac{\log_n}{\log_b}$

$lgn = \log_2 n$
Every level: \( \text{concat} \left( \frac{n}{2k+1} \right) \text{deque pairs} \)

- each length \( 2^k \)
- each takes \( 2 \cdot 2^k \) indels

\[ \# \text{indels} = n \]

Arbitrary: \( O(\log n) \) \[ \text{Best!} \]

Every item: count transfers

- Length of deque containing \( x \)
- \( \geq \)doubles every time \( x \) transfers

\[ a \leq b \Rightarrow a \leq \frac{3a+b}{2} \]

\[ \Rightarrow \text{Each item transfers } O(\log n) \text{ times} \]

- \( n \) items
- \( \Rightarrow O(\log n) \text{ time in total} \)

Am. time for each Concat = \( O(\log n) \)

+ Insert

- Insert \( \Rightarrow \text{Concat (New (n))} \)
- pays for all future transfers

- Insert: \( O(\log n) \) am. time
- Concat: Zero am. time
- Delete: \( O(1) \) am. time
Tombstone:

Sorted array

- Find(x):
- Delete(x):

Naively $\Theta(n)$-time

Trick: Leave $x$ there, mark it "dead"

when half items dead, rebuild

\[0 \times 2 3 \times 8 15 21 \times 30\]