Big-O Terminology

"An algorithm is $O(\text{something})$"

"worst case" $\exists k \geq 0, c \geq 1. \; \forall n \geq k. \;$ algorithm $\leq c \cdot$ something

✓ Bubble sort is $O(n^2)$

✓ Bubble sort is $O(n^3)$

Randomized Quick Sort

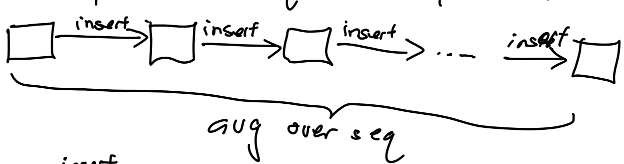
$O(n^2)$

$O(n \log n)$

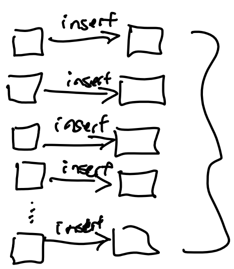
_______ worst case : absolute worst situation ; universe out to get you

expected worst case : bad input, but universe normal ; dice work, not loaded

amortized worst case : ultimately, after a sequence of operations, each op cost X.



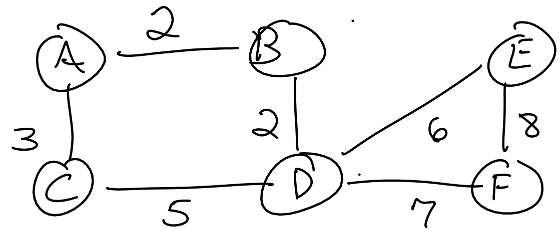
average worst case :



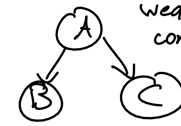
average over all cases

Hash Table operations are <u>amortized average $O(1)$</u>

Minimum Spanning Trees



Tree is a graph which is weakly connected and has $|E| = |V| - 1$



MST is a subgraph of a graph which

1. Has all of the vertices
2. Is connected
3. Is a tree

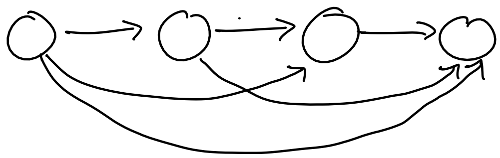
} spanning tree

4. Has total weight $\leq$ all other spanning trees

BFS

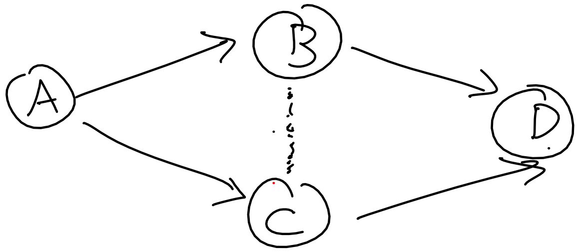
$|E| \leq |V|^2 - |V|$

$\boxed{|E| \text{ is } O(|V|^2)}$



n^2 is $O(n^3)$!!

Topological Sort

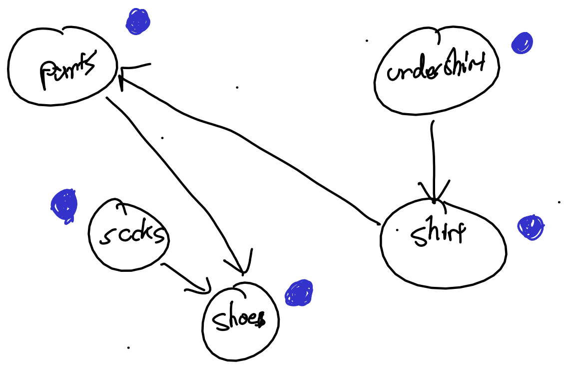


A, B, C, D

A, C, B, D

C, B, A, D ✗

Produce an ordering of vertices in a graph such that each vertex is visited before any vertex it points to.



[undershirt, shirt, socks, pants, shoes]

$O(V+E) =$ $\qquad O(V+V)$

$O(\max(V,E))$ $\qquad = O(V)$

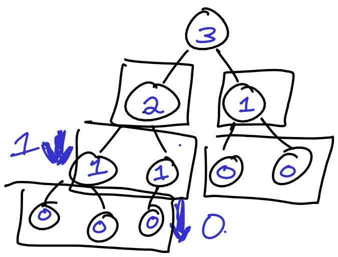
Heapify is $O(n)$.

insert (heap) : amortized $O(\log n)$

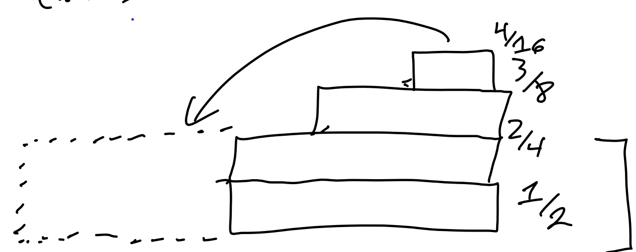
For each of n things:
  heap.insert(O, n) $\Big\}\ O(n \log n)$

h ← heapify(n things) $\Big\}\ O(n)$

"most"
"Almost all bubble downs are small."



Function Heapify:
  For each node from end to beginning:
    bubbleDown(node)



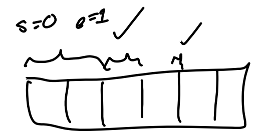
$$\frac{0}{2}n + \frac{1}{4}n + \frac{2}{8}n + \frac{3}{16}n + \cdots$$

$$\sum_{i=1}^{n} \frac{i-1}{2^i} n = n \sum_{i=1}^{n} \frac{i-1}{2^i} \leq n \boxed{\sum_{i=1}^{\infty} \frac{i-1}{2^i}} \leq 2$$

Quick Sort

0 n-1

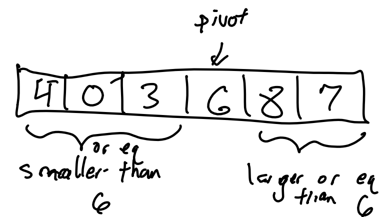
```
Function QS ( array , start , end )
    If start ≥ end : Return
    pivot ← Partition(array, start, end)
    QS ( array , start, pivot - 1 )
    QS( array , pivot + 1, end )
End Function
Function Partition ( array, start, end )
    pivotVal ← array [end]
    frontier ← start
    fence ← start
    While frontier < end :
        frontier ← frontier + 1
        If array [frontier - 1] < pivotVal:
            swap( array [fence], array [frontier - 1] )
            fence ← fence + 1
        End If
    End While
    swap (array [end], array [fence])
    Return fence
End Function
```

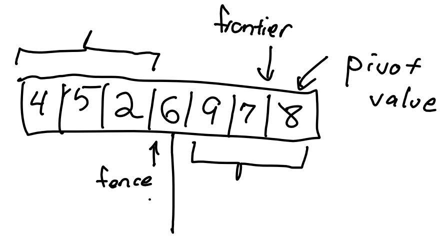


pref O(n)

Partition : given start and end,
return pivot
start ≤ pivot ≤ end
all things to left of pivot index
are smaller than pivot value

- start .. fence : small
- fence ... frontier : large





ADTs

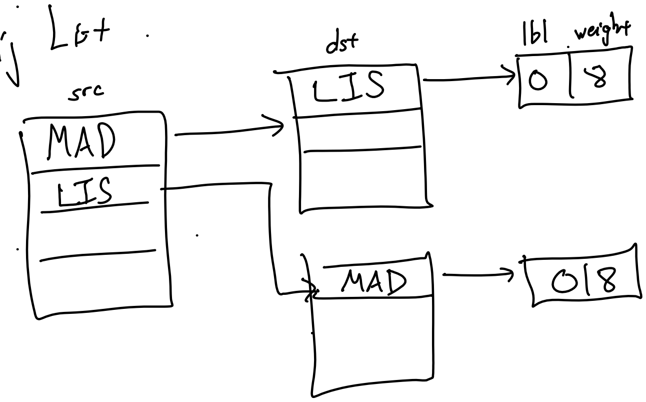
- Dictionary —— Linear, BSTs, AVL trees, Hash Table
- Stack —— LinkedStack, ArrayStack
- List —— LinkedList, ArrayList
- Queue —— LinkedQueue, ArrayQueue
- Graph —— Adjacency List, Adjacency Matrix
- Priority Queue —— Heap

Hash Table good in what way? average amortized O(1) :)
bad? memory hungry :(
need good hash :(
always a set of keys for which hash table is slow

AVL tree good? never under any circumstances worse than O(log n)

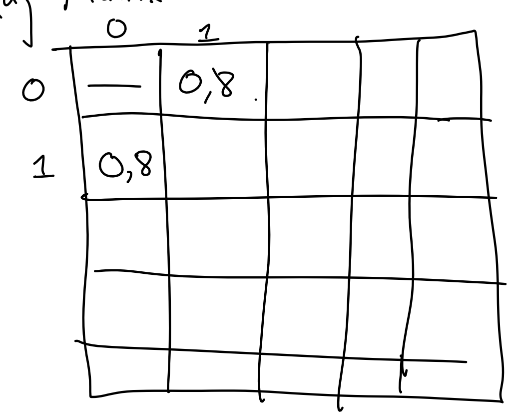
Linear —— small values of n; simple

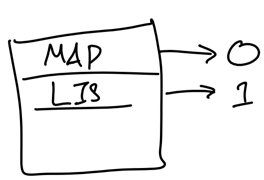
Adj List



- Adding vertices is easy
- Uses less memory

Adj Matrix





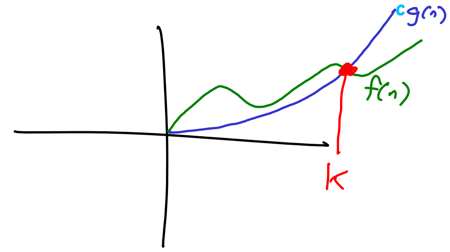
- Cohesive (all in one place)

$f(n)$ is $O(g(n))$

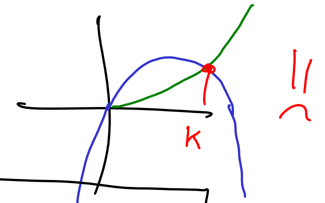
$$\equiv$$

you pick $\longrightarrow$ $\underbrace{\exists c \geq 1,\ k \geq 0.\ \forall n \geq k.}$ $f(n) \leq c\,g(n)$

I pick :)



n^2+n is $O(n^2)$



Let $c=2,\ k=5$

Do not do this.

$$n^2+n \leq 2n^2$$
$$5^2+5 \leq 2\cdot 5^2$$
$$30 \leq 50 \checkmark$$

$$cx \leq cy \qquad c \geq 0$$
$$\Updownarrow$$
$$x \leq y$$
$$\Downarrow$$
$$x-a \leq y-a$$

destructive

$$n^2+n \leq 2n^2$$
$$n \leq n^2$$
$$1 \leq n$$

$$\exists k \geq 0,\ c \geq 1.\ \forall n \geq k.\ n^2+n \leq cn^2$$

$$\forall n \geq 1.\ n^2+n \leq 2n^2$$

$$\boxed{\forall n \geq 1.\ 1 \leq 1}$$

$$\boxed{\forall n \geq 1.\ 1 \leq n}$$

$$\forall n \geq 1.\ 1 \leq n^2$$

$$\forall n \geq 1.\ n \leq n^2$$

$$\forall n \geq 1.\ n^2 \leq n^2$$

$$\forall n \geq 1.\ n^2+n \leq n^2+n^2$$

$$\forall n \geq 1.\ n^2+n \leq 2n^2$$

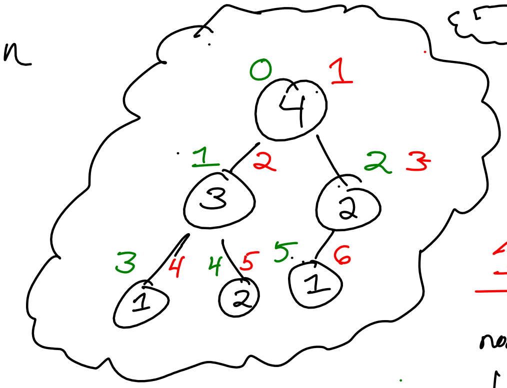
$$\exists c \geq 1.\ \forall n \geq 1.\ n^2+n \leq cn^2$$

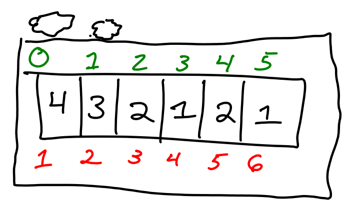
$$\exists k \geq 0.\ c \geq 1.\ \forall n \geq k.\ n^2+n \leq cn^2$$

$$\equiv$$

n^2+n is $O(n^2)$

Bubble Down





1-based land

node : n

left : $2n$

right : $2n+1$

parent : $\lfloor n/2 \rfloor$

0-based land

node : n

left : $2(n+1)-1$

$2n+2-1$

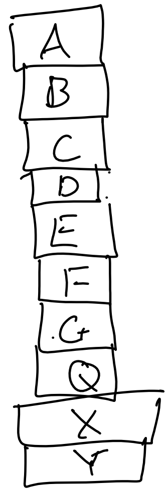
$2n+1$

right : $2(n+1)+1 \quad -1$

$2(n+1)$

$2n+2$

parent : $\lfloor \frac{n+1}{2} \rfloor - 1$


A
B
C
D
E
F
G
Q
X
Y