

G

T

G

T

5

E

5

N

O

N

O

O

S

O

G

T

G

T

5

E

5

N

O

N

O

O

S

O

G

T

G

T

5

E

5

E

S

E

S

E

S Q U E U E S

T

- Linear data Structures

- Flexible Size

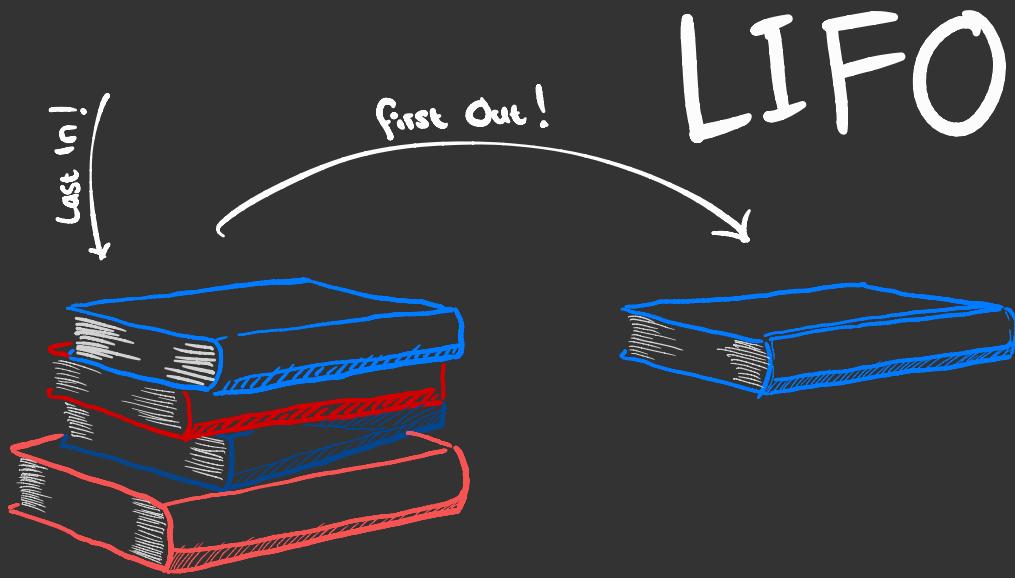
- The Main different is the way data is removed

A

C

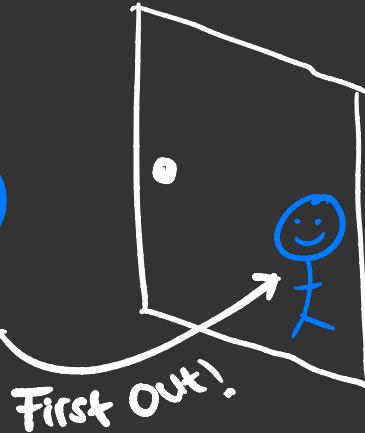
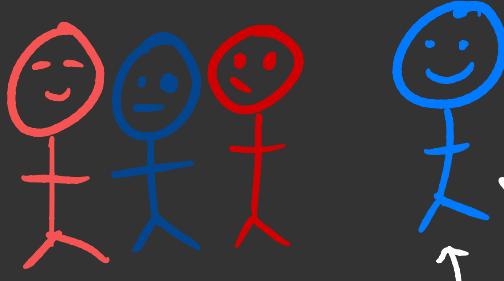
K

S



LIFO

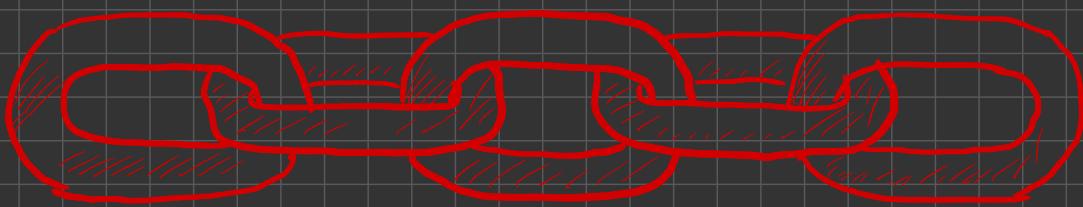
FIFO



First in!

First Out!

LINKED



LIST

STRINGS



} Singly

CHARACTERS



NUMBERS

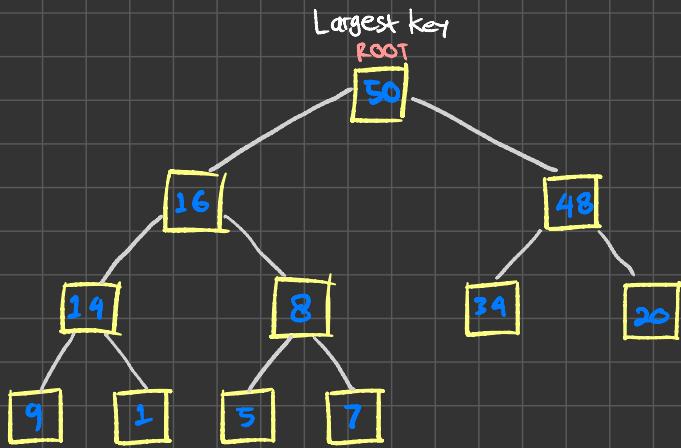


— Doubly

The Elements
can be:

- SORTED
- UNSORTED
- DUPLICATES
- UNIQUES

MIN or MAX HEAP

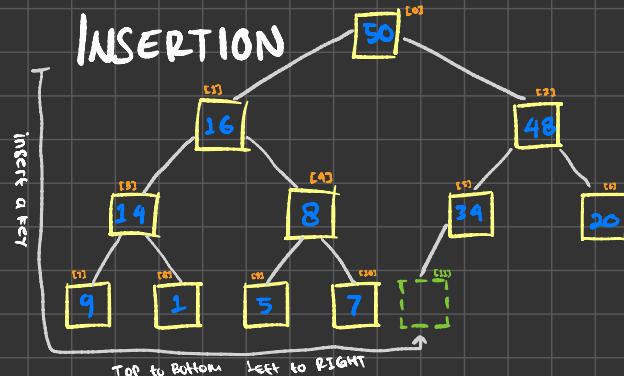
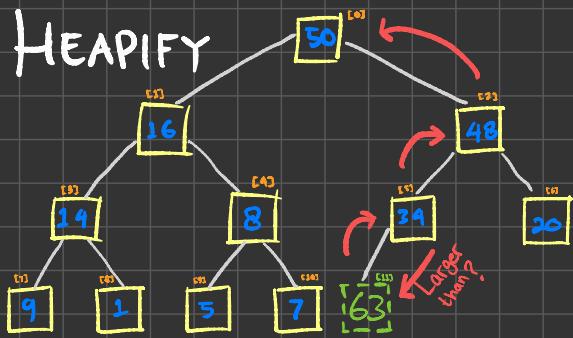


Parent index $\lceil \frac{1}{2} \rceil$ $\times 2 + 1 = \lceil \frac{3}{2} \rceil$

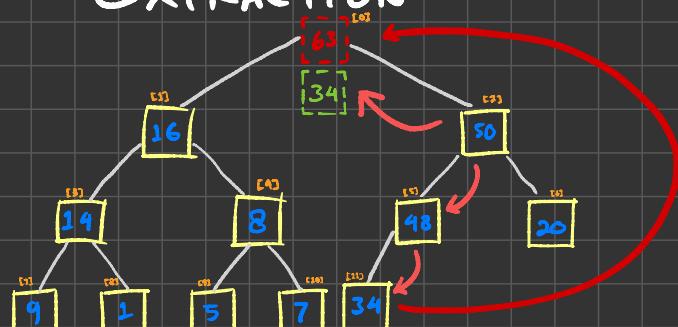
Left Child Index $\lceil \frac{1}{2} \rceil$

Parent index $\lceil \frac{1}{2} \rceil$ $\times 2 + 2 = \lceil \frac{4}{2} \rceil$

Right child index $\lceil \frac{1}{2} \rceil$

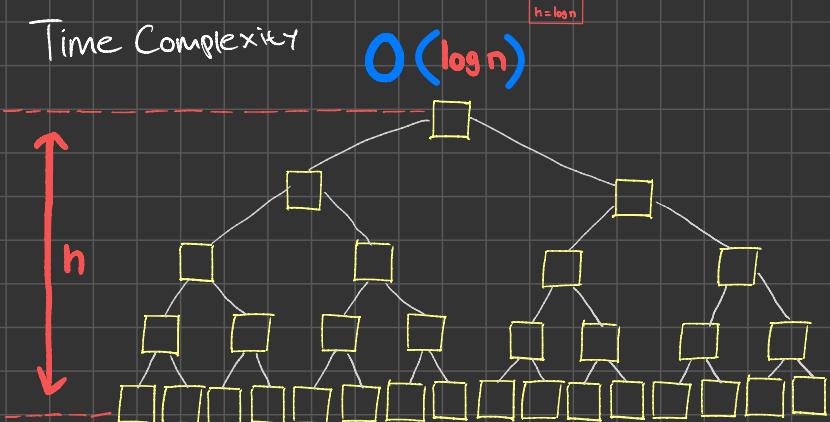


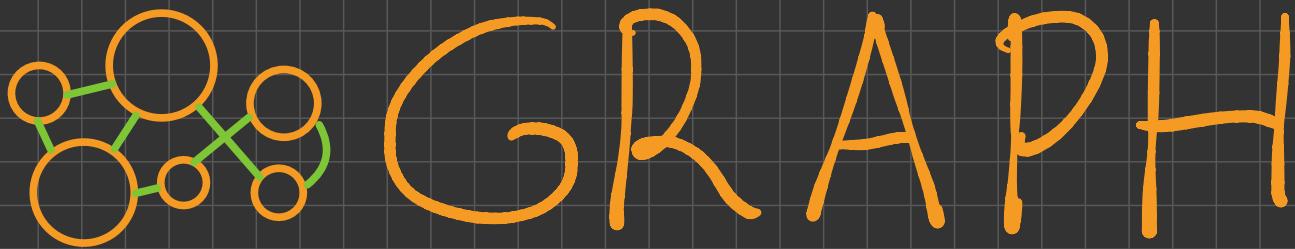
EXTRACTION



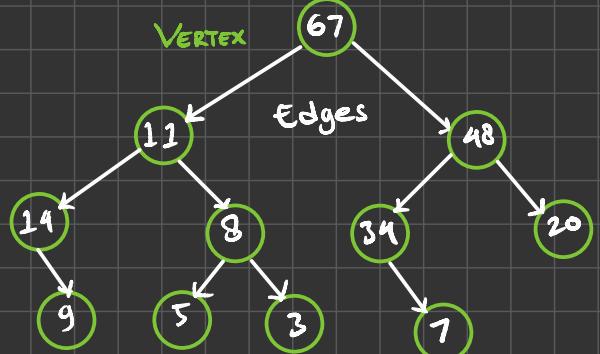
Time Complexity

$O(\log n)$

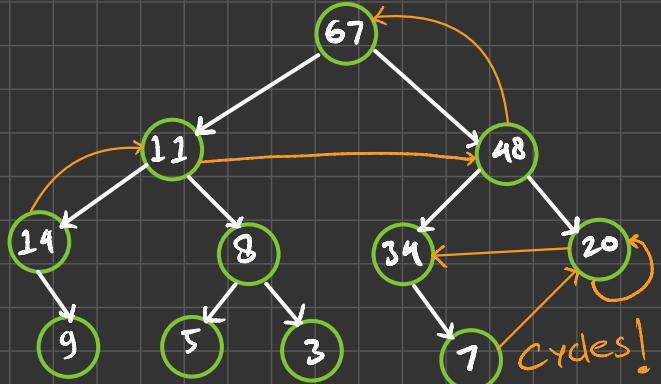




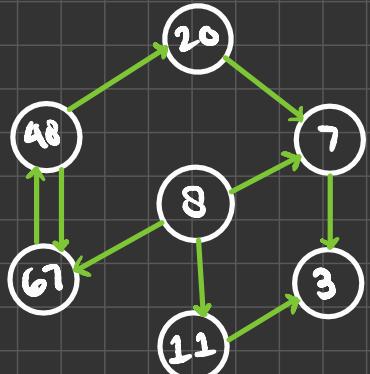
DENSE
When the graph has many edges



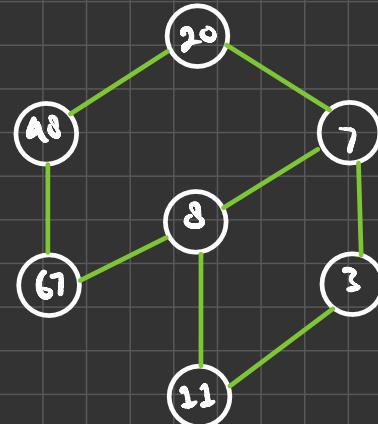
SPARSE
When it has relatively less edges



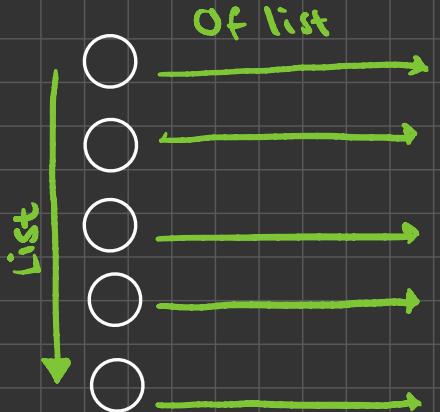
DIRECTED GRAPH



UNDIRECTED GRAPH

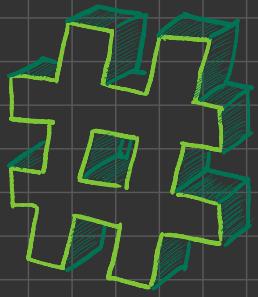


ADJACENCY LIST

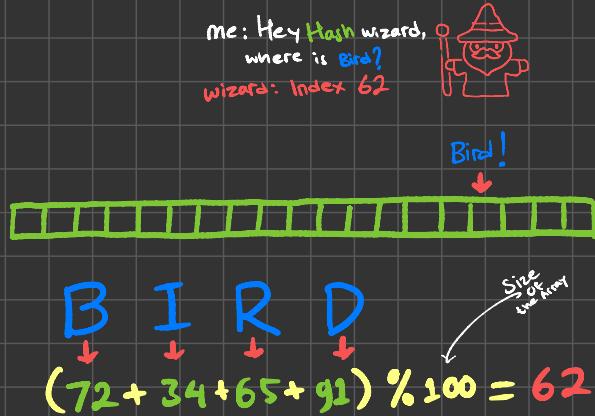


ADJACENCY MATRIX

		TO				
		1	2	3	4	5
FROM	1	1				
	2	1				
3	1	(4)			1	1
4		(3)		1		(7)
5			1		1	



HASH



COLLISION HANDLING

OPEN ADDRESSING



SEPARATE CHAINING

Storing multiple names in one index, how?

by using LINKED LIST

