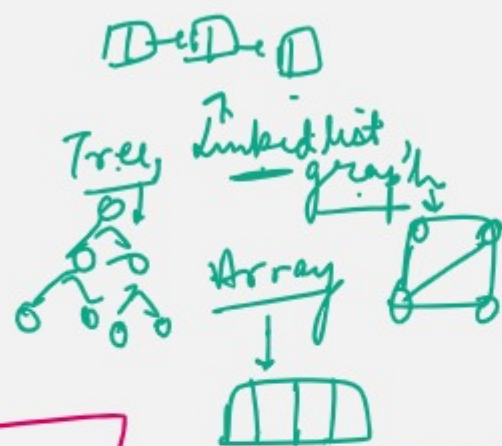


Basic Search Algo

- Linear, Binary
- Hashmap (DS)
- key: value
- using objects & map()



ternary Search (sorted array)

2	4	6	8	10	12	20	30	32
0	1	2	3	4	5	6	7	8

target = 12

$\log_3(n)$

→ Find 2 mid value
 $mid1 = l + (r-l)/3$
 $= 0 + (9)/3$
 $= 3$

$mid2 = r - (r-l)/3$
 $= 9 - (9)/3$
 $= 9 - 3$
 $= 6$

→ $(target == arr[mid1])$ target == arr[mid1]
 $target < arr[mid1]$ target < arr[mid1]
 $target > arr[mid1]$ target > arr[mid1]

→

8	10	12	20
4	5	6	7

 $r = mid1 - 1$
 $l = mid1 + 1$ or $mid1 + 1$

$mid1 = l + (r-l)/3$
 $= 4 + (1)/3$
 $= 4 + 0.33$
 $= 4$

$mid2 = r - (r-l)/3$
 $= 5 - (1)/3$
 $= 5 - 0.33$
 $= 5$

$(target == arr[mid1])$ $(target == arr[mid2])$
 return mid1 return mid2

Exponential Search (sorted)

2	21	24	30	40	50	60	80	90	100	105	109
0	1	2	3	4	5	6	7	8	9	10	11

target = 105

$arr[0] == target$
 $i = 1$ $arr[1] > target$
 $i = 2$ $arr[2] > target$
 $i = 4$ $arr[4] > target$
 $i = 8$ $arr[8] > target$

Complement Binary search

90	100	105	109
----	-----	-----	-----

Jump Search (sorted)

2	3	4	6	10	15	20
0	1	2	3	4	5	6

Linear

→ $start = 0$, $end = \sqrt{n}$, 'n' is length of array
 $= \sqrt{49}$
 $= 7$

$(target > arr[end])$ $\rightarrow start = end + 1$

$(target == arr[end])$ $\rightarrow end = end - 1$

6	10	15	20
3	4	5	6

$start = 3$, $end = \sqrt{n}$
 $= 2 + 2 = 4$

Linear Search

$(target == arr[start])$ return end

→ Linear Search in the final chunk of array

$O(\sqrt{n})$