

Searching algorithm

17/01/25

Binary search

- What is an binary search?
- What is searching?

Searching is the process of finding an item with specific properties from a collection of items. The items may be stored as records in any databases - data elements in arrays, etc.

- Why do we need searching?

→ Searching is a core computer science algorithm since its days computers store a lot of information. To retrieve this information proficiently we need very efficient, searching algorithm.

Linear search

unordered linear search ordered/sorted linear search.

Binary search

→ Searching algorithm in a limited search space.

→ Binary search is applicable in a sorted array. When the data are sorted.

↳ → sorted array.

→ Finding a word in a dictionary.

$$mn = O(n)$$

$$\max = \log(n)$$

= algorithm for binary search.

```
int 'Bin Search(A; n, key)
```

$l = 1 \quad h = n$

while ($l \leq h$) {

mid = $(l+h)/2$;

if (key == A[mid]) {

return mid;

~~else~~ if (key < A[mid]) {

$h = mid - 1$;

else $l = mid + 1$;

}

return 0;

}

When lower value becomes higher or equal to higher then means the element is not found. So basically it should not be higher than the higher value.

→ To show binary search works.

Iterative method:

1	2	5	6	8	9	11	14	16	18	19
1	2	3	4	5	6	7	8	9	10	11

Key = 8.

This is for linear search.

So key = 8 means we need to find that element from the array / list given as one.

A	1	2	5	6	8	9	11	14	16	18	19
	1	2	3	4	5	6	7	8	9	10	11
							↓ mid.				

$$l = 1 \quad h = 11 \quad \text{mid} = \frac{(l+h)}{2}$$

This is for linear search -

$$= \frac{1+11}{2}$$

so other key = 8.

$$= 12/2$$

mid[A] (= key)

= 6

1	2	8	6	8
---	---	---	---	---

$$\text{low}^2 \text{new } \text{mid} = \frac{1+5}{2}$$

1	2	5	6	8
2	3	4	5	

$$= \frac{6}{2} = 3$$

mid[A] = key,

but key greater than
mid of A

80,	5	6	18
	3	4	5

$$\frac{3+5}{2} = 4$$

mid != key

~~mid less~~

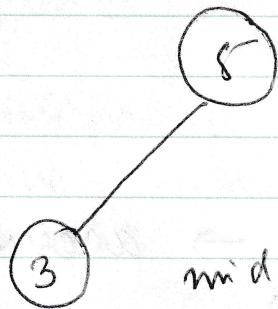
when low > high $\frac{5+8}{2} > 10 \Rightarrow$

both becomes - mid = key, element found -

Using a Tree -

1 2	3 4	5 6	7 8	9				
1	2	3	4	5	6	7	8	9

key = 3



l	h	$mid \leftarrow$
1	3	$10/2 =$
1	5	$8/2 =$

So searching time and steps depends on the height of the tree.