Tổng quan về thuật toán SEARCH - TÌM KIẾM 🚀
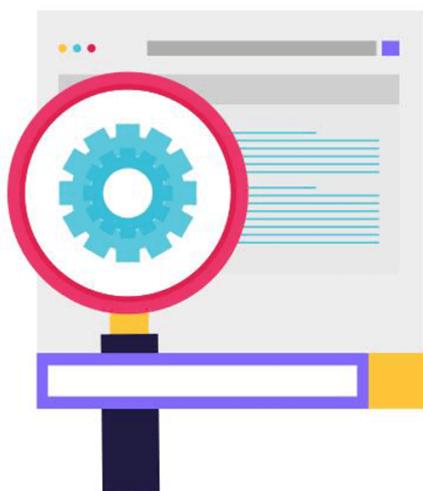
Source: https://www.volusion.com/blog/search-algorithms/

AGENDA

1. Linear search
2. Binary search

#	Linear Search	Binary Search
What	Search one by one in order	Reduce a half of search area by each step
Required sorted array	NO	**YES**
Big O	O(n)	**O(log(n))**

Visualize by https://www.cs.usfca.edu/~galles/visualization/Search.html

1. Linear search

```javascript
function findIndex(numberList, target) {
  if (!Array.isArray(numberList) || numberList.length === 0) return -1;

  for (let i = 0; i < numberList.length; i++) {
    const number = numberList[i];
    if (number === target) return i;
  }

  return -1;
}
```


Source: https://www.geeksforgeeks.org/searching-algorithms/
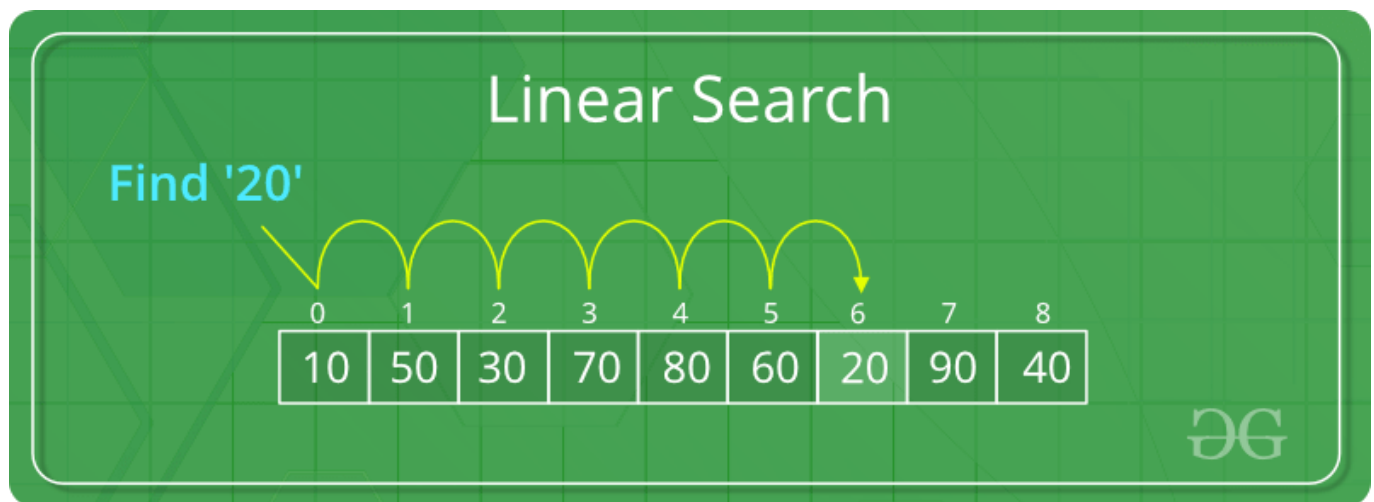
2. Binary search

```javascript
// assume: numberList is a sorted array
function binarySearch(numberList, target, left, right) {
  if (!Array.isArray(numberList) || numberList.length === 0) return -1;

  // base case / termination point (required for recursion)
  if (right < left) return -1;

  const mid = left + Math.trunc((right - left) / 2);
  if (numberList[mid] === target) return mid;

  // search on the right part if target is greater than mid
  if (target > numberList[mid]) {
    return binarySearch(numberList, target, mid + 1, right);
  }

  // otherwise, try to search on the left part
  return binarySearch(numberList, target, left, mid - 1);
}
```
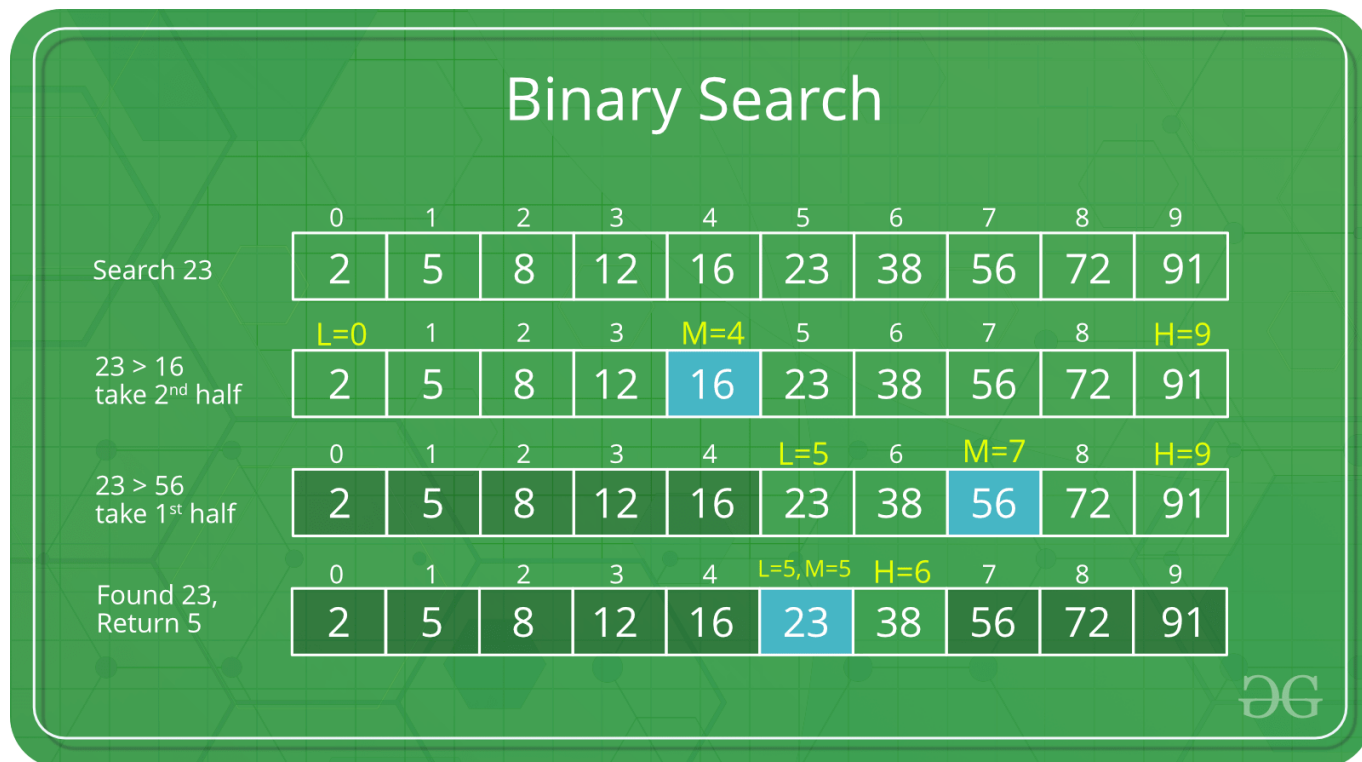
```javascript
// non-recursive
function binarySearch(numberList, target) {
  if (!Array.isArray(numberList) || numberList.length === 0) return -1;

  let left = 0;
  let right = numberList.length;

  while (left <= right) {
    const mid = left + Math.trunc((right - left) / 2);
    if (numberList[mid] === target) return mid;

    if (target > numberList[mid]) {
      left = mid + 1;
    } else {
      right = right - 1;
    }
  }

  return -1;
}
```

Binary Search

	0	1	2	3	4	5	6	7	8	9
Search 23	2	5	8	12	16	23	38	56	72	91

	L=0	1	2	3	M=4	5	6	7	8	H=9
23 > 16 take 2nd half	2	5	8	12	16	23	38	56	72	91

	0	1	2	3	4	L=5	6	M=7	8	H=9
23 > 56 take 1st half	2	5	8	12	16	23	38	56	72	91

	0	1	2	3	4	L=5, M=5	H=6	7	8	9
Found 23, Return 5	2	5	8	12	16	23	38	56	72	91

Source: https://www.geeksforgeeks.org/searching-algorithms/

Tham khảo

- https://www.cs.usfca.edu/~galles/visualization/Search.html
- https://www.geeksforgeeks.org/searching-algorithms/

Khoá học Javascript cho người mới bắt đầu 2021 🎉

- Tác giả: **Hậu Nguyễn** - Founder Easy Frontend
- Khoá học chỉ được published trên Udemy, không thông qua trung gian.
- Khoá học không bán dạng videos upload trên Google Drive hay bất cứ hình thức nào tương tự.
- Khoá học có nhóm discord để hỗ trợ trong quá trình học tập.

☎️ Liên hệ tác giả để được hỗ trợ:

- ✅ Facebook: https://www.facebook.com/nvhauesmn/
- ✅ Fanpage: https://www.facebook.com/learn.easyfrontend
- ✅ Youtube Channel: https://www.youtube.com/easyfrontend