Session - 6

Binary Search

andiyanalex1000@gmail.com

CONTENTS

pandiyanalex10000@gmail.com

01

Introduction To Search Algorithms

Introduction to Search

Linear Search:

Linear search (or sequential search) is the simplest searching algorithm.

It checks each element in the array one by one until the target is found or the end is reached.

How It Works:

- Start from the first element.
- Compare each element with the target.
- If match is found, return the index.
- If end is reached without finding, return -1.

Introduction to Search

Binary Search:

A searching algorithm to find the position of a target value within a sorted array.

Input array must be sorted

Works on: Arrays, Lists, Any random-access data structure

How It Works:

- Divide the search space in half
- Compare middle element with target
- Eliminate half of the data each time

02

Visual Representation

Example of Binary Search

arr = [2] [4] [6] [8] [10] [12] [14]

Target = 10

Step 1: mid = 8

arr = [2] [4] [6] [8] [10] [12] [14]

Target > mid (go right)

Step 2: mid = 12

arr = [10] [12] [14]

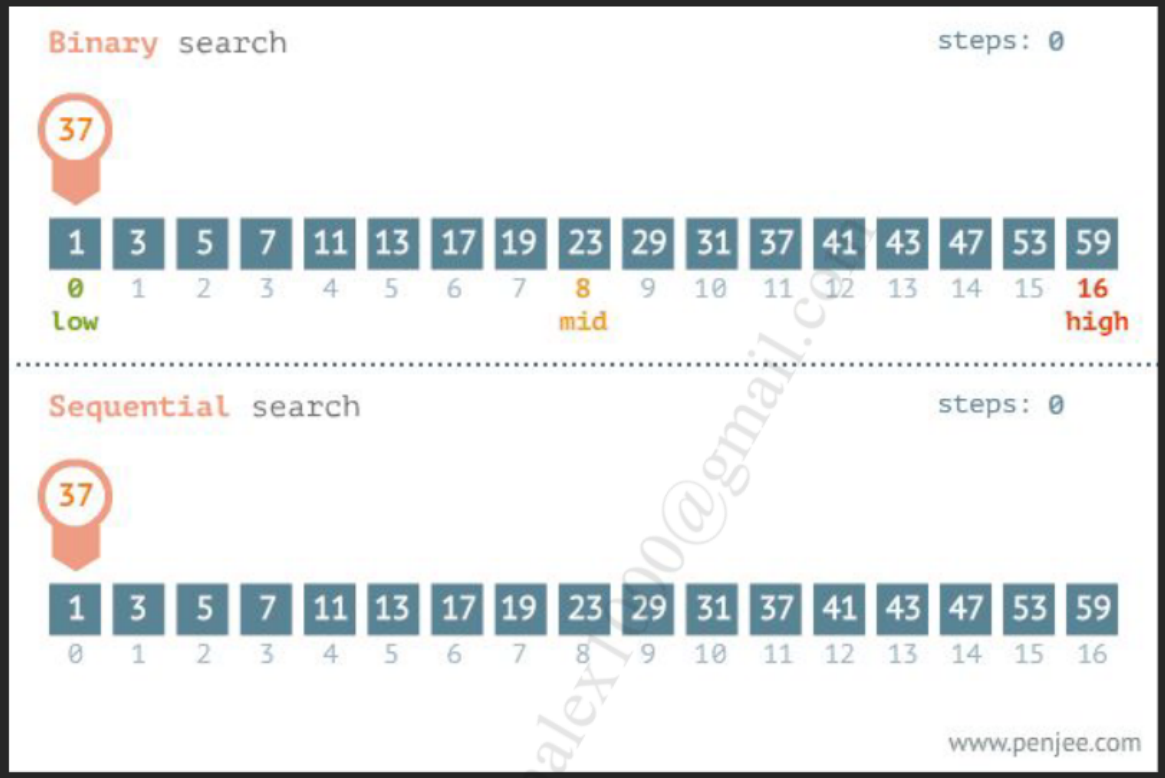
Target < mid (go left)

Step 3: mid = 10

arr = [10]

Target == mid

Visual Representation of Binary Search & Linear Search



Implementation

- Iterative
- Recursive
- Tweaks needed to the implementation
  - Example: Find first occurrence, last occurrence (when an element repeats).
- Are there **library methods** I can use for this?
  - Yes, most languages provide standard methods for binary search, just like sort.
  - But in most problems you'll have to customize the binary search a bit, so you would end up writing your own code.

03

Time & Space Complexity

Time Complexity & Space Complexity for Binary search

Time Complexity:-

Best	O(1)
Average	$O(\log_2 n)$
Worst	$O(\log_2 n)$

In comparison, linear search has average and worst-case time complexity of O(n).

Space Complexity:-

- O(1) for iterative version
- $O(\log_2 n)$ for recursive version (due to stack space)

Linear Search Vs Binary Search

Feature	Linear	Binary
Data Order	Any order	Sorted Only
Time Complexity	O(n)	O(log n)
Space Complexity	Slower	Faster

04

Applications

Applications of Binary Search

- Searching in databases
- Dictionary lookup
- Auto-complete suggestions
- Gaming (e.g., binary space partitioning)

pandyamalex1000@gmail.com

How to identify Binary Search Problems

- Working on **sorted array** - should consider Binary Search as an option.
  - Find number of occurrences of an element in a large search space
  - First or last occurrence of an element in a large search space

- Come up with the **Upper and Lower Bound values**, or Problems related to "**Maximise the minimum**" or "**Minimize the maximum**" something are usually solved using Binary Search (optimization problems)
  - Koko eating bananas 🏡
  - Square root of a number

Template for standard binary search

binarySearch.py

```python
def binary_search(arr, target):
    low, high = 0, len(arr) - 1

    while low <= high:
        mid = (low + high) // 2
        if arr[mid] == target:
            return mid  # Found
        elif arr[mid] < target:
            low = mid + 1
        else:
            high = mid - 1

    return -1  # Not found
```

Template for find first occurrence

findFirstOccurence.py

```python
def first_occurrence(arr, target):
    low, high = 0, len(arr) - 1
    ans = -1

    while low <= high:
        mid = (low + high) // 2
        if arr[mid] >= target:
            if arr[mid] == target:
                ans = mid  # Store answer
            high = mid - 1  # Move left
        else:
            low = mid + 1

    return ans
```

Template for standard binary search

findLastOccurence.py

```python
def last_occurrence(arr, target):
    low, high = 0, len(arr) - 1
    ans = -1

    while low <= high:
        mid = (low + high) // 2
        if arr[mid] <= target:
            if arr[mid] == target:
                ans = mid  # Store answer
            low = mid + 1  # Move right
        else:
            high = mid - 1

    return ans
```

05

Activity

Activity 1

- Binary Search
- Search in Rotated Sorted Array
- Median of 2 Sorted Arrays
- Search a 2D Matrix
- Find Minimum in Rotated Sorted Array
- First Bad Version
- Find Peak Element

pandiyanalex1000@gmail.com

Activity 2

- Kth Smallest Element in a Sorted Matrix
- Find First and Last Position of Element in Sorted Array
- Find the Duplicate Number
- sqrt(x)
- Single Element in a Sorted Array
- Find Smallest Letter Greater Than Target