# Binary Search

IIITS

# Linear Search

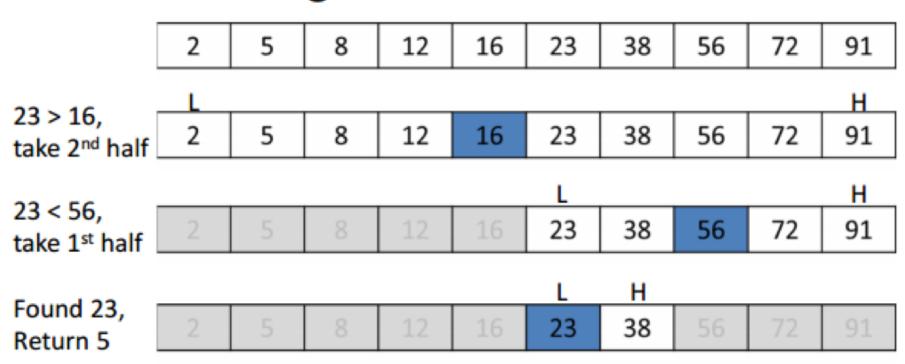Given a sorted array *arr[]* of *n* elements, write a function to search a given element *x* in *arr[]*.

A simple approach is to do **linear search**. The time complexity of above algorithm is *O(n)*. Another approach to perform the same task is using Binary Search.

# Binary Search

**Binary Search:** Search a sorted array by repeatedly dividing the search interval in half. Begin with an interval covering the whole array. If the value of the search key is less than the item in the middle of the interval, narrow the interval to the lower half. Otherwise narrow it to the upper half. Repeatedly check until the value is found or the interval is empty.

# Binary Search

## If searching for 23 in the 10-element array:

| 2 | 5 | 8 | 12 | 16 | 23 | 38 | 56 | 72 | 91 |
|---|---|---|----|----|----|----|----|----|----|

**23 > 16, take 2nd half**

| 2 | 5 | 8 | 12 | **16** | 23 | 38 | 56 | 72 | 91 |
|---|---|---|----|----|----|----|----|----|----|

L ... H

**23 < 56, take 1st half**

| 2 | 5 | 8 | 12 | 16 | 23 | 38 | **56** | 72 | 91 |
|---|---|---|----|----|----|----|----|----|----|

L ... H

**Found 23, Return 5**

| 2 | 5 | 8 | 12 | 16 | **23** | 38 | 56 | 72 | 91 |
|---|---|---|----|----|----|----|----|----|----|

L H

# Assignment

- Implement binary search for linked-list.
- Implement binary search for priority queue.