



SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with ‘A’ Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING-IOT Including CS&BCT

COURSE NAME : 19CS307- DATA STRUCTURES

II YEAR / III SEMESTER

Unit V- SORTING AND SEARCHING



TOPIC:Searching- Linear Search - Binary Search



Searching



- **What is Searching?**
- Searching is the process of finding a given value position in a list of values.
- It decides whether a search key is present in the data or not.
- It is the algorithmic process of finding a particular item in a collection of items.
- It can be done on internal data structure or on external data structure.



Searching Techniques



- Searching Techniques
- **To search an element in a given array, it can be done in following ways:**
 1. Sequential Search
 2. Binary Search



Sequential (or) Linear Search



- Sequential search is also called as **Linear Search**.
- Sequential search starts at the beginning of the list and **checks every element** of the list.
- It is a basic and **simple search** algorithm.
- Sequential search compares the element with all the other elements given in the list.
- If the element is matched, it returns the value index, else it returns -1.



Syntax

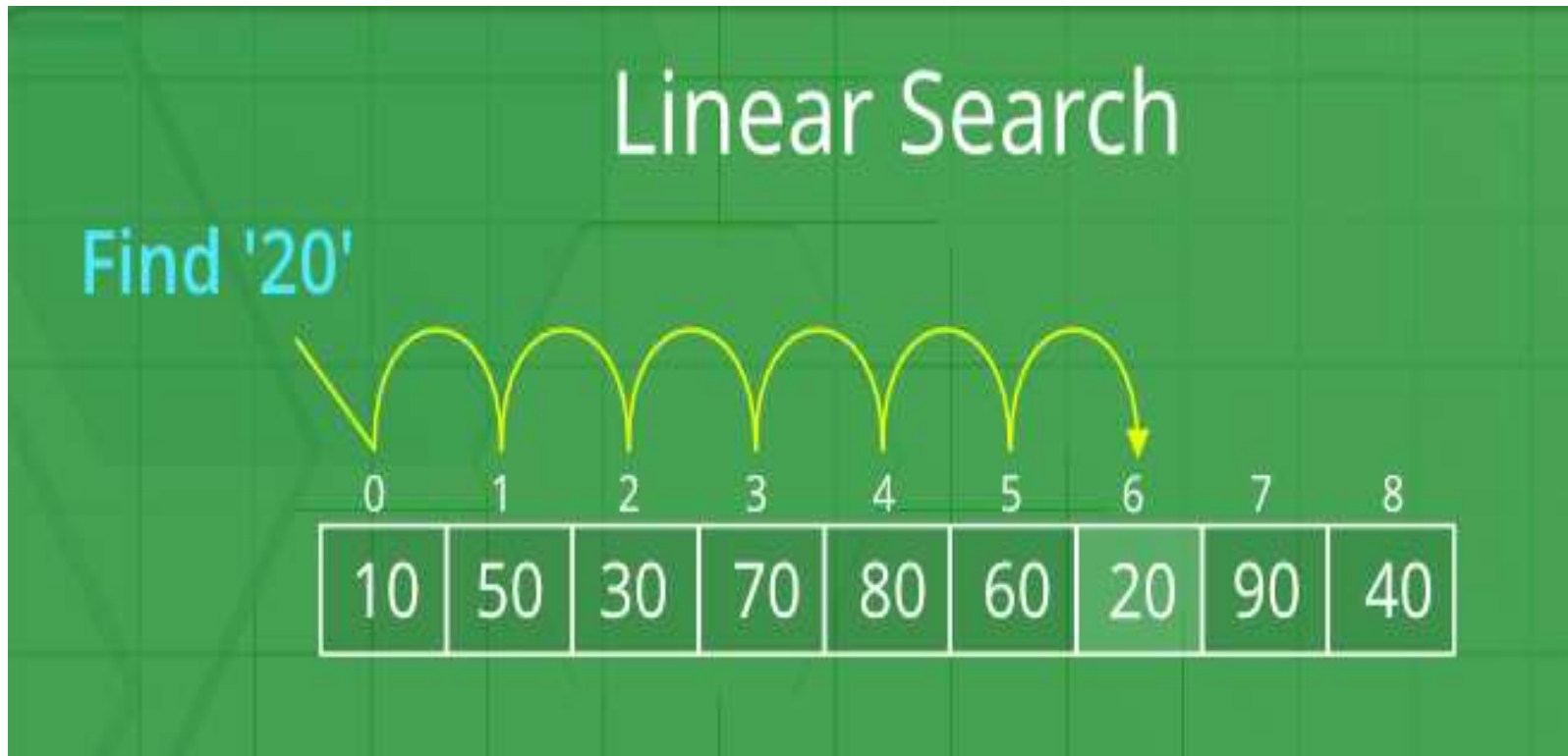


- **The following code snippet shows the sequential search operation**

- ```
function searchValue(value, target)
{
 for (var i = 0; i < value.length; i++)
 {
 if (value[i] == target)
 {
 return i;
 }
 }
 return -1;
}
```



# Example





# Binary Search



- Binary Search is used for **searching an element** in a sorted array.
- It is a **fast search algorithm** with run-time complexity of  $O(\log n)$ .
- Binary search works on the principle of **divide and conquer**.
- This searching technique looks for a particular element by comparing the middle most element of the collection.
- It is useful when there are large number of elements in an array.





# Program for Binary Search



- `#include<stdio.h>`  
`#include<conio.h>`  
`void main()`  
`{`  
    `int f, l, m, size, i, sElement, list[50];`
- `//int f, l, m : First, Last, Middle`  
    `clrscr();`
- `printf("Enter the size of the list: ");`  
    `scanf("%d",&size);`  
  
    `printf("Enter %d integer values : \n", size);`



# Cont..



- ```
for (i = 0; i < size; i++)
    scanf("%d",&list[i]);
printf("Enter value to be search: ");
scanf("%d", &sElement);
f = 0;
l = size - 1;
m = (f+l)/2;
```
- ```
while (f <= l) {
 if (list[m] < sElement)
 f = m + 1;
 else if (list[m] == sElement) {
 printf("Element found at index %d.\n",m);
 break;
 }
}
```



else

```
 l = m - 1;
```

```
 m = (f + l)/2;
```

```
}
```

```
if (f > l)
```

```
 printf("Element Not found in the list.");
```

```
 getch();
```

```
}
```



# Example

Search Element : 25

|   |    |    |    |    |    |    |
|---|----|----|----|----|----|----|
| 5 | 10 | 15 | 20 | 25 | 30 | 35 |
|---|----|----|----|----|----|----|

Starts with middle element

|   |    |    |    |    |    |    |
|---|----|----|----|----|----|----|
| 5 | 10 | 15 | 20 | 25 | 30 | 35 |
|---|----|----|----|----|----|----|

$25 > 20$

|   |    |    |    |    |    |    |
|---|----|----|----|----|----|----|
| 5 | 10 | 15 | 20 | 25 | 30 | 35 |
|---|----|----|----|----|----|----|

|   |    |    |    |    |    |    |
|---|----|----|----|----|----|----|
| 5 | 10 | 15 | 20 | 25 | 30 | 35 |
|---|----|----|----|----|----|----|

$25 < 30$

|   |    |    |    |    |    |    |
|---|----|----|----|----|----|----|
| 5 | 10 | 15 | 20 | 25 | 30 | 35 |
|---|----|----|----|----|----|----|

Element Found

|   |    |    |    |    |    |    |
|---|----|----|----|----|----|----|
| 5 | 10 | 15 | 20 | 25 | 30 | 35 |
|---|----|----|----|----|----|----|

Fig. Working Structure of Binary Search



# Thank you