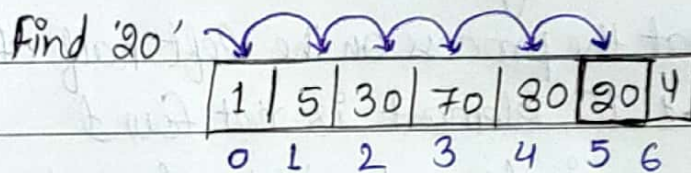


Experiment-1

Ans 2. Linear search on an array of n elements

- Steps :-
- (1) Input of an array of size n
 - (2) Make an array of size n
 - (3) Input of array elements (using loop)
 - (4) Input of element for which search happens (x)
 - (5) Compare x with each array element from left to right using loop.
 - (6) Return the index at which x was equal to $array[index]$



Time Complexity :- $O(n)$

f order of ' n '

Ans 3. Binary search on a sorted array of n elements (iterative)

Steps (1): Take array elements input in sorted manner

(2) Input, for which element to be searched (x)

(3) $mid = \frac{left + right}{2}$

(4) Check if $x = mid$, if true return mid

(5) else if $x > mid$ then update left as $mid + 1$

(6) else update right as $mid - 1$

7) Repeat above steps (3, 4, 5, 6) until $x = mid$

- 8) returns index when $x = \text{mid}$
 9) else return -1 (showing absence of element)

Binary search (recursive)

- Steps:-
- (1) Take array input in sorted way
 - (2) Input of element for which searched is to be done (x)
 - (3) Compare x with the middle element of array.
 - (4) if ($x == \text{array}[\text{mid}]$) return mid
 - (5) else if ($x > \text{array}[\text{mid}]$) repeat the process on the right half of the array.
 - (6) else repeat the process on the left half of the array
 - (7) return -1 if element is not found
 else return index at which element was found.

Search 23

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

$23 > 16$
take 2nd half

$L=0$				$M=4$					$R=9$
2	5	8	12	16	23	38	56	72	91

$23 > 56$
take 1st half

				$L=5$		$M=7$			$R=9$
2	5	8	12	16	23	38	56	72	91

Found 23
Return 5

					23				
				$M=5$					
				$L=5$		$R=6$			
2	5	8	12	16	23	38	56	72	91

Time Complexity:- $T = O(\log n)$

Ans 4. Binary search functions in STL library

Steps (1) Initialize the array

(2) `int aSize = sizeof(array) / sizeof(array[0]);`

(3) `sort(array, array + aSize)`

(4) take input for which element to be searched (x)

(5) now perform binary search i.e.,

`if (binary_search(array, array + sizeof(array), x))`

`print → element found`

`else`

`print → element not found`

(6) end.

Ans 5. Binary search functions in Java

`import java.util. Arrays;`

`class BinarySearch {`

`public static void main(String[] args) {`

`int arr[] = {10, 20, 30, 15, 22};`

`Arrays.sort(arr);`

`int key = 15;`

`int result = Arrays.binarySearch(arr, key);`

`if (result >= 0)`

`System.out.println(key + " Found at index : " + result);`

`else`

`System.out.println("Not Found");`

`}`

`}`