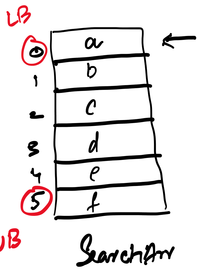
Binary Search

Tuesday, 17 November 2020 8:12 AM

- In BinSearch all elements in the array are to be sorted in ascending order.



SearchArr

Data	UB	LB	MID
'e'	5	0	2
		3	(5+3) DIV 2 = 4
'a'	5	0	2
	1		(1+0) DIV 2 = 0

MID ← (UB+LB) DIV 2

(5+0) DIV 2

2 DIV

2) 5

4

1

MID ← INT ((UB+LB) ÷ 2)

1. Input Data
2. FIND MID
3. Compare MID with the Data
4. IF Data matches with the MID element, function returns the MID index.
5. ELSE IF Data is greater than the MID element, then Data can only lie in the greater half (means; make LB ← MID + 1). Goto 2.
6. ElseIF Data is smaller than the MID element, then Data can only lie in the lower half (mean; make UB ← MID - 1). Go to 2.

PYTHON: (Iterative) MU-EDITOR

```
def binary_search(arr, data):
    lb = 0
    ub = len(arr) - 1
    while lb <= ub:
        mid = (ub + lb) // 2
        # Check if data is greater than mid
        if arr[mid] < data:
            lb = mid + 1
        # Check if data is smaller than mid
        elif arr[mid] > data:
            ub = mid - 1
        # Check if data is present at mid
        else:
            return mid
    # The required Data is not found
    return -1
```

PYTHON (Recursive):

```
def binary_search(arr, lb, ub, Data):
    if ub >= lb:
        mid = (ub + lb) // 2
        if arr[mid] == Data:
            return mid
        elif arr[mid] > Data:
            return binary_search(arr, low, mid - 1, Data)
        else:
            return binary_search(arr, mid + 1, ub, Data)
    else:
        return -1

# Test array
arr = [2, 3, 4, 10, 40]

Data = 10

# Function Call
result = binary_search(arr, 0, len(arr) - 1, Data)

if result == -1:
    print("Element is not present in given array.")
else:
    print("Element is present at position", str(result))
```