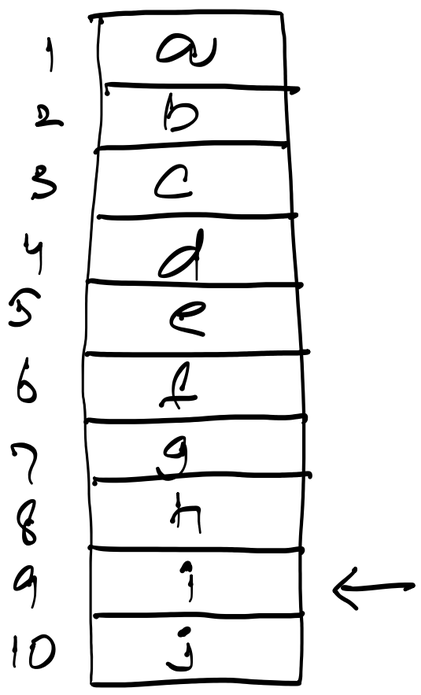
Binary Search

Saturday, 12 February 2022 4:34 PM




```
INPUT item
isFound = False
REPEAT
    Centre ← (lb+ub) DIV 2
    IF SA[Centre] = item
        THEN
            OUTPUT "found at :", Centre
            isFound = True
    ELSEIF item > SA[Centre]
        THEN
            lb ← Centre +1
    ELSE
            ub ← Centre -1
    ENDIF
Until isFound = True OR lb>ub.
IF isfound = False THEN OUTPUT "Not found!!!"
```

Algorithm
1. Linear Search
2. Bubble Sort
3. Binary Search
4. Insertion Sort.

isFound	item	Centre	lb	ub
F	'h'	5	1 6	10
F	'a'	5 2 1	1	10 4 1
F	'm'	5 8 9 10	1 6 9 10 11	10

Recursive Binary Search.

```
Function BSearch(item : char, lb : int, ub : int) Returns int
    DECLARE Centre : INT
    IF   lb > ub  THEN
        Return -1
    ELSE
        Centre ← (lb+ub) DIV 2
        IF  SA[Centre] = item  THEN
            Return Centre
        ELSEIF item > SA[Centre] THEN
            Return  BSearch(item, Centre +1, ub)
        ELSEIF item < SA[Centre]  THEN
            Return  BSearch(item, lb, Centre -1)
        ENDIF
    ENDIF
ENDFUNCTION.
```