

162. Implement the Binary Search algorithm in a programming language of your choice and test it on the array 5,10,15,20,25,30,35,40,45 to find the position of the element 20. Execute your code and provide the index of the element 20. Modify your implementation to count the number of comparisons made during the search process. Print this count along with the result.

Input : N= 9, a[] = {5,10,15,20,25,30,35,40,45}, search key = 20

Output : 4

Test cases

Input : N= 6, a[] = {10,20,30,40,50,60}, search key = 50

Output : 5

Input : N= 7, a[] = {21,32,40,54,65,76,87}, search key = 32

Output : 2

PROGRAM :-

```
def binary_search(arr, x):
```

```
    low = 0
```

```
    high = len(arr) - 1
```

```
    count = 0
```

```
    while low <= high:
```

```
        mid = (low + high) // 2
```

```
        count += 1
```

```
        if arr[mid] < x:
```

```
            low = mid + 1
```

```
        elif arr[mid] > x:
```

```
            high = mid - 1
```

```
        else:
```

```
            return mid, count
```

```
    return -1, count
```

```
# Test the binary search algorithm
```

```
arr = [5, 10, 15, 20, 25, 30, 35, 40, 45]
```

```
search_key = 20
```

```
position, comparisons = binary_search(arr, search_key)
```

```
print(f"Element {search_key} found at index {position} with {comparisons} comparisons.")
```

OUTPUT:-

```
Element 20 found at index 3 with 4 comparisons.
```

```
=== Code Execution Successful ===
```

TIME COMPLEXITY:-O(log N)