Data Structures

Searching Technique – Binary Search

CODE FORTHINGS

Algorithm - Binary Search Recursive

Algorithm

Binary_search_recursive(arr, key,low,high)

Input Specification:

arr	:	Array to hold the elements
key	:	Item to be searched
low	:	Index Variable to indicate starting point
high	:	Index Variable to indicate ending point

Output Specification:

mid	:	Positon of key element
-1	:	Key is not found

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
  mid = (low+high) / 2
  if (arr[mid] == key )
       return mid
  else if (key < a[mid])
    binary_search_recursive(arr,key,low,mid - 1)
  else
    binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
   mid = (low+high) / 2
    if (arr[mid] == key )
          return mid
   else if (key < a[mid])
     binary_search_recursive(arr,key,low,mid - 1)
   else
     binary_search_recursive(arr,key,mid+1,high)
else
      return -1
```

arr[SIZE]

SIZE = 10

2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
   mid = (low+high) / 2
    if (arr[mid] == key )
          return mid
   else if (key < a[mid])
     binary_search_recursive(arr,key,low,mid - 1)
   else
     binary_search_recursive(arr,key,mid+1,high)
else
       return -1
```

arr[SIZE]

SIZE = 10

key=12

2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

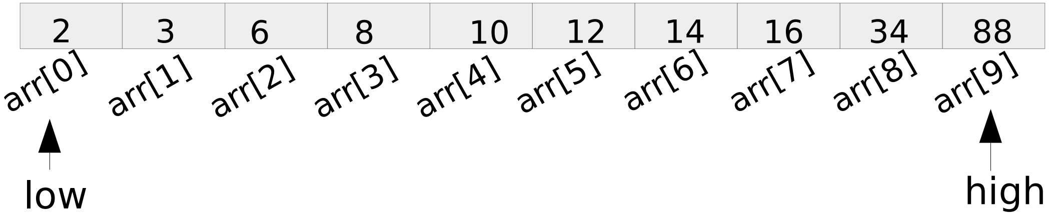
Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
   mid = (low+high) / 2
    if (arr[mid] == key )
         return mid
   else if (key < a[mid])
     binary_search_recursive(arr,key,low,mid - 1)
   else
     binary_search_recursive(arr,key,mid+1,high)
else
      return -1
```

arr[SIZE]

SIZE = 10

key=12



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

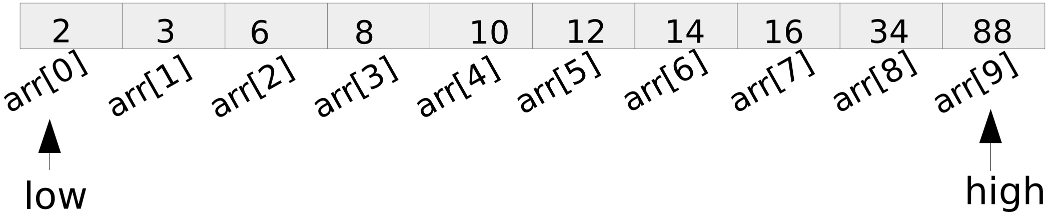
Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
   mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
   else if (key < a[mid])
    binary_search_recursive(arr,key,low,mid - 1)
   else
    binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

arr[SIZE]

SIZE = 10

key=12



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Data Structure – Algorithm

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
  mid = (low+high) / 2
   if (arr[mid] == key )
        return mid
  else if (key < a[mid])
    binary_search_recursive(arr,key,low,mid - 1)
  else
    binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

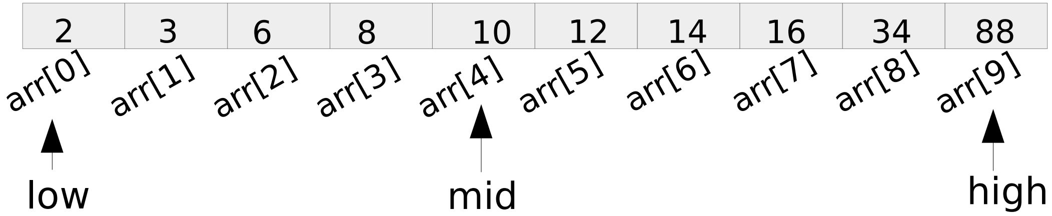
arr[SIZE]

SIZE = 10

key=12

mid

(0+9)/2 = 4



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
  mid = (low+high) / 2
  if (arr[mid] == key )
       return mid
  else if (key < a[mid])
    binary_search_recursive(arr,key,low,mid - 1)
  else
    binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

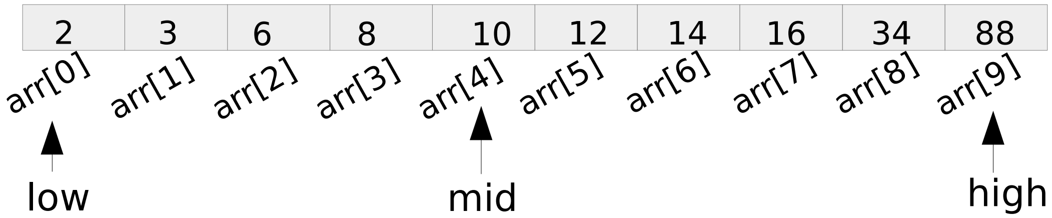
arr[SIZE]

SIZE = 10

key=12

mid

(0+9)/2 = 4



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Data Structure – Algorithm

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
   mid = (low+high) / 2
    if (arr[mid] == key )
          return mid
   else if (key < a[mid])
     binary_search_recursive(arr,key,low,mid - 1)
   else
     binary_search_recursive(arr,key,mid+1,high)
else
      return -1
```

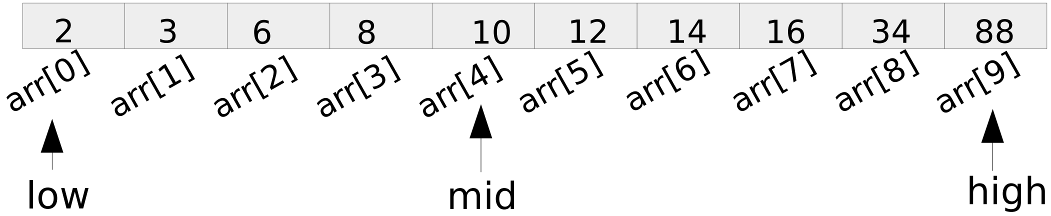
arr[SIZE]

SIZE = 10

key=12

mid

(0+9)/2 = 4



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
   mid = (low+high) / 2
    if (arr[mid] == key )
          return mid
   else if (key < a[mid])
     binary_search_recursive(arr,key,low,mid - 1)
   else
     binary_search_recursive(arr,key,mid+1,high)
else
      return -1
```

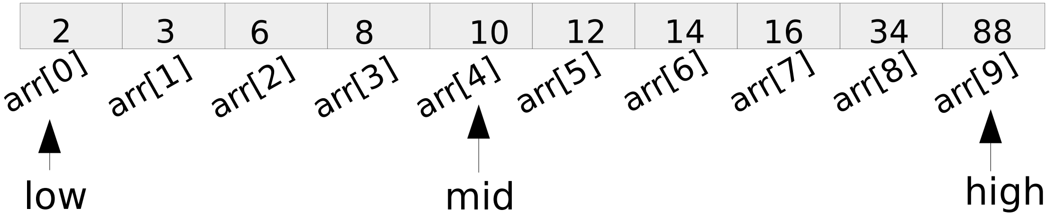
arr[SIZE]

SIZE = 10

key=12

mid

$(0+9)/2 = 4$



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
   mid = (low+high) / 2
   if (arr[mid] == key )
        return mid
   else if (key < a[mid])
     binary_search_recursive(arr,key,low,mid - 1)
   else
     binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

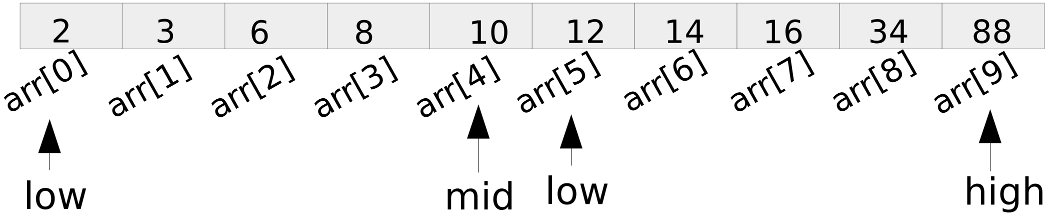
arr[SIZE]

SIZE = 10

key=12

mid

(0+9)/2 = 4



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Data Structure – Algorithm

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
   mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
   else if (key < a[mid])
    binary_search_recursive(arr,key,low,mid - 1)
   else
    binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

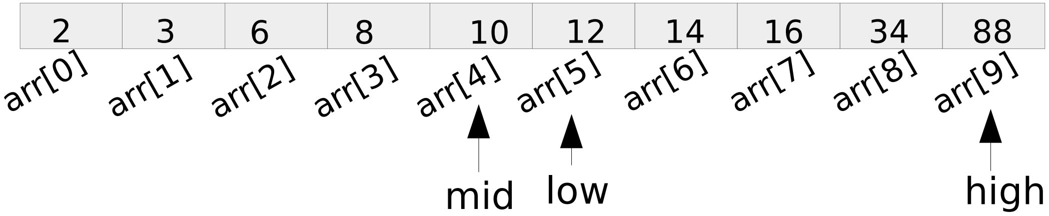
arr[SIZE]

SIZE = 10

key=12

mid

(0+9)/2 = 4



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Data Structure – Algorithm

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
  mid = (low+high) / 2
   if (arr[mid] == key )
       return mid
  else if (key < a[mid])
   binary_search_recursive(arr,key,low,mid - 1)
  else
   binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

arr[SIZE]

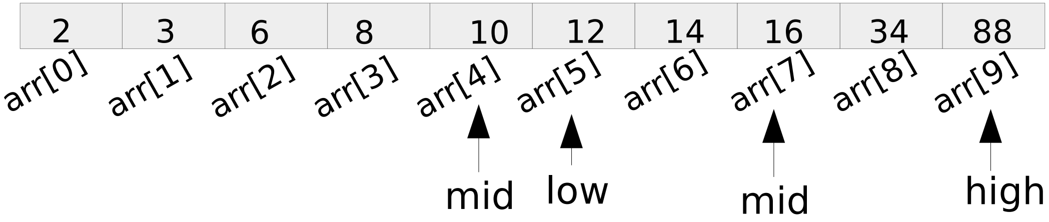
SIZE = 10

key=12

mid

(0+9)/2 = 4

(5+9)/2 = 7



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
   mid = (low+high) / 2
   if (arr[mid] == key )
         return mid
   else if (key < a[mid])
     binary_search_recursive(arr,key,low,mid - 1)
   else
     binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

arr[SIZE]

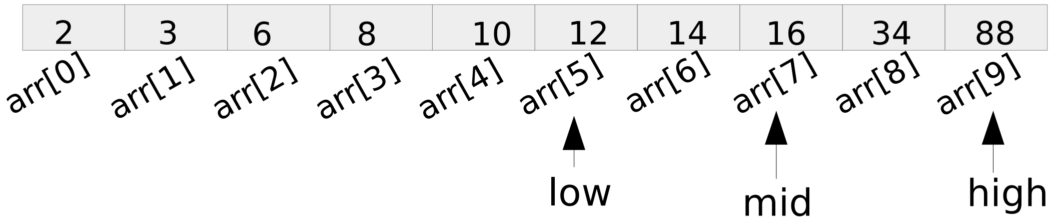
SIZE = 10

key=12

mid

$(0+9)/2 = 4$

$(5+9)/2 = 7$



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]
					low		mid		high

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
  mid = (low+high) / 2
  if (arr[mid] == key )
       return mid
  else if (key < a[mid])
    binary_search_recursive(arr,key,low,mid - 1)
  else
    binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

arr[SIZE]

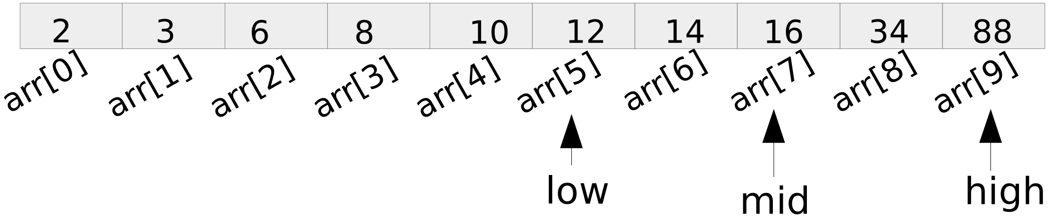
SIZE = 10

key=12

mid

(0+9)/2 = 4

(5+9)/2 = 7



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]
					low		mid		high

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
   mid = (low+high) / 2
   if (arr[mid] == key )
        return mid
   else if (key < a[mid])
     binary_search_recursive(arr,key,low,mid - 1)
   else
     binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

arr[SIZE]

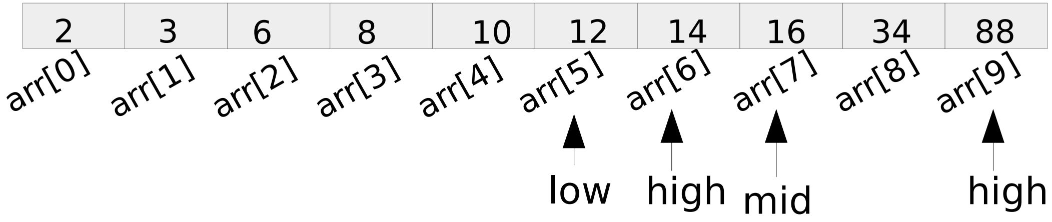
SIZE = 10

key=12

mid

(0+9)/2 = 4

(5+9)/2 = 7



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
   mid = (low+high) / 2
    if (arr[mid] == key )
        return mid
   else if (key < a[mid])
    binary_search_recursive(arr,key,low,mid - 1)
   else
    binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

arr[SIZE]

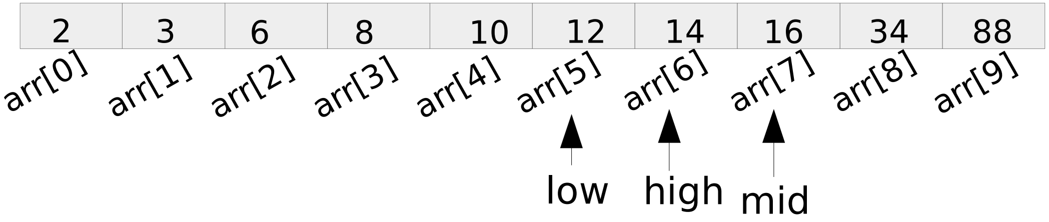
SIZE = 10

key=12

mid

(0+9)/2 = 4

(5+9)/2 = 7



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]
					low	high	mid		

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
  mid = (low+high) / 2
   if (arr[mid] == key )
       return mid
  else if (key < a[mid])
   binary_search_recursive(arr,key,low,mid - 1)
  else
   binary_search_recursive(arr,key,mid+1,high)
else
    return -1
```

arr[SIZE]

SIZE = 10

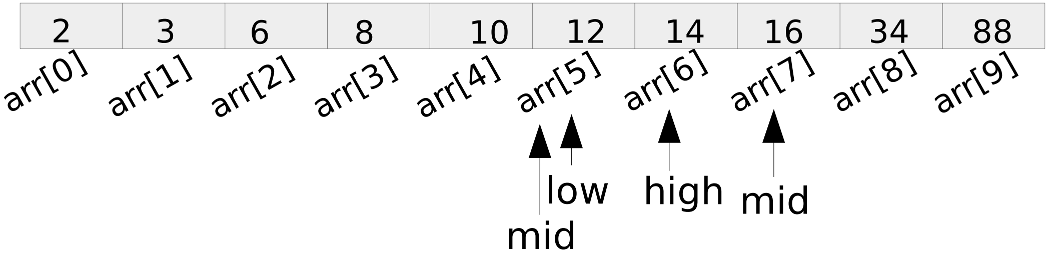
key=12

mid

(0+9)/2 = 4

(5+9)/2 = 7

(5+6)/2 = 5



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
  mid = (low+high) / 2
  if (arr[mid] == key )
      return mid
  else if (key < a[mid])
   binary_search_recursive(arr,key,low,mid - 1)
  else
   binary_search_recursive(arr,key,mid+1,high)
else
    return -1
```

arr[SIZE]

SIZE = 10

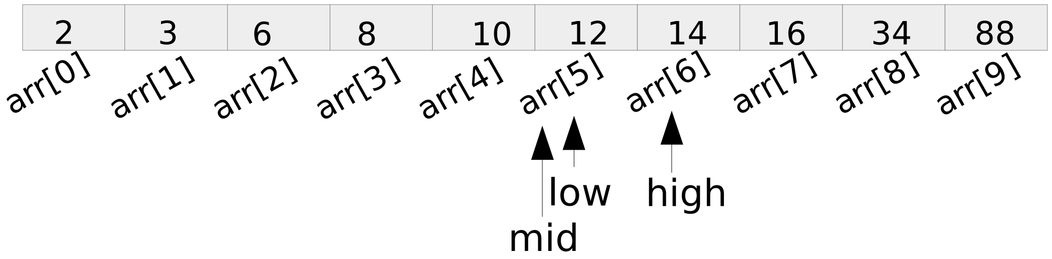
key=12

mid

(0+9)/2 = 4

(5+9)/2 = 7

(5+6)/2 = 5



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Data Structure – Algorithm

Binary_search_recursive(arr,key,low,high)

```
If  (low<= high)
  mid = (low+high) / 2
   if (arr[mid] == key )
      return mid
  else if (key < a[mid])
    binary_search_recursive(arr,key,low,mid - 1)
  else
    binary_search_recursive(arr,key,mid+1,high)
else
     return -1
```

arr[SIZE]

SIZE = 10

key=12

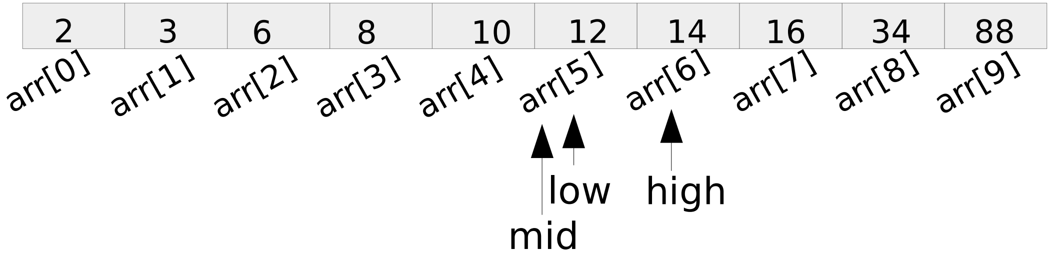
mid

(0+9)/2 = 4

(5+9)/2 = 7

(5+6)/2 = 5

mid=5



2	3	6	8	10	12	14	16	34	88
arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]	arr[6]	arr[7]	arr[8]	arr[9]

Data Structure – Binary Search

Comparison

Linear Search

- Time Complexity = O(n)

Data Structure – Binary Search

Comparison

Linear Search

- Time Complexity = O(n)

Binary Search Iterative

- Efficient
- Space Complexity = O(1)
- Time Complexity = O(logn)

Binary Search Recursive

- Easy
- Space Complexity = O(logn)
- Time Complexity = O(logn)

Code - Binary Search Recursive