Data Structure and Algorithm Practicals

13. Practical based on Divide and Conquer Technique-Binary Search

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">

<script>
let recursiveFunction = function (arr, x, start, end) {

        // Base Condition
        if (start > end) return false;

        // Find the middle index
        let mid=Math.floor((start + end)/2);

        // Compare mid with given key x
        if (arr[mid]===x) return true;

        // If element at mid is greater than x,
        // search in the left half of mid
        if(arr[mid] > x)
                return recursiveFunction(arr, x, start, mid-1);
        else

                // If element at mid is smaller than x,
                // search in the right half of mid
                return recursiveFunction(arr, x, mid+1, end);
}

// Driver code
let arr = [1, 3, 5, 7, 8, 9];
let x = 5;

if (recursiveFunction(arr, x, 0, arr.length-1))
        document.write("Element found!<br>");
else document.write("Element not found!<br>");

x = 6;

if (recursiveFunction(arr, x, 0, arr.length-1))
        document.write("Element found!<br>");
else document.write("Element not found!<br>");
</script>
```

```
    <title>Document</title>

</head>
<body>

</body>
</html>
```