

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1

Solution 2

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 public class Program {
4     // O(log(n)) time | O(1) space
5     public static int BinarySearch(int[] array, int target) {
6         return BinarySearch(array, target, 0, array.Length - 1);
7     }
8
9     public static int BinarySearch(int[] array, int target, int left, int right) {
10        while (left <= right) {
11            int middle = (left + right) / 2;
12            int potentialMatch = array[middle];
13            if (target == potentialMatch) {
14                return middle;
15            } else if (target < potentialMatch) {
16                right = middle - 1;
17            } else {
18                left = middle + 1;
19            }
20        }
21        return -1;
22    }
23 }
24
```

Solution 1

Solution 2

Solution 3

```
1 public class Program {
2     public static int BinarySearch(int[] array, int target) {
3         // Write your code here.
4         return -1;
5     }
6 }
7
```

Our Tests

Custom Output

Submit Code

```
1 public class Program {
2     // ...
3     public static void Main() {
4         int[] array = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
5         int target = 5;
6         int result = BinarySearch(array, target);
7         Console.WriteLine(result);
8     }
9 }
```

