

```

1  /*Author: Bochen (mddboc@foxmail.com)
2  Last Modified: Tue Apr 10 22:28:45 CST 2018*/
3
4  /*Given a sorted array and a target value, return the index if the target is found.
   If not, return the index where it would be if it were inserted in order.
5
6  .....You may assume no duplicates in the array.
7
8  .....Example 1:
9
10 .....Input: [1,3,5,6], 5
11 .....Output: 2
12 .....Example 2:
13
14 .....Input: [1,3,5,6], 2
15 .....Output: 1
16 .....Example 3:
17
18 .....Input: [1,3,5,6], 7
19 .....Output: 4
20 .....Example 1:
21
22 .....Input: [1,3,5,6], 0
23 .....Output: 0*/
24
25
26 import java.util.*;
27 import java.lang.Math;
28 import java.lang.System;
29 import java.lang.Integer;
30
31
32 public class Main {
33
34     ....public static void main(String[] args) throws ArithmeticException {
35
36         .....int[] input = {7, 1, 5, 3, 6, 4};
37
38         .....Solution solution = new Solution();
39
40         .....int result = solution.maxProfit(input);
41
42         .....System.out.println("haha");
43     ....}
44
45 }
46
47
48 class ListNode {
49     ....int val;
50     ....ListNode next;
51
52     ....ListNode(int x) {
53         .....val = x;
54     ....}
55 }
56
57
58 class TreeNode {
59     ....int val;
60     ....TreeNode left;
61     ....TreeNode right;
62
63     ....TreeNode(int x) {
64         .....val = x;
65     ....}
66 }
67
68
69 class Solution {
70     ....public int searchInsert(int[] nums, int target) {
71
72         .....if (nums[0] >= target) {

```

```

73     ..... return 0;
74     ..... }
75     ..... if (nums[nums.length - 1] == target) {
76     .....     return nums.length - 1;
77     ..... } else if (nums[nums.length - 1] < target) {
78     .....     return nums.length;
79     ..... }
80
81     ..... int startPoint = 0, endPoint = nums.length - 1;
82     ..... while (startPoint <= endPoint) {
83
84     .....     int mid = (startPoint + endPoint) / 2;
85     .....     if (nums[mid] == target) {
86     .....         return mid;
87     .....     }
88
89     .....     if (nums[mid] < target) {
90     .....         startPoint = mid + 1;
91     .....     } else {
92     .....         endPoint = mid - 1;
93     .....     }
94     ..... }
95     ..... return startPoint;
96     ..... }
97 }

```