

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1Solution 2

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 #include <vector>
4 #include <climits>
5 using namespace std;
6
7 int binarySearch(int startIdx, int endIdx, vector<int> indices,
8                 vector<int> array, int num);
9 vector<int> buildSequence(vector<int> array, vector<int> sequences,
10                          int currentIdx);
11
12 // O(nlogn) time | O(n) space
13 vector<int> longestIncreasingSubsequence(vector<int> array) {
14     vector<int> sequences(array.size(), 0);
15     vector<int> indices(array.size() + 1, INT_MIN);
16     int length = 0;
17     for (int i = 0; i < array.size(); i++) {
18         int num = array[i];
19         int newLength = binarySearch(1, length, indices, array, num);
20         sequences[i] = indices[newLength - 1];
21         indices[newLength] = i;
22         length = max(length, newLength);
23     }
24     return buildSequence(array, sequences, indices[length]);
25 }
26
27 int binarySearch(int startIdx, int endIdx, vector<int> indices,
28                 vector<int> array, int num) {
29     if (startIdx > endIdx) {
30         return startIdx;
31     }
32     int middleIdx = (startIdx + endIdx) / 2;
33     if (array[indices[middleIdx]] < num) {
```

Solution 1Solution 2Solution 3

```
1 #include <vector>
2 using namespace std;
3
4 vector<int> longestIncreasingSubsequence(vector<int> array) {
5     // Write your code here.
6     return {};
7 }
8
```

Our Tests

Custom Output

Submit Code

```
10 vector<int> vector1{0, 1, 2, 3, 4, 5};
11 vector<int> vector2{0, 1, 2, 3, 4, 5, 6, 7};
12 vector<int> vector3{0, 1, 2, 3, 4, 5};
13
14 // Copy Constructor - public TestVector()
15 // public:
16 //     Test() {}
17
18 // TestVector(const TestVector &t): Test() {}
19 //     vector<int> vector1{0};
20 //     vector<int> vector2{0, 1, 2, 3, 4, 5, 6, 7};
21 //     cout<<endl;
22
23 // TestVector(const TestVector &t): Test() {}
24 //     vector<int> vector1{0, 1};
```

Run or submit code when you're ready.