AlgoExpert   Quad Layout   Python   14px   Sublime   Monokai   00:00:

Our Solution(s)   Run Code

Solution 1   Solution 2

```python
# Copyright © 2020 AlgoExpert, LLC. All rights reserved.

# O(nlogn) time | O(n) space
def longestIncreasingSubsequence(array):
    sequences = [None for x in array]
    indices = [None for x in range(len(array) + 1)]
    length = 0
    for i, num in enumerate(array):
        newLength = binarySearch(1, length, indices, array, num)
        sequences[i] = indices[newLength - 1]
        indices[newLength] = i
        length = max(length, newLength)
    return buildSequence(array, sequences, indices[length])


def binarySearch(startIdx, endIdx, indices, array, num):
    if startIdx > endIdx:
        return startIdx
    middleIdx = (startIdx + endIdx) // 2
    if array[indices[middleIdx]] < num:
        startIdx = middleIdx + 1
    else:
        endIdx = middleIdx - 1
    return binarySearch(startIdx, endIdx, indices, array, num)


def buildSequence(array, sequences, currentIdx):
    sequence = []
    while currentIdx is not None:
        sequence.append(array[currentIdx])
        currentIdx = sequences[currentIdx]
    return list(reversed(sequence))
```

Your Solutions   Run Code

Solution 1   Solution 2   Solution 3

```python
def longestIncreasingSubsequence(array):
    # Write your code here.
    pass
```

Our Tests

[illegible]

Custom Output   Submit Code

[illegible]

Run or submit code when you're ready.