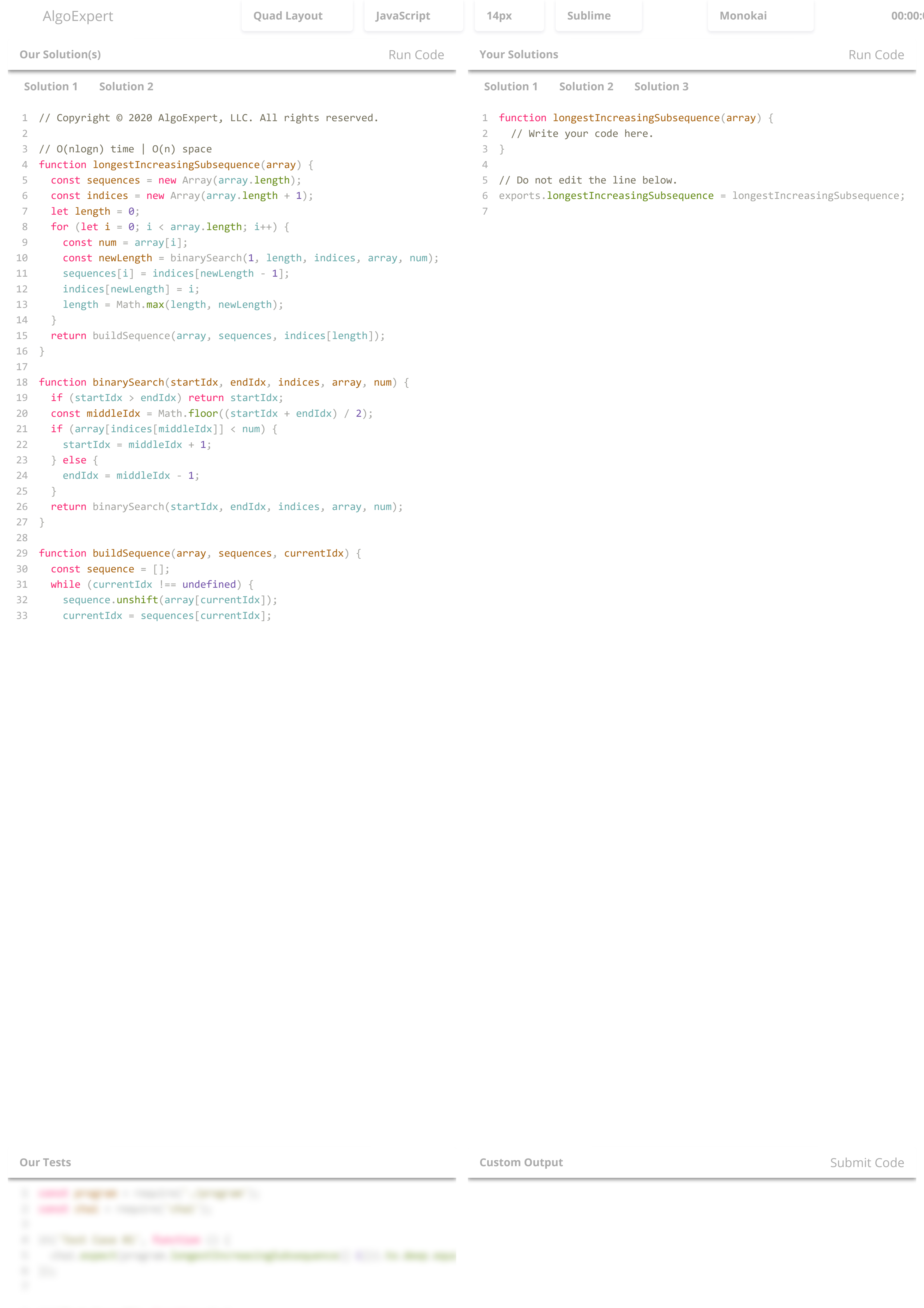

Our Solution(s)

```javascript
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

// O(nlogn) time | O(n) space
function longestIncreasingSubsequence(array) {
  const sequences = new Array(array.length);
  const indices = new Array(array.length + 1);
  let length = 0;
  for (let i = 0; i < array.length; i++) {
    const num = array[i];
    const newLength = binarySearch(1, length, indices, array, num);
    sequences[i] = indices[newLength - 1];
    indices[newLength] = i;
    length = Math.max(length, newLength);
  }
  return buildSequence(array, sequences, indices[length]);
}

function binarySearch(startIdx, endIdx, indices, array, num) {
  if (startIdx > endIdx) return startIdx;
  const middleIdx = Math.floor((startIdx + endIdx) / 2);
  if (array[indices[middleIdx]] < num) {
    startIdx = middleIdx + 1;
  } else {
    endIdx = middleIdx - 1;
  }
  return binarySearch(startIdx, endIdx, indices, array, num);
}

function buildSequence(array, sequences, currentIdx) {
  const sequence = [];
  while (currentIdx !== undefined) {
    sequence.unshift(array[currentIdx]);
    currentIdx = sequences[currentIdx];
```

Your Solutions

```javascript
function longestIncreasingSubsequence(array) {
  // Write your code here.
}

// Do not edit the line below.
exports.longestIncreasingSubsequence = longestIncreasingSubsequence;
```

Our Tests

[illegible]

Custom Output

```
it('Test Case #[illegible]', function () {
  chai.expect(program.longestIncreasingSubsequence([illegible])).to.deep.[illegible]
});

it('Test Case #[illegible]', function () {
  chai.expect(program.longestIncreasingSubsequence([illegible])).to.[illegible]
});

it('Test Case #[illegible]', function () {
  chai.expect(program.longestIncreasingSubsequence([illegible])).[illegible]
});

it('Test Case #[illegible]', function () {
  chai.expect(program.longestIncreasingSubsequence([illegible]
});
```

Run or submit code when you're ready.