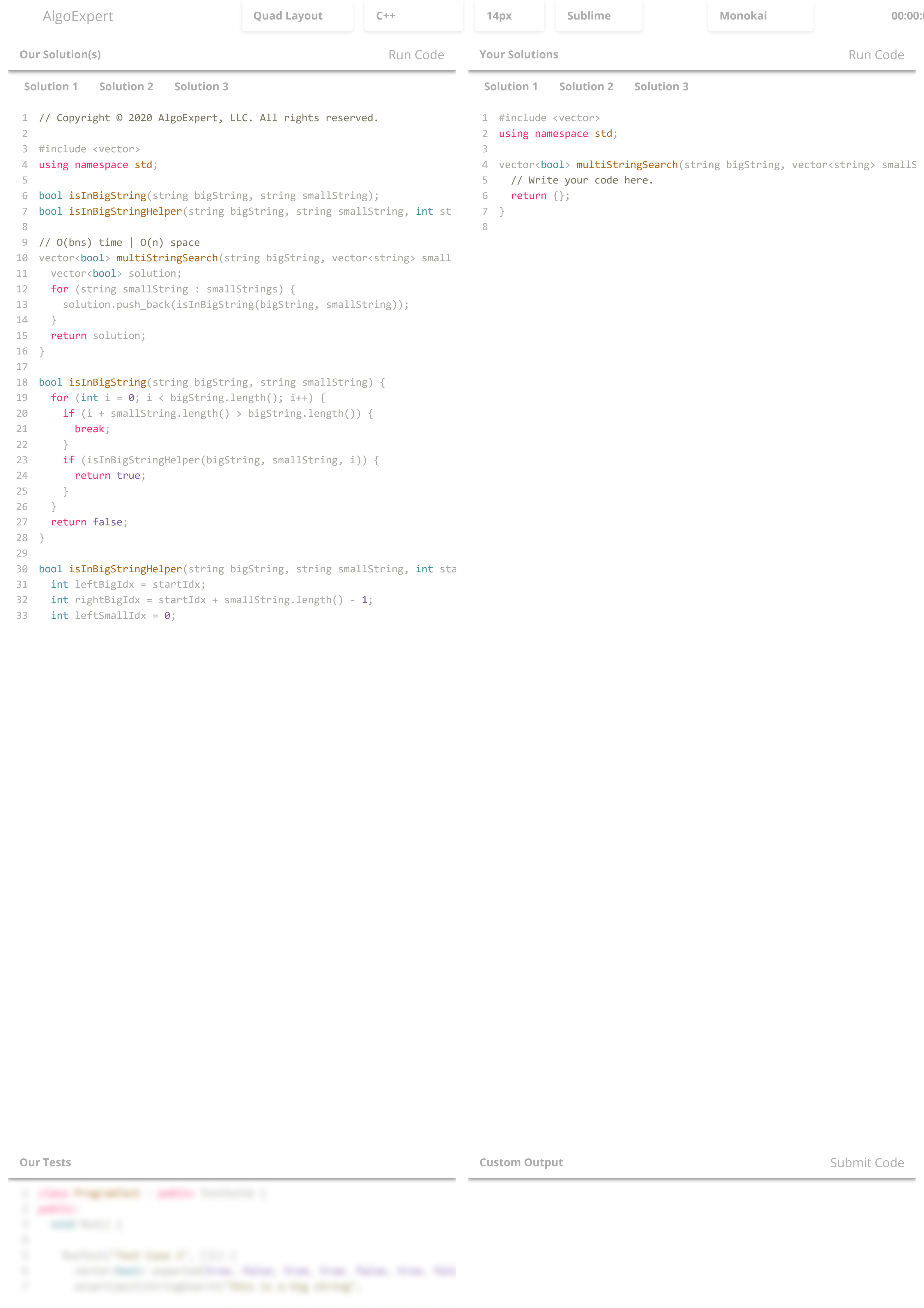

Our Solution(s)

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

#include <vector>
using namespace std;

bool isInBigString(string bigString, string smallString);
bool isInBigStringHelper(string bigString, string smallString, int st

// O(bns) time | O(n) space
vector<bool> multiStringSearch(string bigString, vector<string> small
  vector<bool> solution;
  for (string smallString : smallStrings) {
    solution.push_back(isInBigString(bigString, smallString));
  }
  return solution;
}

bool isInBigString(string bigString, string smallString) {
  for (int i = 0; i < bigString.length(); i++) {
    if (i + smallString.length() > bigString.length()) {
      break;
    }
    if (isInBigStringHelper(bigString, smallString, i)) {
      return true;
    }
  }
  return false;
}

bool isInBigStringHelper(string bigString, string smallString, int sta
  int leftBigIdx = startIdx;
  int rightBigIdx = startIdx + smallString.length() - 1;
  int leftSmallIdx = 0;
```

Your Solutions

```
#include <vector>
using namespace std;

vector<bool> multiStringSearch(string bigString, vector<string> smallS
  // Write your code here.
  return {};
}

```

Our Tests

[illegible]

Custom Output

Run or submit code when you're ready.