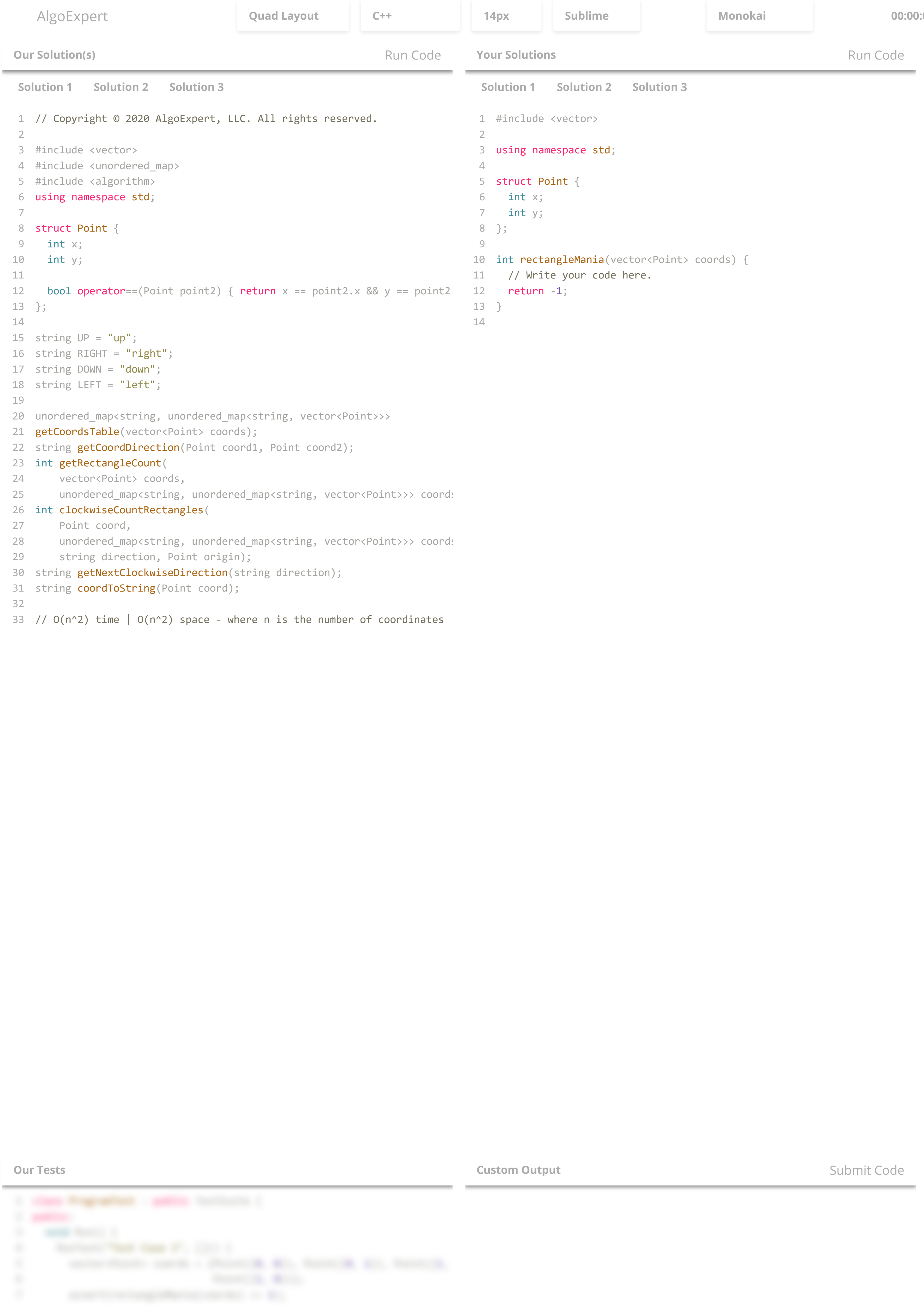

```cpp
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

#include <vector>
#include <unordered_map>
#include <algorithm>
using namespace std;

struct Point {
  int x;
  int y;

  bool operator==(Point point2) { return x == point2.x && y == point2.
};

string UP = "up";
string RIGHT = "right";
string DOWN = "down";
string LEFT = "left";

unordered_map<string, unordered_map<string, vector<Point>>>
getCoordsTable(vector<Point> coords);
string getCoordDirection(Point coord1, Point coord2);
int getRectangleCount(
    vector<Point> coords,
    unordered_map<string, unordered_map<string, vector<Point>>> coord
int clockwiseCountRectangles(
    Point coord,
    unordered_map<string, unordered_map<string, vector<Point>>> coord
    string direction, Point origin);
string getNextClockwiseDirection(string direction);
string coordToString(Point coord);

// O(n^2) time | O(n^2) space - where n is the number of coordinates
```

```cpp
#include <vector>

using namespace std;

struct Point {
  int x;
  int y;
};

int rectangleMania(vector<Point> coords) {
  // Write your code here.
  return -1;
}

```

```
[illegible]
```

Run or submit code when you're ready.