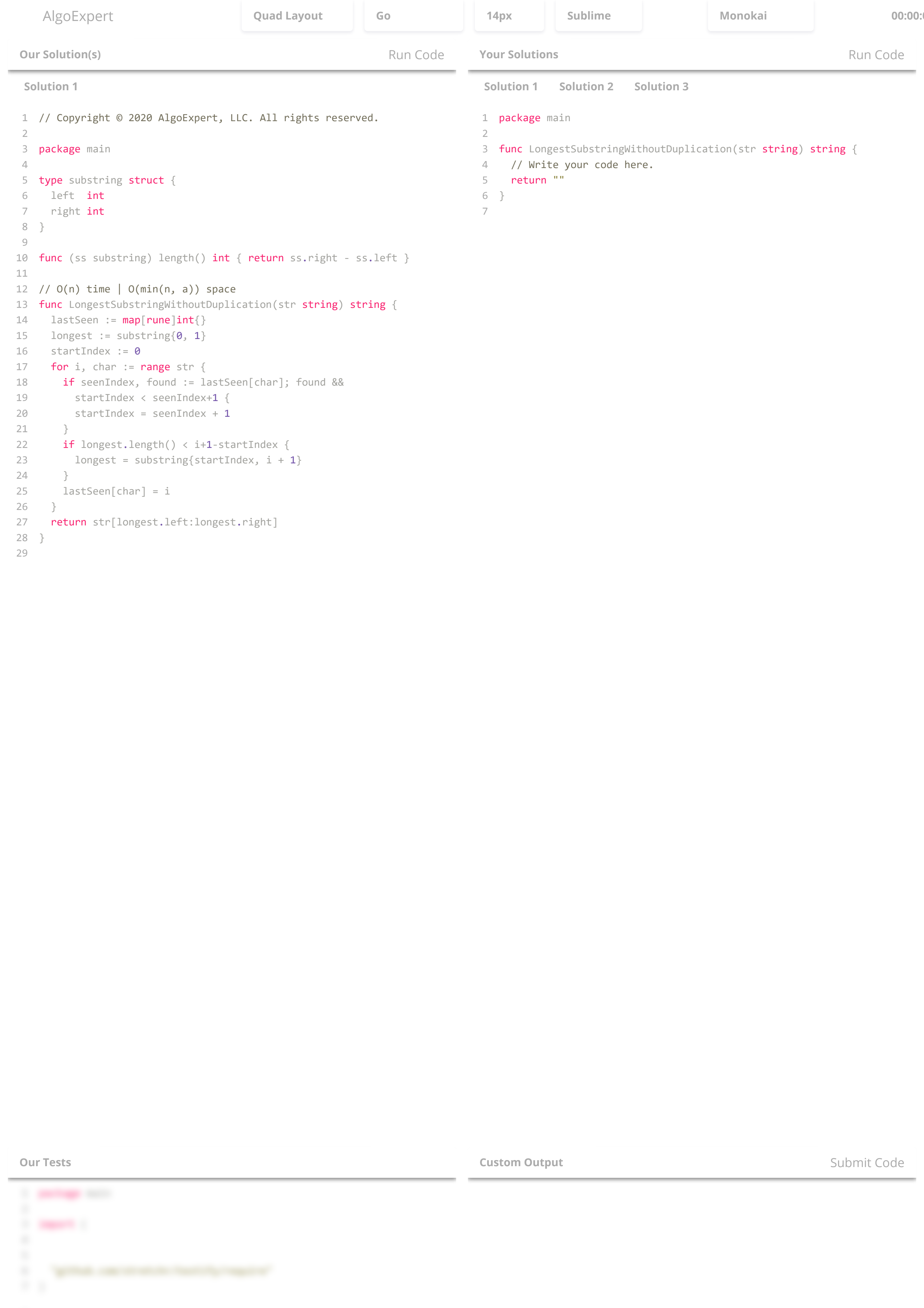

```go
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

package main

type substring struct {
  left  int
  right int
}

func (ss substring) length() int { return ss.right - ss.left }

// O(n) time | O(min(n, a)) space
func LongestSubstringWithoutDuplication(str string) string {
  lastSeen := map[rune]int{}
  longest := substring{0, 1}
  startIndex := 0
  for i, char := range str {
    if seenIndex, found := lastSeen[char]; found &&
      startIndex < seenIndex+1 {
      startIndex = seenIndex + 1
    }
    if longest.length() < i+1-startIndex {
      longest = substring{startIndex, i + 1}
    }
    lastSeen[char] = i
  }
  return str[longest.left:longest.right]
}

```

```go
package main

func LongestSubstringWithoutDuplication(str string) string {
  // Write your code here.
  return ""
}

```

```go
package main

import (

  "github.com/stretchr/testify/require"
)
```

Run or submit code when you're ready.