Our Solution(s)

Solution 1

```go
1  // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3  package main
4
5  type MinMaxStack struct {
6    stack        []int
7    minMaxStack []entry
8  }
9
10 type entry struct {
11   min int
12   max int
13 }
14
15 func NewMinMaxStack() *MinMaxStack {
16   return &MinMaxStack{}
17 }
18
19 // O(1) time | O(1) space
20 func (stack *MinMaxStack) Peek() int {
21   return stack.stack[len(stack.stack)-1]
22 }
23
24 // O(1) time | O(1) space
25 func (stack *MinMaxStack) Pop() int {
26   stack.minMaxStack = stack.minMaxStack[:len(stack.minMaxStack)-1]
27   out := stack.stack[len(stack.stack)-1]
28   stack.stack = stack.stack[:len(stack.stack)-1]
29   return out
30 }
31
32 // O(1) time | O(1) space
33 func (stack *MinMaxStack) Push(number int) {
34   newMinMax := entry{min: number, max: number}
35   if len(stack.minMaxStack) > 0 {
36     lastMinMax := stack.minMaxStack[len(stack.minMaxStack)-1]
37     newMinMax.min = min(lastMinMax.min, number)
38     newMinMax.max = max(lastMinMax.max, number)
39   }
40   stack.minMaxStack = append(stack.minMaxStack, newMinMax)
41   stack.stack = append(stack.stack, number)
42 }
43
44 // O(1) time | O(1) space
45 func (stack *MinMaxStack) GetMin() int {
46   return stack.minMaxStack[len(stack.minMaxStack)-1].min
47 }
48
49 // O(1) time | O(1) space
50 func (stack *MinMaxStack) GetMax() int {
51   return stack.minMaxStack[len(stack.minMaxStack)-1].max
52 }
53
54 func min(a, b int) int {
55   if a < b {
56     return a
57   }
58   return b
59 }
60
61 func max(a, b int) int {
62   if a < b {
63     return b
64   }
65   return a
66 }
67
```

Your Solutions

Solution 1   Solution 2   Solution 3

```go
1  package main
2
3  type MinMaxStack struct {
4    // Write your code here.
5  }
6
7  func NewMinMaxStack() *MinMaxStack {
8    // Write your code here.
9    return nil
10 }
11
12 func (stack *MinMaxStack) Peek() int {
13   // Write your code here.
14   return -1
15 }
16
17 func (stack *MinMaxStack) Pop() int {
18   // Write your code here.
19   return -1
20 }
21
22 func (stack *MinMaxStack) Push(number int) {
23   // Write your code here.
24 }
25
26 func (stack *MinMaxStack) GetMin() int {
27   // Write your code here.
28   return -1
29 }
30
31 func (stack *MinMaxStack) GetMax() int {
32   // Write your code here.
33   return -1
34 }
35
```

Custom Output   Raw Output   Submit Code

Run or submit code when you're ready.