

```
1 package StackAndQueue;
2
3 class FindMinimumElement {
4
5     ImplementStackWithLinkedList<Integer> stack;
6     ImplementStackWithLinkedList<Integer> minValueStack;
7
8     public FindMinimumElement() {
9
10         stack = new ImplementStackWithLinkedList<Integer>();
11         minValueStack = new ImplementStackWithLinkedList<Integer>();
12     }
13
14     public void push(Integer val) {
15
16         stack.push(val);
17
18         if ( minValueStack.isEmpty() || val < minValueStack.peek() ) {
19             minValueStack.push(val);
20         }
21     }
22
23     public Integer pop() {
24
25         if ( stack.isEmpty() ) {
26             return null;
27         }
28
29         Integer val = stack.pop();
30         if ( val.equals(minValueStack.peek()) ) {
31             minValueStack.pop();
32         }
33
34         return val;
35     }
36
37     public Integer getMinValue() {
38         if ( minValueStack.isEmpty() ) {
39             return null;
40         } else {
41             return minValueStack.peek();
42         }
43     }
44
45
46     public Integer peek() {
47         return stack.peek();
48     }
49 }
```