```java
package StackAndQueue;

class FindMinimumElement {

    ImplementStackWithLinkedList<Integer> stack;
    ImplementStackWithLinkedList<Integer> minValueStack;

    public FindMinimumElement() {

        stack = new ImplementStackWithLinkedList<Integer>();
        minValueStack = new ImplementStackWithLinkedList<Integer>();
    }

    public void push(Integer val) {

        stack.push(val);

        if ( minValueStack.isEmpty() || val < minValueStack.peek() ) {
            minValueStack.push(val);
        }
    }

    public Integer pop() {

        if ( stack.isEmpty() ) {
            return null;
        }

        Integer val = stack.pop();
        if ( val.equals(minValueStack.peek()) ) {
            minValueStack.pop();
        }

        return val;
    }

    public Integer getMinValue() {
        if ( minValueStack.isEmpty() ) {
            return null;
        } else {
            return minValueStack.peek();
        }
    }


    public Integer peek() {
        return stack.peek();
    }
}
```