

数据结构-栈

栈对应的最小栈：

155. 最小栈

难度 简单

👍 991



设计一个支持 `push` , `pop` , `top` 操作，并能在常数时间内检索到最小元素的栈。

- `push(x)` —— 将元素 `x` 推入栈中。
- `pop()` —— 删除栈顶的元素。
- `top()` —— 获取栈顶元素。
- `getMin()` —— 检索栈中的最小元素。

```
class Solution {
public:
    stack<int> stackData; // 保存数据用的栈stackData
    stack<int> stackMin; // 保存最小的数的栈stackMin，其中它的栈顶始终为最小的数
    void push(int value) {
        stackData.push(value);
        if(stackMin.empty())
            stackMin.push(value); // 如果stackMin为空，则value是最小的值，入栈
        else{
            if(stackMin.top() >= value)
                stackMin.push(value); // 否则当value小于等于stackMin的栈顶元素
            // 时，入栈（等于的时候也入栈是因为我考虑有相同的数）
        }
    }
    void pop() {
        if(stackData.top() == stackMin.top()) // 如果出栈的数刚好是最小的数，那么
            stackMin也应该出栈
        stackMin.pop();
        stackData.pop();
    }
    int top() {
        return stackData.top(); // 栈顶元素应返回stackData的栈顶元素
    }
};
```

```
    }  
    int min() {  
        return stackMin.top(); //stackMin的栈顶元素即是最小的数  
    }  
};
```

版权声明：本文为CSDN博主「GeekZW」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/zichen_ziqi/article/details/80807989