

用两个栈实现队列

题目描述

用两个栈来实现一个队列，完成队列的Push和Pop操作。 队列中的元素为int类型。

stack1负责入队, stack2负责出队

有元素入队, 直接stack1进栈, 一旦有元素要出队, 则跳转到stack2. 如果stack2为空, 就把stack1中的所有元素逐一出栈, 入stack2栈. 这样, stack1中的元素以相反的顺序存到了stack2中. 要出队的元素就是stack2 的栈顶元素. 只要stack2不为空, 所有的出队操作就都这样进, 如果stack2为空, 说明上一次从stack1中取出的元素均已出队, 则stack1依次出栈, 入stack2栈.

```
import java.util.Stack;
```

```
public class Solution {  
    Stack<Integer> stack1 = new Stack<Integer>();  
    Stack<Integer> stack2 = new Stack<Integer>();  
  
    public void push(int node) {  
        stack1.push(node);  
    }  
  
    public int pop() {  
        if(stack2.empty()){  
            while(!stack1.empty()){  
                int a = stack1.pop();  
                stack2.push(a);  
            }  
        }  
        return stack2.pop();  
    }  
}
```