# 9. 用两个栈实现队列

## 题目链接

[牛客网](https://www.nowcoder.com/practice/54275ddae22f475981afa2244dd448c6?tpId=13&tqId=11158&tPage=1&rp=1&ru=/ta/coding-interviews&qru=/ta/coding-interviews/question-ranking&from=cyc_github)

## 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。

## 解题思路

in 栈用来处理入栈（push）操作，out 栈用来处理出栈（pop）操作。一个元素进入 in 栈之后，出栈的顺序被反转。当元素要出栈时，需要先进入 out 栈，此时元素出栈顺序再一次被反转，因此出栈顺序就和最开始入栈顺序是相同的，先进入的元素先退出，这就是队列的顺序。

Stack<Integer> in = new Stack<Integer>();
Stack<Integer> out = new Stack<Integer>();

public void push(int node) {
 in.push(node);
}

public int pop() throws Exception {
 if (out.isEmpty())
 while (!in.isEmpty())
 out.push(in.pop());

 if (out.isEmpty())
 throw new Exception("queue is empty");

 return out.pop();
}