

如何仅用递归函数和栈操作逆序一个栈

【题目】

一个栈依次压入 1、2、3、4、5，那么从栈顶到栈底分别为 5、4、3、2、1。将这个栈转置后，从栈顶到栈底为 1、2、3、4、5，也就是实现栈中元素的逆序，但是只能用递归函数来实现，不能用其他数据结构。

【难度】

尉 ★★☆☆

【解答】

本题考查栈的操作和递归函数的设计，我们需要设计出两个递归函数。

递归函数一：将栈 `stack` 的栈底元素返回并移除。

具体过程就是如下代码中的 `getAndRemoveLastElement` 方法。

```
public static int getAndRemoveLastElement(Stack<Integer> stack) {
    int result = stack.pop();
    if (stack.isEmpty()) {
        return result;
    } else {
        int last = getAndRemoveLastElement(stack);
        stack.push(result);
        return last;
    }
}
```

如果从 `stack` 的栈顶到栈底依次为 3、2、1，这个函数的具体过程如图 1-4 所示。

递归函数二：逆序一个栈，就是题目要求实现的方法，具体过程就是如下代码中的 `reverse` 方法。该方法使用了上面提到的 `getAndRemoveLastElement` 方法。

```
public static void reverse(Stack<Integer> stack) {
    if (stack.isEmpty()) {
        return;
    }
    int i = getAndRemoveLastElement(stack);
    reverse(stack);
    stack.push(i);
}
```

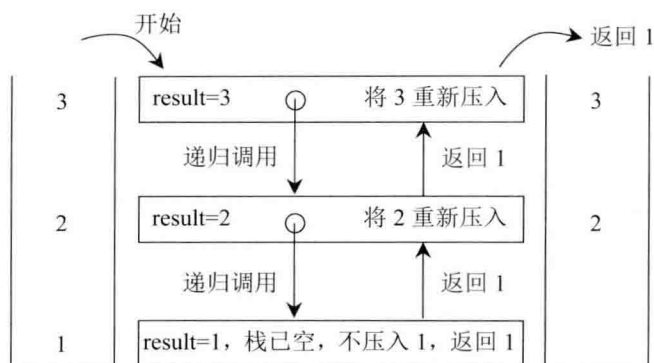


图 1-4

如果从 stack 的栈顶到栈底依次为 3、2、1，reverse 函数的具体过程如图 1-5 所示。

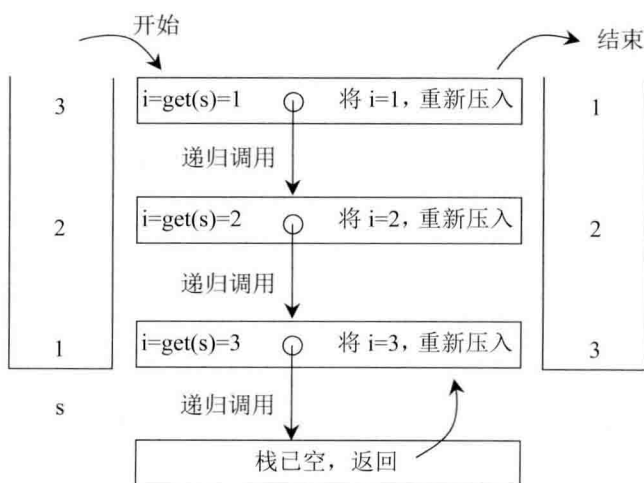


图 1-5

getAndRemoveLastElement 方法在图中简单表示为 get 方法，表示移除并返回当前栈底元素。