

# 高阶程序设计第二次报告

陈德创 19030500217

计算机科学与技术学院

日期：2020 年 4 月 16 日

## 目录

<b>1</b>	<b>总览</b>	<b>2</b>
<b>2</b>	<b>P1449、后缀表达式</b>	<b>2</b>
<b>3</b>	<b>P1996、约瑟夫问题</b>	<b>2</b>
<b>4</b>	<b>P1160、队列安排</b>	<b>2</b>
<b>5</b>	<b>P1540、机器翻译</b>	<b>2</b>
<b>6</b>	<b>P2058、海港</b>	<b>2</b>
<b>7</b>	<b>P1241、括号序列</b>	<b>3</b>
7.1	题目复述 . . . . .	3
7.2	解题思路 . . . . .	3
7.3	复杂度分析 . . . . .	3
7.4	结果 . . . . .	4
<b>8</b>	<b>P4387、验证栈序列</b>	<b>4</b>
<b>9</b>	<b>个人总结</b>	<b>4</b>

# 1 总览

✓	P1449	后缀表达式	模拟 字符串 线性结构 栈	普及-	<div><div></div></div>
✓	P1996	约瑟夫问题	模拟 树状数组 队列	普及-	<div><div></div></div>
✓	P1160	队列安排	模拟 线性结构 队列 构造	普及/提高-	<div><div></div></div>
✓	P1540	机器翻译	模拟 队列	普及-	<div><div></div></div>
✓	P2058	海港	模拟	普及/提高-	<div><div></div></div>
✓	P1241	括号序列	递推	普及+/提高	<div><div></div></div>
✓	P4387	【深基15.习9】验证栈序列	栈	普及/提高-	<div><div></div></div>

## 2 P1449、后缀表达式

算法标签：栈。表达式也是栈的经典应用了吧。

 LetMyself 04-10 23:37:41	Accepted 100	P1449 后缀表达式	21ms / 696.00KB / 684B C++
 LetMyself 04-10 23:37:25	Compile Error	P1449 后缀表达式	0ms / 0B / 666B C++

## 3 P1996、约瑟夫问题

算法标签：链表。约瑟夫问题，这个问题也很经典了，链表的入门题。

 LetMyself 04-10 23:52:00	Accepted 100	P1996 约瑟夫问题	16ms / 824.00KB / 535B C++
---	-----------------	-------------	----------------------------

## 4 P1160、队列安排

算法标签：栈。虽然说着是队列安排，但实际上这是一道链表题，而且很典型。

 LetMyself 04-12 20:57:42	Accepted 100	P1160 队列安排	227ms / 1.37MB / 830B C++
 LetMyself 04-12 20:56:25	Unaccepted 0	P1160 队列安排	366ms / 1.32MB / 865B C++

## 5 P1540、机器翻译

算法标签：队列。

 LetMyself 06-22 18:23:02	Accepted 100	P1540 机器翻译	29ms / 792.00KB / 268B C++
 LetMyself 06-22 18:05:25	Unaccepted 90	P1540 机器翻译	29ms / 860.00KB / 273B C++

## 6 P2058、海港

算法标签：队列。

 LetMyself 04-11 00:30:28	Accepted 100	P2058 海港	2.42s / 1.54MB / 985B C++
 LetMyself 04-11 00:18:30	Unaccepted 40	P2058 海港	2.02s / 1.40MB / 943B C++
 LetMyself 04-11 00:17:43	Compile Error	P2058 海港	0ms / 0B / 965B C++
 LetMyself 04-11 00:16:10	Compile Error	P2058 海港	0ms / 0B / 945B C++

## 7 P1241、括号序列

算法标签：模拟，栈。关于括号的题目描述也是栈的经典应用了吧。

### 7.1 题目复述

定义如下规则序列 (字符串)：

1. 空序列是规则序列；
2. 如果  $S$  是规则序列，那么  $(S)$  和  $[S]$  也是规则序列；
3. 如果  $A$  和  $B$  都是规则序列，那么  $AB$  也是规则序列。

例如，下面的字符串都是规则序列： $()$ ， $[]$ ， $(())$ ， $([])$ ， $[][]$ ， $()[()]$

而以下几个则不是： $($ ， $[$ ， $]$ ， $)()$ ， $(()$ ， $([]$

现在，给你一些由 ‘(’，‘)’，‘[’，‘]’ 构成的序列，你要做的，是补全该括号序列，即扫描一遍原序列，对每一个右括号，找到在它左边最靠近它的左括号匹配，如果没有就放弃。在以这种方式把原序列匹配完成后，把剩下的未匹配的括号补全。

### 7.2 解题思路

这题我 WA 了好多次，主要是对题目说的匹配了解不对。我一开始以为是根据给定序列，求出一个序列包含给定序列且是规则序列，那这样这题就是一道构造题了 (难度也大大下降)。

于是我一直 WA。这题实际上要求我们先匹配，匹配完了再去补全。如何匹配呢？我们用一个栈，然后遍历整个括号序列  $s$ ，如果  $s[i] == '['$  或者  $s[i] == '('$ ，那我们将它入栈。如果  $s[i] == ')'$  或者  $s[i] == ']'$ ，那我们检查栈顶元素和  $s[i]$  是否匹配，如果匹配，那我们给它做上成功匹配的

标志并弹栈，那么输出的时候直接输出就可以了。如果不成功匹配，那就做匹配失败标记，代表着我们在输出的时候需要匹配输出，比如']' 如果不成功匹配，那么就要输出"["。

算法的正确性是容易想的，我们栈维护的其实是一个没有被匹配的序列，那么显然栈顶元素就是最靠近  $s[i]$  的左括号。在具体实现的时候，要注意空栈的判断，否则应该是 RE 的。

### 7.3 复杂度分析

时间复杂度: $O(n)$ ，空间复杂度: $O(n)$ ，这题应该是没有在线做法的吧，有的话也得存栈，所以空间复杂度为  $O(n)$  没问题。

### 7.4 结果

图 1: 一定要注重细节

 LetMyself 12-24 16:39:03	Accepted 100	P1241 括号序列	41ms / 828.00KB / 640B C++
 LetMyself 12-24 16:37:22	Unaccepted 53	P1241 括号序列	23ms / 784.00KB / 612B C++
 LetMyself 12-24 16:25:05	Unaccepted 62	P1241 括号序列	30ms / 788.00KB / 654B C++
 LetMyself 12-24 16:23:51	Unaccepted 62	P1241 括号序列	41ms / 788.00KB / 654B C++
 LetMyself 12-24 16:17:50	Unaccepted 56	P1241 括号序列	42ms / 792.00KB / 647B C++

## 8 P4387、验证栈序列

算法标签：栈。就是火车入栈，上课讲过这个。

 LetMyself 04-12 21:20:01	Accepted 100	P4387 【深基15.习9】验证栈序列	680ms / 1.66MB / 653B C++
 LetMyself 04-12 21:17:54	Unaccepted 0	P4387 【深基15.习9】验证栈序列	582ms / 1.68MB / 640B C++
 LetMyself 04-12 21:14:37	Unaccepted 0	P4387 【深基15.习9】验证栈序列	642ms / 1.64MB / 638B C++

## 9 个人总结

这次做起来感觉顺多了，感觉刷题还是要一直刷，一停下来就容易手生。说实话刷题还是有好处的，只是今天和昨天差的太少了，难以察觉，直到某一天突然回头看看才发现原来已经进步这么多了。希望可以越来越顺吧。