

# Assignment #4: 排序、栈、队列和树

Updated 0005 GMT+8 March 11, 2024

2024 spring, Compiled by 李鹏辉, 元培学院

## 说明:

1) The complete process to learn DSA from scratch can be broken into 4 parts:

Learn about Time complexities, learn the basics of individual Data Structures, learn the basics of Algorithms, and practice Problems.

2) 请把每个题目解题思路（可选），源码Python, 或者C++（已经在Codeforces/Openjudge上AC），截图（包含Accepted），填写到下面作业模版中（推荐使用 typora <https://typoraio.cn>，或者用 word）。AC 或者没有AC，都请标上每个题目大致花费时间。

3) 提交时候先提交pdf文件，再把md或者doc文件上传到右侧“作业评论”。Canvas需要有同学清晰头像、提交文件有pdf、“作业评论”区有上传的md或者doc附件。

4) 如果不能在截止前提交作业，请写明原因。

## 编程环境

Windows 10 Home, PyCharm 2022.3.2 (Community Edition)

操作系统: macOS Ventura 13.4.1 (c)

Python编程环境: Spyder IDE 5.2.2, PyCharm 2023.1.4 (Professional Edition)

C/C++编程环境: Mac terminal vi (version 9.0.1424), g++/gcc (Apple clang version 14.0.3, clang-1403.0.22.14.1)

## 1. 题目

### 05902: 双端队列

<http://cs101.openjudge.cn/practice/05902/>

思路: 10分钟。

代码

```
1 def q1():
2     for _ in range(int(input())):
3         s = []
4         for _ in range(int(input())):
5             op, num = tuple(map(int, input().split()))
6             if op == 1:
```

```

7         s.append(num)
8     elif s:
9         if num == 0:
10            s.pop(0)
11        else:
12            s.pop()
13    else:
14        break
15    if s:
16        print(' '.join(map(str, s)))
17    else:
18        print('NULL')
19
20
21 q1()

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```

def q1():
    for _ in range(int(input())):
        s = []
        for _ in range(int(input())):
            op, num = tuple(map(int, input().split()))
            if op == 1:
                s.append(num)
            elif s:

```

基本信息

#: 44296457  
 题目: 05902  
 提交人: 2100017777\_李鹏辉  
 (2100017777)  
 内存: 3760kB  
 时间: 39ms  
 语言: Python3  
 提交时间: 2024-03-19 13:04:30

## 02694: 波兰表达式

<http://cs101.openjudge.cn/practice/02694/>

思路: 15分钟。

代码

```

1  import operator as op
2
3
4  def q2():
5      chars = input().split()
6      s = []
7      op_d = {'+': op.add, '-': op.sub, '*': op.mul, '/': op.truediv}
8      for char in chars:
9          if char in op_d:
10             s.append(char)
11        else:
12            s.append(float(char))
13            while len(s) > 2 and s[-2] not in op_d:
14                second = s.pop()
15                first = s.pop()

```

```

16         s.append(op_d[s.pop()](first, second))
17     return s[0]
18
19
20 print(format(q2(), '.6f'))

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```

import operator as op

def q2():
    chars = input().split()
    s = []
    op_d = {'+': op.add, '-': op.sub, '*': op.mul, '/': op.truediv}

```

基本信息

#: 44301329  
 题目: 02694  
 提交人: 2100017777\_李鹏辉  
 (2100017777)  
 内存: 3640kB  
 时间: 23ms  
 语言: Python3  
 提交时间: 2024-03-19 17:56:41

## 24591: 中序表达式转后序表达式

<http://cs101.openjudge.cn/practice/24591/>

思路: 改良了一下课件上的调度场, 数字不需要转换成int或float。一小时。

代码

```

1 def q3():
2     prec = {'+':1, '-':1, '*':2, '/':2}
3
4     for _ in range(int(input())):
5         infix = input()
6         s = []
7         re = []
8         num = ''
9
10        for char in infix:
11            if char.isnumeric() or char == '.':
12                num += char
13            else:
14                if num:
15                    re.append(num)
16                    num = ''
17                if char == '(':
18                    s.append(char)
19                if char == ')':
20                    while s and s[-1] != '(':
21                        re.append(s.pop())
22                    s.pop()
23                if char in '+-*/':
24                    while s and s[-1] in '+-*/' and prec[char] <=
prec[s[-1]]:

```

```

25         re.append(s.pop())
26         s.append(char)
27
28     if num:
29         re.append(num)
30     while s:
31         re.append(s.pop())
32     print(' '.join(str(x) for x in re))
33
34
35 q3()

```

代码运行截图 (AC代码截图, 至少包含有"Accepted")

状态: Accepted

源代码

```

def q3():
    prec = {'+':1, '-':1, '*':2, '/':2}

    for _ in range(int(input())):
        infix = input()
        s = []

```

基本信息

#: 44307090  
 题目: 24591  
 提交人: 2100017777  
 内存: 3676kB  
 时间: 27ms  
 语言: Python3  
 提交时间: 2024-03-19 23:02:14

## 22068: 合法出栈序列

<http://cs101.openjudge.cn/practice/22068/>

思路: 主要改变输入的字符串去匹配原来的字符串, 40分钟。

代码

```

1  def q4():
2      chars = input()
3      while True:
4          try:
5              wait = input()
6              if len(wait) != len(chars):
7                  print('NO')
8                  continue
9              c_r = chars[:] # chars_remained
10             fixed = ''
11             while wait:
12                 f_l = wait[0] # first_letter
13                 if f_l not in chars:
14                     print('NO')
15                     break
16                 elif f_l not in fixed:
17                     f_i = c_r.index(f_l) # first_index
18                     fixed += c_r[:f_i]
19                     c_r = c_r[f_i+1:]

```

```

20         elif f_l == fixed[-1]:
21             fixed = fixed[:-1]
22         else:
23             print('NO')
24             break
25         wait = wait[1:]
26         if not wait:
27             print('YES')
28     except EOFError:
29         break
30
31
32 q4()

```

代码运行截图 (AC代码截图, 至少包含有"Accepted")

## 06646: 二叉树的深度

<http://cs101.openjudge.cn/practice/06646/>

思路: bfs. 15分钟。

代码

```

1  def q5():
2      t_d = {} # tree_dict
3      for _ in range(int(input())):
4          t_d[_+1] = list(map(int, input().split()))
5      m_d = 0 # max_depth
6      n_q = [1] # node_queue
7      while n_q:
8          m_d += 1
9          n_q_c = n_q[:] # n_q_copy
10         n_q = []
11         for node in n_q_c:
12             for leave in t_d[node]:
13                 if leave != -1:
14                     n_q.append(leave)
15     return m_d
16
17
18 print(q5())

```

代码运行截图 (AC代码截图, 至少包含有"Accepted")

状态: Accepted

源代码

```
def q5():
    t_d = {} # tree_dict
    for _ in range(int(input())):
        t_d[_+1] = list(map(int, input().split()))
    m_d = 0 # max_depth
    n_q = [1] # node_queue
    while n_q:
```

基本信息

#: 44303581  
题目: 06646  
提交人: 2100017777\_李鹏辉  
(2100017777)  
内存: 3636kB  
时间: 22ms  
语言: Python3  
提交时间: 2024-03-19 20:06:21

## 02299: Ultra-QuickSort

<http://cs101.openjudge.cn/practice/02299/>

思路：一开始只用sorted排序，后来发现超时。经提醒改用merge sort和栈，最后总计约两小时。

代码

```
1 def q6():
2
3     while True:
4         size = int(input())
5         if size == 0:
6             break
7         numbers = [int(input()) for _ in range(size)]
8         s = [numbers[:size//2], numbers[size//2:]]
9         result = []
10        times = 0
11
12        def merge(l1, l2):
13            nonlocal times
14            re = []
15            while l1 and l2:
16                if l1[0] <= l2[0]:
17                    re.append(l1.pop(0))
18                else:
19                    re.append(l2.pop(0))
20                    times += len(l1)
21            if l1:
22                re += l1
23            if l2:
24                re += l2
25            return re
26
27        while s:
28            top = s.pop()
29            t_s = len(top)
30            if t_s > 1:
31                s.append(top[:t_s//2])
32                s.append(top[t_s//2:])
33            else:
34                c_l = top
```

```
35         while result and len(result[-1]) == len(c_1):
36             c_1 = merge(result.pop(), c_1)
37             result.append(c_1)
38     while len(result) > 1:
39         l2 = result.pop()
40         l1 = result.pop()
41         result.append(merge(l1, l2))
42     print((size * size - size) // 2 - times)
43
44
45 q6()
```

代码运行截图 (AC代码截图, 至少包含有"Accepted")

状态: Accepted

源代码

```
def q6():
    while True:
        size = int(input())
        if size == 0:
            break
```

基本信息

#: 44306130  
题目: 02299  
提交人: 2100017777  
内存: 35396kB  
时间: 40672ms  
语言: Python3  
提交时间: 2024-03-19 22:09:59

## 2. 学习总结和收获

如果作业题目简单, 有否额外练习题目, 比如: OJ"2024spring每日选做"、CF、LeetCode、洛谷等网站题目。

第四题其实如果换种思路, 改成通过改变原来的字符串去匹配新输入的字符串, 会方便很多。第六题应该对大数足够敏感到能意识到要用归并。