Assignment #A: 图论：算法，树算及栈

Updated 2018 GMT+8 Apr 21, 2024

2024 spring, Complied by 李鹏辉，元培学院

说明：

1）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。

2）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。

3）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

20743: 整人的提词本

http://cs101.openjudge.cn/practice/20743/

思路：about 25 mins.

代码

```
def q1():
    os = input()  # original string
    s = []
    cw = ''  # current word
    for _, l in enumerate(os):
        if l == '(':
            if cw:
                s.append(cw)
                cw = ''
            s.append(l)
        elif l == ')':
```

```
            s.append(cw)
            w = ''
            while s[-1] != '(':
                w += s.pop()[::-1]
            s.pop()
            cw = w
        else:
            cw += l
    if cw:
        s.append(cw)
    print(''.join(s))


q1()
```

代码运行截图 （至少包含有"Accepted"）

状态: Accepted

源代码

```
def q1():
    os = input()  # original string
    s = []
    cw = ''  # current word
    for _, l in enumerate(os):
        if l == '(':
```

基本信息

#: 44836361
题目: 20743
提交人: 2100017777
内存: 3604kB
时间: 21ms
语言: Python3
提交时间: 2024-04-30 16:12:56

02255: 重建二叉树

http://cs101.openjudge.cn/practice/02255/

思路：14 mins.

代码

```
class BTree:
    def __init__(self, node):
        self.node = node
        self.left = None
        self.right = None


def q2():

    def bt(preo, ino):
        if not preo:
            return None
        root = preo[0]
        ri = ino.index(root)
        lino = ino[:ri]
        rino = ino[ri+1:]
        ll = len(lino)  # left length
```

```
        lpreo = preo[1:1+ll]
        rpreo = preo[1+ll:]
        t = BTree(root)
        t.left = bt(lpreo, lino)
        t.right = bt(rpreo, rino)
        return t

    def posto(t):
        re = ''
        if t.left:
            re += posto(t.left)
        if t.right:
            re += posto(t.right)
        re += t.node
        return re

    while True:
        try:
            preo, ino = input().split()
            print(posto(bt(preo, ino)))
        except EOFError:
            return


q2()
```

代码运行截图 （至少包含有"Accepted"）

状态: Accepted

源代码

```
class BTree:
    def __init__(self, node):
        self.node = node
        self.left = None
        self.right = None
```

基本信息

#: 44836413
题目: 02255
提交人: 2100017777
内存: 3620kB
时间: 24ms
语言: Python3
提交时间: 2024-04-30 16:26:24

01426: Find The Multiple

http://cs101.openjudge.cn/practice/01426/

要求用bfs实现

思路：about 30 mins.

代码

```
from collections import deque


def q3():
```

```
    while True:
        q = deque([1])
        n = int(input())
        if n == 0:
            return
        while True:
            c = q.popleft()
            if c % n == 0:
                print(c)
                break
            else:
                q.append(c * 10)
                q.append(c * 10 + 1)


q3()
```

代码运行截图 （AC代码截图，至少包含有"Accepted"）

状态: Accepted

源代码

```
from collections import deque


def q3():
    while True:
        q = deque([1])
```

基本信息

#: 44836651
题目: 01426
提交人: 2100017777
内存: 32828kB
时间: 622ms
语言: Python3
提交时间: 2024-04-30 17:03:37

04115: 鸣人和佐助

bfs, http://cs101.openjudge.cn/practice/04115/

思路：about 1 hr.

代码

```
from collections import deque


def q4():
    m, n, t = map(int, input().split())
    graph = []
    for x in range(m):
        rs = input()  # raw string
        graph.append(rs)
        for y, l in enumerate(rs):
            if l == '@':
                s = [x, y]
            if l == '+':
                d = [x, y]

```

```
    v = [[-1] * n for _ in range(m)]
    q = deque([[s, 0, t]])
    while q:
        cp, time, hits = q.popleft()
        if v[cp[0]][cp[1]] < hits:
            v[cp[0]][cp[1]] = hits
        else:
            continue
        if cp == d:
            print(time)
            return

        dirs = [[1, 0], [-1, 0], [0, 1], [0, -1]]
        ori_nps = [[cp[0]+ix, cp[1]+iy] for ix, iy in dirs]
        nps = []
        for npx, npy in ori_nps:
            if npx >= m or npx < 0 or npy >= n or npy < 0 or [npx, npy] in
v:
                continue
            nps.append([npx, npy])
        for np in nps:
            if graph[np[0]][np[1]] == '#':
                if hits > 0:
                    q.append([np, time+1, hits-1])
            else:
                    q.append([np, time+1, hits])

    print(-1)
    return


q4()
```

代码运行截图 （AC代码截图，至少包含有"Accepted"）

状态：Accepted

源代码

```
from collections import deque


def q4():
    m, n, t = map(int, input().split())
    graph = []
```

基本信息

#: 44838530
题目: 04115
提交人: 2100017777
内存: 3964kB
时间: 237ms
语言: Python3
提交时间: 2024-04-30 22:54:33

20106: 走山路

Dijkstra, http://cs101.openjudge.cn/practice/20106/

思路：时间不够了.....

代码

```
#

```

代码运行截图 （AC代码截图，至少包含有"Accepted"）

05442: 兔子与星空

Prim, http://cs101.openjudge.cn/practice/05442/

思路：时间不够了.....

代码

```
#

```

代码运行截图 （AC代码截图，至少包含有"Accepted"）

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。

粗略看了一眼题目，后面两道题实在赶不及在ddl前做完了，很抱歉只能先交四道保证不迟交，剩余两道题我会后面自己补做完，不会浪费这两道好题的。