Assignment #A: 图论：算法，树算及栈

Updated 2018 GMT+8 Apr 21, 2024

2024 spring, Complied by 尹柚鑫 2100015878 光华管理学院

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

==（请改为同学的操作系统、编程环境等）==

操作系统：Win11

Python编程环境：jupter notebook

1. 题目

20743: 整人的提词本

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

耗时：10mins

思路：用栈的思想可以轻松做出

代码

```
#
string=input()
stack=[]
strlist=list(string)
temp=[]
for i in range(len(strlist)):
    if strlist[i] !=')':
        stack.append(strlist[i])
    else:
        while stack[-1]!='(':
            temp.append(stack.pop())
        stack.pop()
        stack.extend(temp)
        temp=[]
```

```
    if i==len(strlist)-1:
        print(''.join(stack))
```

代码运行截图 ==（至少包含有"Accepted"）==

状态: Accepted

源代码

```
string=input()
stack=[]
strlist=list(string)
temp=[]
for i in range(len(strlist)):
    if strlist[i] !=')':
        stack.append(strlist[i])
    else:
        while stack[-1]!='(':
            temp.append(stack.pop())
        stack.pop()
        stack.extend(temp)
        temp=[]
    if i==len(strlist)-1:
        print(''.join(stack))
```

基本信息

#: 44835962
题目: 20743
提交人: 尹柚鑫(2100015878)
内存: 3620kB
时间: 21ms
语言: Python3
提交时间: 2024-04-30 14:39:45

02255: 重建二叉树

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

耗时：30mins

思路：和之前的一道作业题类似，我之前的做法是比较繁琐的，很多个函数在一起，规规矩矩完成，这次参考了答案，答案递归得到了左右子树，非常巧妙简洁

代码

```
#
def build_tree(preorder, inorder):
    if not preorder:
        return ''

    root = preorder[0]
    root_index = inorder.index(root)

    left_preorder = preorder[1:1 + root_index]
    right_preorder = preorder[1 + root_index:]

    left_inorder = inorder[:root_index]
    right_inorder = inorder[root_index + 1:]

    left_tree = build_tree(left_preorder, left_inorder)
    right_tree = build_tree(right_preorder, right_inorder)

    return left_tree + right_tree + root

while True:
    try:
        preorder, inorder = input().split()
```

```
        postorder = build_tree(preorder, inorder)
        print(postorder)
    except EOFError:
        break
```

代码运行截图 ==（至少包含有"Accepted"）==

01426: Find The Multiple

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

要求用bfs实现

思路：参考了答案，这题居然可以用BFS，第一眼看上去像是面试的数学题，这里是对余数进行BFS

代码

```
#
from collections import deque

def find_multiple(n):
    # 使用队列实现BFS
    q = deque()
    # 初始化队列，存储的是(模n值，对应的数字字符串)
    q.append((1 % n, "1"))
    visited = set([1 % n])  # 用于记录访问过的模n值，避免重复搜索

    while q:
        mod, num_str = q.popleft()

        # 检查当前模n值是否为0，是则找到答案
        if mod == 0:
            return num_str

        # 尝试在当前数字后加0或加1，生成新的数字，并计算模n值
        for digit in ["0", "1"]:
            new_num_str = num_str + digit
            new_mod = (mod * 10 + int(digit)) % n
```

```
            # 如果新模n值未访问过，则加入队列继续搜索
            if new_mod not in visited:
                q.append((new_mod, new_num_str))
                visited.add(new_mod)

def main():
    while True:
        n = int(input())
        if n == 0:
            break
        print(find_multiple(n))

if __name__ == "__main__":
    main()
```

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 find_multiple(n):
    # 使用队列实现BFS
    q = deque()
    # 初始化队列，存储的是(模n值，对应的数字字符串)
    q.append((1 % n, "1"))
    visited = set([1 % n])  # 用于记录访问过的模n值，避免重复搜索

    while q:
        mod, num_str = q.popleft()
```

基本信息

#: 44838700
题目: 01426
提交人: 尹柚鑫(2100015878)
内存: 3596kB
时间: 43ms
语言: Python3
提交时间: 2024-04-30 23:42:16

04115: 鸣人和佐助

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

思路：与上个题目相比，这个题很像是一个BFS问题，只不过有点复杂，需要仔细维护查克拉数量

代码

```
#
from collections import deque

M, N, T = map(int, input().split())
graph = [list(input()) for i in range(M)]
direc = [(0,1), (1,0), (-1,0), (0,-1)]
start, end = None, None
for i in range(M):
    for j in range(N):
        if graph[i][j] == '@':
            start = (i, j)
def bfs():
    q = deque([start + (T, 0)])
    visited = [[-1]*N for i in range(M)]
```

```
    visited[start[0]][start[1]] = T
    while q:
        x, y, t, time = q.popleft()
        time += 1
        for dx, dy in direc:
            if 0<=x+dx<M and 0<=y+dy<N:
                if (elem := graph[x+dx][y+dy]) == '*' and t > visited[x+dx]
[y+dy]:
                    visited[x+dx][y+dy] = t
                    q.append((x+dx, y+dy, t, time))
                elif elem == '#' and t > 0 and t-1 > visited[x+dx][y+dy]:
                    visited[x+dx][y+dy] = t-1
                    q.append((x+dx, y+dy, t-1, time))
                elif elem == '+':
                    return time
    return -1
print(bfs())
```

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

M, N, T = map(int, input().split())
graph = [list(input()) for i in range(M)]
direc = [(0,1), (1,0), (-1,0), (0,-1)]
start, end = None, None
for i in range(M):
    for j in range(N):
```

基本信息

#:	44838742
题目:	04115
提交人:	尹柚鑫(2100015878)
内存:	4104kB
时间:	65ms
语言:	Python3
提交时间:	2024-04-30 23:52:26

20106: 走山路

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

思路:

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：

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。==

01426: Find The Multiple 这个题目可以多想想，一开始并没有意识到这个是BFS的题目

五一假期，诸事繁多，没有很好规划时间，这次的作业很多都是先看答案，再理解答案的思路，下面要迅速补齐