AssignmentA

姓名：尹廖丹华

学号：2000092113

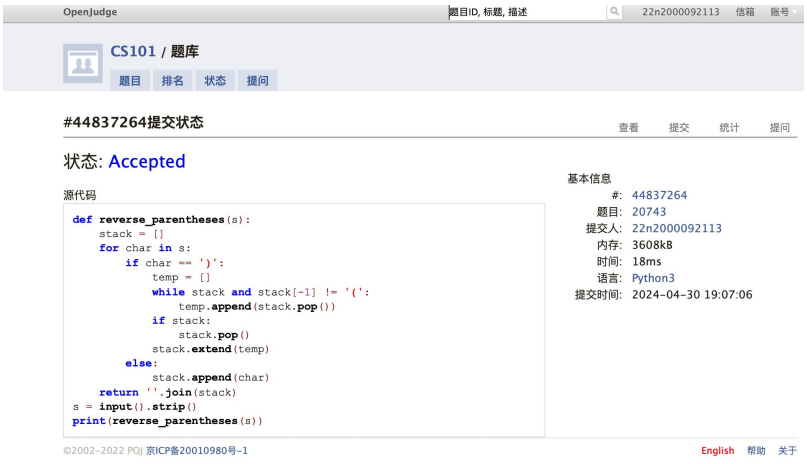
院系：心理与认知科学学院

一、 作业题目

1、20743: 整人的提词本（1h）

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

```
def reverse_parentheses(s):
    stack = []
    for char in s:
        if char == ')':
            temp = []
            while stack and stack[-1] != '(':
                temp.append(stack.pop())
            if stack:
                stack.pop()
            stack.extend(temp)
        else:
            stack.append(char)
    return ''.join(stack)
s = input().strip()
print(reverse_parentheses(s))
```

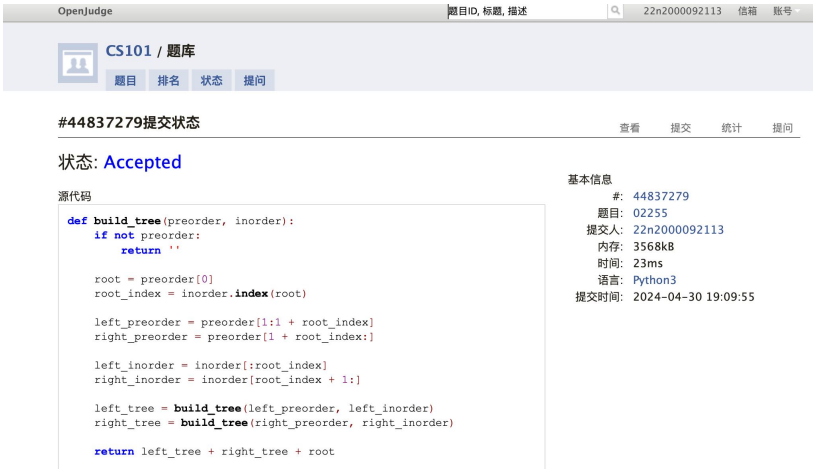


2、02255: 重建二叉树（1h35min）

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

```
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
```

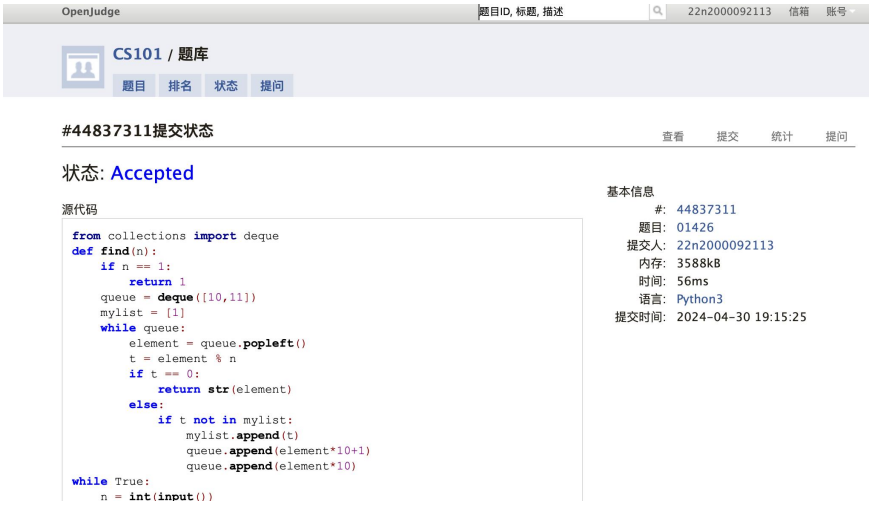


3、01426: Find The Multiple（2h40min）

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

```
from collections import deque
def find(n):
    if n == 1:
        return 1
    queue = deque([10,11])
    mylist = [1]
    while queue:
        element = queue.popleft()
        t = element % n
        if t == 0:
            return str(element)
        else:
            if t not in mylist:
                mylist.append(t)
```

```
                queue.append(element*10+1)
                queue.append(element*10)
while True:
    n = int(input())
    if n == 0:
        break
    else:
        print(find(n))
```

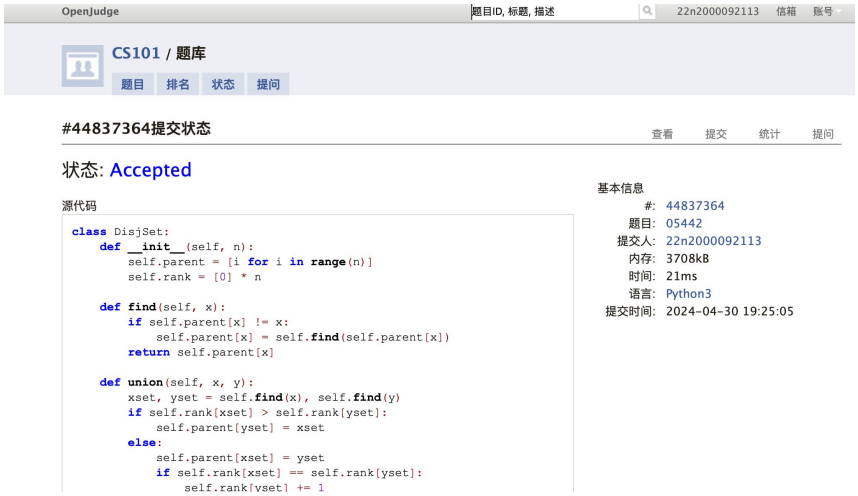


4、05442: 兔子与星空（3h30min）

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

```
class DisjSet:
    def __init__(self, n):
        self.parent = [i for i in range(n)]
        self.rank = [0] * n

    def find(self, x):
        if self.parent[x] != x:
            self.parent[x] = self.find(self.parent[x])
        return self.parent[x]

    def union(self, x, y):
        xset, yset = self.find(x), self.find(y)
        if self.rank[xset] > self.rank[yset]:
            self.parent[yset] = xset
        else:
            self.parent[xset] = yset
            if self.rank[xset] == self.rank[yset]:
                self.rank[yset] += 1

def kruskal(n, edges):
    dset = DisjSet(n)
    edges.sort(key=lambda x: x[2])
```

```
        sol = 0
        for u, v, w in edges:
            u, v = ord(u) - 65, ord(v) - 65
            if dset.find(u) != dset.find(v):
                dset.union(u, v)
                sol += w
        if len(set(dset.find(i) for i in range(n))) > 1:
            return -1
        return sol

n = int(input())
edges = []
for _ in range(n - 1):
    arr = input().split()
    root, m = arr[0], int(arr[1])
    for i in range(m):
        edges.append((root, arr[2 + 2 * i], int(arr[3 + 2 * i])))
print(kruskal(n, edges))
```



二、 学习总结和收获

每周都是不断地赶进度，十分焦虑了属于是，每次都看着同学们热烈讨论，但我还在看题解挣扎半天，太难了数算！！