Assignment #D: May月考

Updated 1654 GMT+8 May 8, 2024

2024 spring, Complied by ==同学的姓名、院系==

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：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

02808: 校门外的树

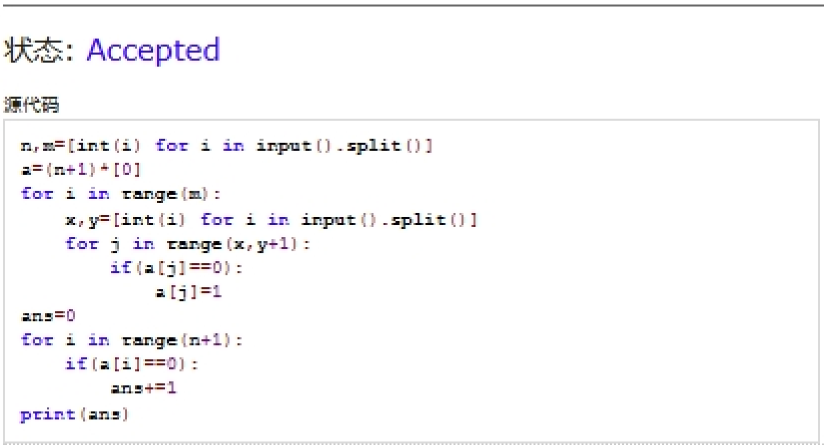
http://cs101.openjudge.cn/practice/02808/

思路：

代码

```
#
n,m=[int(i) for i in input().split()]
a=(n+1)*[0]
for i in range(m):
    x,y=[int(i) for i in input().split()]
    for j in range(x,y+1):
        if(a[j]==0):
            a[j]=1
ans=0
for i in range(n+1):
    if(a[i]==0):
        ans+=1
print(ans)
```

代码运行截图 ==（至少包含有"Accepted"）==



状态: Accepted

源代码

```
n,m=[int(i) for i in input().split()]
a=(n+1)*[0]
for i in range(m):
    x,y=[int(i) for i in input().split()]
    for j in range(x,y+1):
        if(a[j]==0):
            a[j]=1
ans=0
for i in range(n+1):
    if(a[i]==0):
        ans+=1
print(ans)
```

20449: 是否被5整除

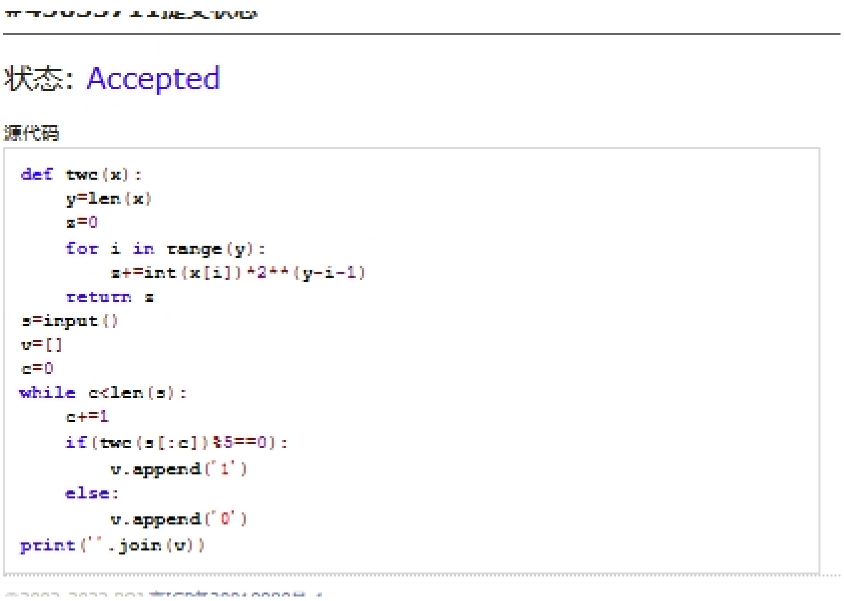
http://cs101.openjudge.cn/practice/20449/

思路：

代码

```
#
def twc(x):
    y=len(x)
    z=0
    for i in range(y):
        z+=int(x[i])*2**(y-i-1)
    return z
s=input()
v=[]
c=0
while c<len(s):
    c+=1
    if(twc(s[:c])%5==0):
        v.append('1')
    else:
        v.append('0')
print(''.join(v))
```

代码运行截图 ==（至少包含有"Accepted"）==



状态: Accepted

源代码

```
def twc(x):
    y=len(x)
    s=0
    for i in range(y):
        s+=int(x[i])*2**(y-i-1)
    return s
s=input()
v=[]
c=0
while c<len(s):
    c+=1
    if(twc(s[:c])%5==0):
        v.append('1')
    else:
        v.append('0')
print(''.join(v))
```

01258: Agri-Net

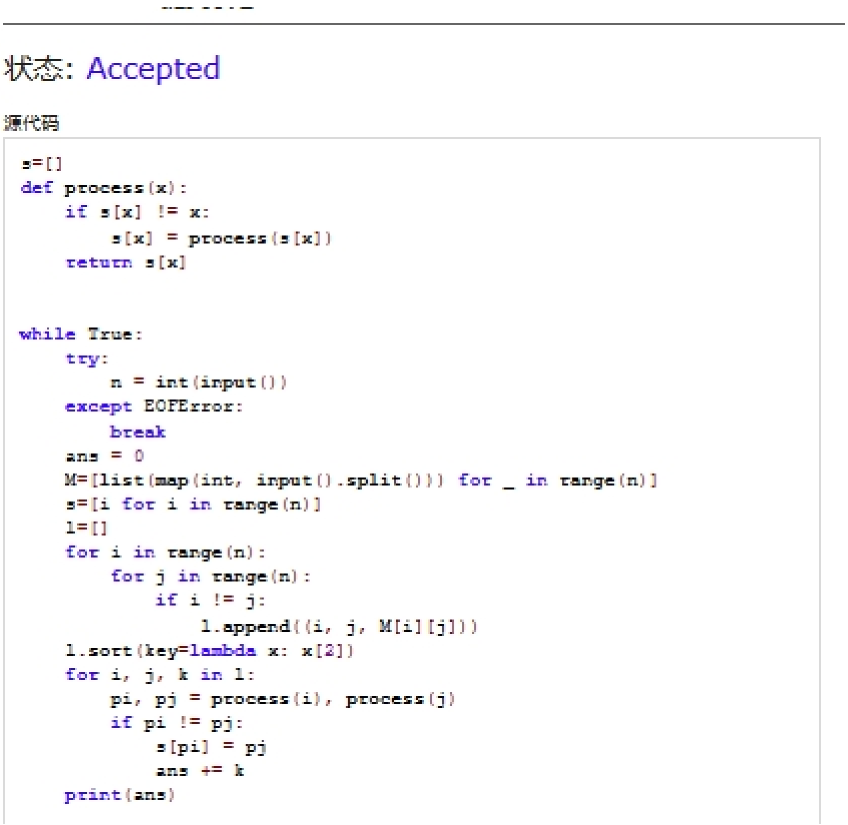
http://cs101.openjudge.cn/practice/01258/

思路:

代码

```
# 
s=[]
def process(x):
    if s[x] != x:
        s[x] = process(s[x])
    return s[x]


while True:
    try:
        n = int(input())
    except EOFError:
        break
    ans = 0
    M=[list(map(int, input().split())) for _ in range(n)]
    s=[i for i in range(n)]
    l=[]
    for i in range(n):
        for j in range(n):
            if i != j:
                l.append((i, j, M[i][j]))
    l.sort(key=lambda x: x[2])
    for i, j, k in l:
        pi, pj = process(i), process(j)
        if pi != pj:
            s[pi] = pj
            ans += k
    print(ans)
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==



状态: Accepted

源代码

```
s=[]
def process(x):
    if s[x] != x:
        s[x] = process(s[x])
    return s[x]


while True:
    try:
        n = int(input())
    except EOFError:
        break
    ans = 0
    M=[list(map(int, input().split())) for _ in range(n)]
    s=[i for i in range(n)]
    l=[]
    for i in range(n):
        for j in range(n):
            if i != j:
                l.append((i, j, M[i][j]))
    l.sort(key=lambda x: x[2])
    for i, j, k in l:
        pi, pj = process(i), process(j)
        if pi != pj:
            s[pi] = pj
            ans += k
    print(ans)
```

27635: 判断无向图是否连通有无回路(同23163)

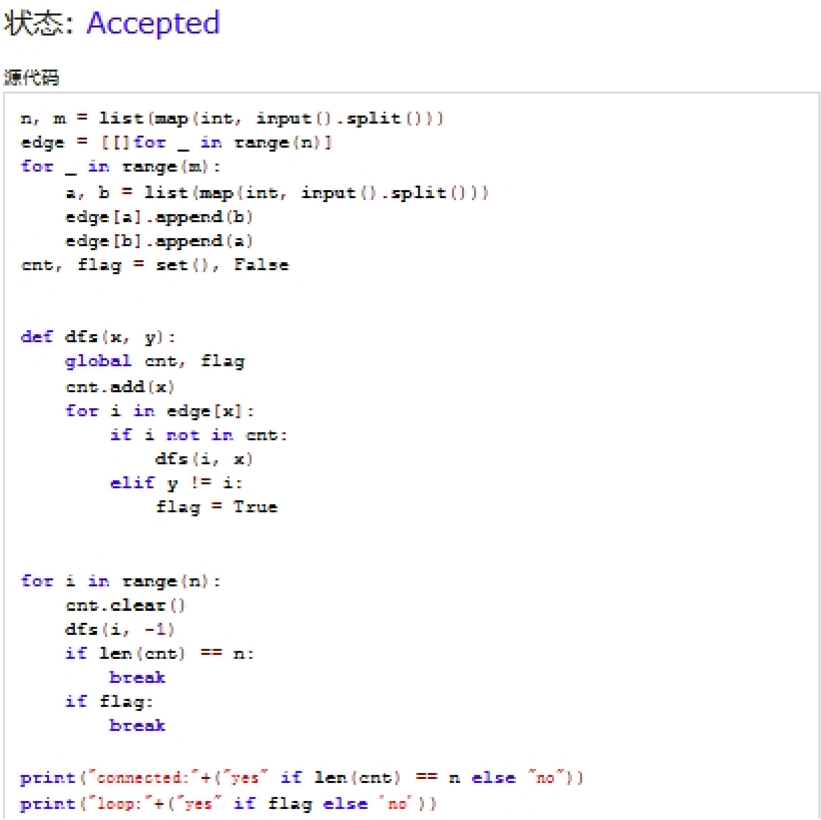
http://cs101.openjudge.cn/practice/27635/

思路：

代码

```
#
n, m = list(map(int, input().split()))
edge = [[]for _ in range(n)]
for _ in range(m):
    a, b = list(map(int, input().split()))
    edge[a].append(b)
    edge[b].append(a)
cnt, flag = set(), False


def dfs(x, y):
    global cnt, flag
    cnt.add(x)
    for i in edge[x]:
        if i not in cnt:
            dfs(i, x)
        elif y != i:
            flag = True


for i in range(n):
    cnt.clear()
    dfs(i, -1)
    if len(cnt) == n:
        break
    if flag:
        break

print("connected:"+("yes" if len(cnt) == n else "no"))
print("loop:"+("yes" if flag else 'no'))
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==



状态: Accepted

源代码

```
n, m = list(map(int, input().split()))
edge = [[]for _ in range(n)]
for _ in range(m):
    a, b = list(map(int, input().split()))
    edge[a].append(b)
    edge[b].append(a)
cnt, flag = set(), False


def dfs(x, y):
    global cnt, flag
    cnt.add(x)
    for i in edge[x]:
        if i not in cnt:
            dfs(i, x)
        elif y != i:
            flag = True


for i in range(n):
    cnt.clear()
    dfs(i, -1)
    if len(cnt) == n:
        break
    if flag:
        break

print("connected:"+("yes" if len(cnt) == n else "no"))
print("loop:"+("yes" if flag else 'no'))
```

27947: 动态中位数

http://cs101.openjudge.cn/practice/27947/

思路：

代码

```
#
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==

28190: 奶牛排队

http://cs101.openjudge.cn/practice/28190/

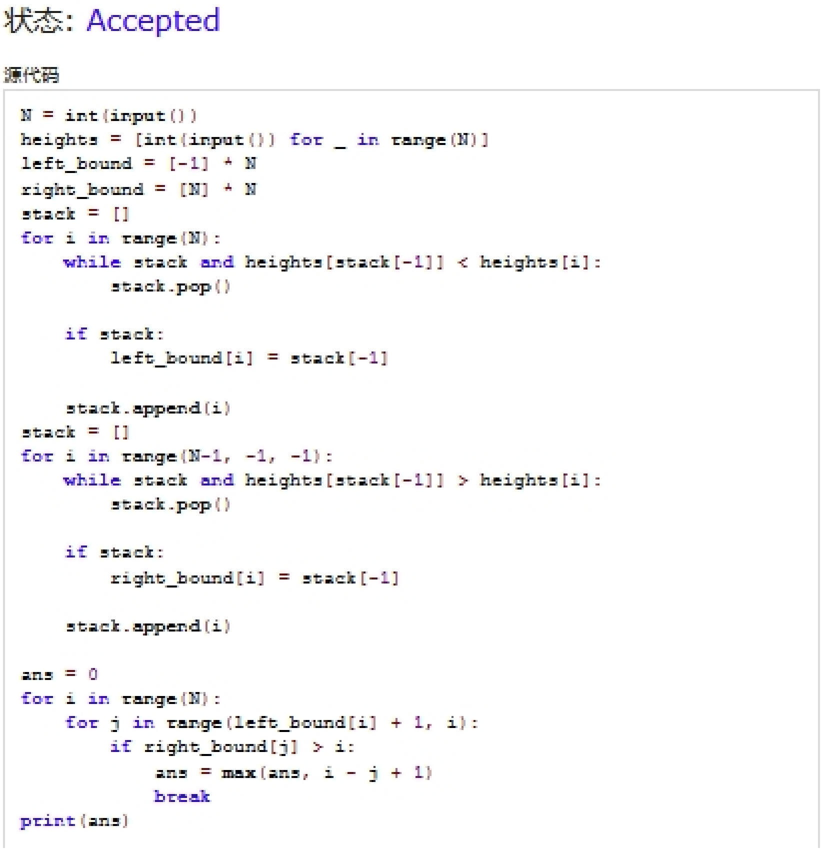
思路：

代码

```
#
N = int(input())
heights = [int(input()) for _ in range(N)]
left_bound = [-1] * N
right_bound = [N] * N
stack = []
for i in range(N):
    while stack and heights[stack[-1]] < heights[i]:
        stack.pop()

    if stack:
        left_bound[i] = stack[-1]

    stack.append(i)
stack = []
for i in range(N-1, -1, -1):
    while stack and heights[stack[-1]] > heights[i]:
        stack.pop()

    if stack:
        right_bound[i] = stack[-1]

    stack.append(i)
```

```python
ans = 0
for i in range(N):
    for j in range(left_bound[i] + 1, i):
        if right_bound[j] > i:
            ans = max(ans, i - j + 1)
            break
print(ans)
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==



状态: Accepted

源代码

```python
N = int(input())
heights = [int(input()) for _ in range(N)]
left_bound = [-1] * N
right_bound = [N] * N
stack = []
for i in range(N):
    while stack and heights[stack[-1]] < heights[i]:
        stack.pop()

    if stack:
        left_bound[i] = stack[-1]

    stack.append(i)
stack = []
for i in range(N-1, -1, -1):
    while stack and heights[stack[-1]] > heights[i]:
        stack.pop()

    if stack:
        right_bound[i] = stack[-1]

    stack.append(i)

ans = 0
for i in range(N):
    for j in range(left_bound[i] + 1, i):
        if right_bound[j] > i:
            ans = max(ans, i - j + 1)
            break
print(ans)
```

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。==

感觉模版题挺多的...但是我还是不太会用堆TT