Assignment #F: All-Killed 满分

Updated 1844 GMT+8 May 20, 2024

2024 spring, Complied by 潘子轩、信科

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

22485: 升空的焰火，从侧面看

http://cs101.openjudge.cn/practice/22485/

思路：

代码

```
from collections import deque

n = int(input())
tree = [-1]
h = [0] * (n + 1)
ans = {0: [1]}
for i in range(1, n + 1):
    ans[i] = []
for _ in range(n):
    tree.append(tuple(map(int, input().split())))
q = deque()
q.append(1)
while q:
    a = q.popleft()
    b, c = tree[a][0], tree[a][1]
    if b != -1:
        q.append(b)
        h[b] = h[a] + 1
        ans[h[b]].append(b)
    if c != -1:
        q.append(c)
        h[c] = h[a] + 1
```

```
            ans[h[c]].append(c)
    print(' '.join(map(str, [x[-1] for x in ans.values() if x != []])))
```

代码运行截图 （至少包含有"Accepted"）

状态: Accepted

源代码

```
from collections import deque

n = int(input())
tree = [-1]
h = [0] * (n + 1)
ans = {0: [1]}
for i in range(1, n + 1):
    ans[i] = []
for _ in range(n):
    tree.append(tuple(map(int, input().split())))
q = deque()
q.append(1)
while q:
    a = q.popleft()
    b, c = tree[a][0], tree[a][1]
    if b != -1:
        q.append(b)
        h[b] = h[a] + 1
        ans[h[b]].append(b)
    if c != -1:
        q.append(c)
        h[c] = h[a] + 1
        ans[h[c]].append(c)
print(' '.join(map(str, [x[-1] for x in ans.values() if x != []])))
```

28203:【模板】单调栈

http://cs101.openjudge.cn/practice/28203/

思路：

代码

```
n = int(input())
a = list(map(int, input().split()))
st = []
first = True
ans = [0] * n
for i in range(n):
    while st and a[st[-1]] < a[i]:
        ans[st[-1]] = i + 1
        st.pop()
    st.append(i)
print(*ans)
```

代码运行截图 （至少包含有"Accepted"）

状态: Accepted

源代码

```
n = int(input())
a = list(map(int, input().split()))
st = []
first = True
ans = [0] * n
for i in range(n):
    while st and a[st[-1]] < a[i]:
        ans[st[-1]] = i + 1
        st.pop()
    st.append(i)
print(*ans)
```

09202: 舰队、海域出击!

http://cs101.openjudge.cn/practice/09202/

思路:

代码

```
t = int(input())
for __ in range(t):
    n, m = map(int, input().split())
    g = [[] for _ in range(n)]
    for _ in range(m):
        u, v = map(lambda x: int(x) - 1, input().split())
        g[u].append(v)
    loop = False


    def dfs(num):
        v[num] = 1
        result = False
        for neighbour in g[num]:
            if v[neighbour] == 0 and dfs(neighbour):
                result = True
            if v[neighbour] == 1:
                result = True
        v[num] = 2
        return result


    v = [0 for i in range(n)]
    found = False
```

```
        for i in range(n):
            if v[i] == 0:
                if dfs(i):
                    print('Yes')
                    found = True
                    break
        if not found:
            print('No')
```

代码运行截图 （AC代码截图，至少包含有"Accepted"）

状态: Accepted

源代码

```
t = int(input())
for __ in range(t):
    n, m = map(int, input().split())
    g = [[] for _ in range(n)]
    for _ in range(m):
        u, v = map(lambda x: int(x) - 1, input().split())
        g[u].append(v)
    loop = False


    def dfs(num):
        v[num] = 1
        result = False
        for neighbour in g[num]:
            if v[neighbour] == 0 and dfs(neighbour):
                result = True
                break
            if v[neighbour] == 1:
                result = True
                break
        v[num] = 2
        return result


    v = [0 for i in range(n)]
    found = False
    for i in range(n):
        if v[i] == 0:
            if dfs(i):
                print('Yes')
                found = True
                break
    if not found:
        print('No')
```

04135: 月度开销

http://cs101.openjudge.cn/practice/04135/

思路：二分查找

代码

```
n, m = map(int, input().split())
a = [0] * n
for i in range(n):
    a[i] = int(input())

left, right = 0, sum(a)
while right - left > 0:
    mid = (right + left) // 2
    group_count = 0
    i = 0
    while i < n and group_count < m:
        current_sum = 0
        j = 0
        while i + j < n and current_sum + a[i + j] <= mid:
            current_sum += a[i + j]
            j += 1
        i = i + j
        group_count += 1
    if i == n:
        right = mid
    else:
        if left == right - 1:
            break
        left = mid
print(right)
```

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 = map(int, input().split())
a = [0] * n
for i in range(n):
    a[i] = int(input())

left, right = 0, sum(a)
while right - left > 0:
    mid = (right + left) // 2
    group_count = 0
    i = 0
    while i < n and group_count < m:
        current_sum = 0
        j = 0
        while i + j < n and current_sum + a[i + j] <= mid:
            current_sum += a[i + j]
            j += 1
        i = i + j
        group_count += 1
    if i == n:
        right = mid
    else:
        if left == right - 1:
            break
        left = mid
print(right)
```

07735: 道路

http://cs101.openjudge.cn/practice/07735/

思路:

代码

```python
from heapq import heappush, heappop

k = int(input())
n = int(input())
r = int(input())
graph = [[] for _ in range(n)]
for i in range(r):
    s, d, l, t = map(int, input().split())
    graph[s - 1].append((d - 1, l, t))

pq = [(0, 0, 0)]
found = False
while pq:
    l, d, f = heappop(pq)
    if d == n - 1:
        print(l)
        found = True
        break
    for nd, dl, df in graph[d]:
        nl = dl + l
        nf = df + f
        if nf <= k:
            heappush(pq, (nl, nd, nf))
if not found:
    print(-1)
```

代码运行截图 （AC代码截图，至少包含有"Accepted"）

状态: Accepted

源代码

```
from heapq import heappush, heappop

k = int(input())
n = int(input())
r = int(input())
graph = [[] for _ in range(n)]
for i in range(r):
    s, d, l, t = map(int, input().split())
    graph[s - 1].append((d - 1, l, t))

pq = [(0, 0, 0)]
found = False
while pq:
    l, d, f = heappop(pq)
    if d == n - 1:
        print(l)
        found = True
        break
    for nd, dl, df in graph[d]:
        nl = dl + l
        nf = df + f
        if nf <= k:
            heappush(pq, (nl, nd, nf))
if not found:
    print(-1)
```

01182: 食物链

http://cs101.openjudge.cn/practice/01182/

思路:

代码

```python
def find(x):
    if x != pre[x]:
        t = find(pre[x])
        h[x] += h[pre[x]]
        pre[x] = t
    return pre[x]


n, k = map(int, input().split())
ans = 0
pre = [i for i in range(n)]
h = [0 for _ in range(n)]
for _ in range(k):
    sign, a, b = map(lambda x: int(x) - 1, input().split())
    if a >= n or b >= n:
        ans += 1
        continue
    fa, fb = find(a), find(b)
    if fa == fb and (h[a] - h[b]) % 3 != sign:
        ans += 1
    else:
        pre[fa] = fb
        h[fa] = h[b] - h[a] + sign
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

```
def find(x):
    if x != pre[x]:
        t = find(pre[x])
        h[x] += h[pre[x]]
        pre[x] = t
    return pre[x]


n, k = map(int, input().split())
ans = 0
pre = [i for i in range(n)]
h = [0 for _ in range(n)]
for _ in range(k):
    sign, a, b = map(lambda x: int(x) - 1, input().split())
    if a >= n or b >= n:
        ans += 1
        continue
    fa, fb = find(a), find(b)
    if fa == fb and (h[a] - h[b]) % 3 != sign:
        ans += 1
    else:
        pre[fa] = fb
        h[fa] = h[b] - h[a] + sign
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。

前四题比较套路，但是细节还需要想清楚。后两题自己实在很难想到，得多做多看。