Assignment #F: All-Killed 满分

Updated 1844 GMT+8 May 20, 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

22485: 升空的焰火，从侧面看

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

思路：

代码

```
#
class tree:
    def __init__(self,value):
        self.left=None
        self.right=None
        self.value=value
def findright(M):
    flag=0
    cnt=0
    N=[]
    for i in range(len(M)-1,-1,-1):
```

```
        if(t[M[i]].right!=-2 and flag==0):
            flag=1
            cnt=t[M[i]].right
        elif(t[M[i]].left!=-2 and flag==0):
            flag=1
            cnt=t[M[i]].left
    for i in M:
        if(t[i].left!=-2):
            N.append(t[i].left)
        if(t[i].right!=-2):
            N.append(t[i].right)
    if(cnt==0):
        return []
    else:
        return [str(cnt+1)]+findright(N)
n=int(input())
t=n*[0]
for i in range(n):
    a,b=[int(i) for i in input().split()]
    t[i]=tree(i)
    t[i].left=a-1
    t[i].right=b-1
print(' '.join([str(1)]+findright([0])))
```

代码运行截图 ==（至少包含有"Accepted"）==

状态: Accepted

源代码

```
class tree:
    def __init__(self,value):
        self.left=None
        self.right=None
        self.value=value
def findright(M):
    flag=0
    cnt=0
    N=[]
    for i in range(len(M)-1,-1,-1):
        if(t[M[i]].right!=-2 and flag==0):
            flag=1
            cnt=t[M[i]].right
        elif(t[M[i]].left!=-2 and flag==0):
            flag=1
            cnt=t[M[i]].left
    for i in M:
        if(t[i].left!=-2):
            N.append(t[i].left)
        if(t[i].right!=-2):
            N.append(t[i].right)
    if(cnt==0):
        return []
    else:
        return [str(cnt+1)]+findright(N)
n=int(input())
t=n*[0]
for i in range(n):
    a,b=[int(i) for i in input().split()]
    t[i]=tree(i)
    t[i].left=a-1
    t[i].right=b-1
print(' '.join([str(1)]+findright([0])))
```

28203:【模板】单调栈

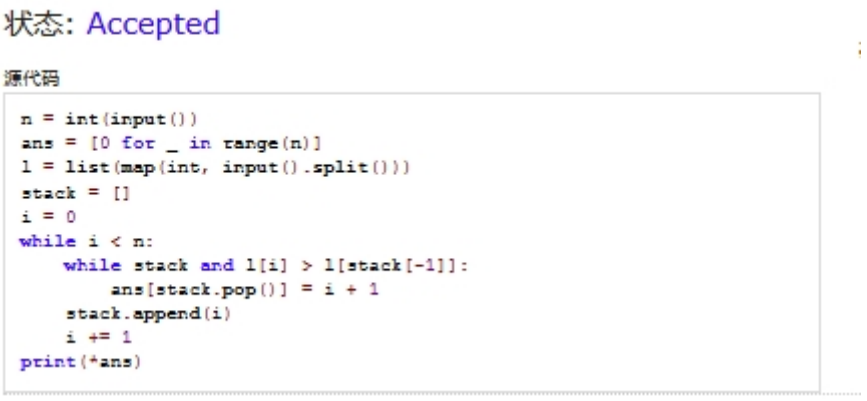
http://cs101.openjudge.cn/practice/28203/

思路:

代码

```
#
n = int(input())
ans = [0 for _ in range(n)]
l = list(map(int, input().split()))
stack = []
i = 0
while i < n:
    while stack and l[i] > l[stack[-1]]:
        ans[stack.pop()] = i + 1
    stack.append(i)
    i += 1
print(*ans)
```

代码运行截图 ==（至少包含有"Accepted"）==



09202: 舰队、海域出击!

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

思路:

代码

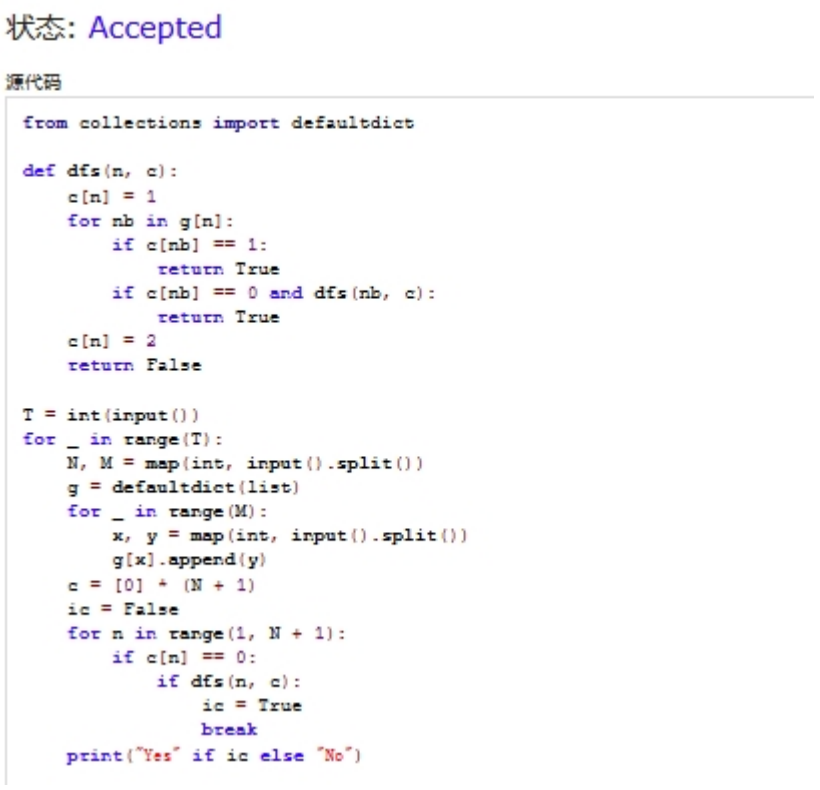
```
# from collections import defaultdict

def dfs(n, c):
```

```
    c[n] = 1
    for nb in g[n]:
        if c[nb] == 1:
            return True
        if c[nb] == 0 and dfs(nb, c):
            return True
    c[n] = 2
    return False

T = int(input())
for _ in range(T):
    N, M = map(int, input().split())
    g = defaultdict(list)
    for _ in range(M):
        x, y = map(int, input().split())
        g[x].append(y)
    c = [0] * (N + 1)
    ic = False
    for n in range(1, N + 1):
        if c[n] == 0:
            if dfs(n, c):
                ic = True
                break
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==



04135: 月度开销

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

思路:

代码

```
#
n, m = map(int, input().split())
ex = []
for _ in range(n):
    ex.append(int(input()))

def check(x):
    num, s = 1, 0
    for i in range(n):
        if s + ex[i] > x:
            s = ex[i]
            num += 1
        else:
            s += ex[i]

    return [False, True][num > m]

lo = max(ex)
hi = sum(ex) + 1
ans = 1
while lo < hi:
    mid = (lo + hi) // 2
    if check(mid):
        lo = mid + 1
    else:
        ans = mid
        hi = mid

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

```
n, m = map(int, input().split())
ex = []
for _ in range(n):
    ex.append(int(input()))

def check(x):
    num, s = 1, 0
    for i in range(n):
        if s + ex[i] > x:
            s = ex[i]
            num += 1
        else:
            s += ex[i]

    return [False, True][num > m]

lo = max(ex)
hi = sum(ex) + 1
ans = 1
while lo < hi:
    mid = (lo + hi) // 2
    if check(mid):
        lo = mid + 1
    else:
        ans = mid
        hi = mid

print(ans)
```

07735: 道路

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

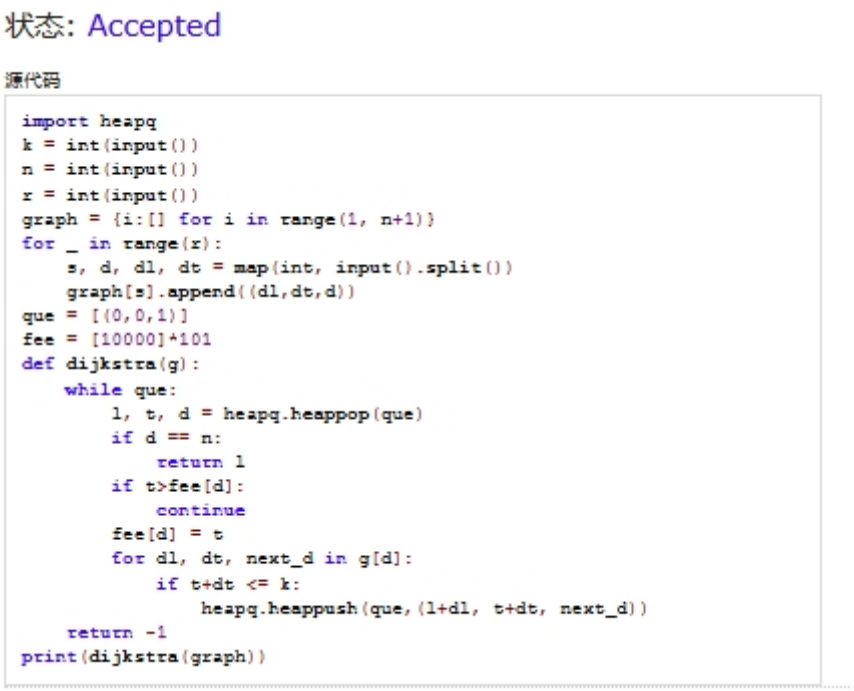
思路:

代码

```
#
import heapq
k = int(input())
n = int(input())
r = int(input())
graph = {i:[] for i in range(1, n+1)}
for _ in range(r):
    s, d, dl, dt = map(int, input().split())
    graph[s].append((dl,dt,d))
que = [(0,0,1)]
fee = [10000]*101
def dijkstra(g):
    while que:
        l, t, d = heapq.heappop(que)
        if d == n:
            return l
        if t>fee[d]:
            continue
        fee[d] = t
        for dl, dt, next_d in g[d]:
            if t+dt <= k:
                heapq.heappush(que,(l+dl, t+dt, next_d))
```

```
    return -1
print(dijkstra(graph))
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==


状态: Accepted

源代码

```
import heapq
k = int(input())
n = int(input())
r = int(input())
graph = {i:[] for i in range(1, n+1)}
for _ in range(r):
    s, d, dl, dt = map(int, input().split())
    graph[s].append((dl,dt,d))
que = [(0,0,1)]
fee = [10000]*101
def dijkstra(g):
    while que:
        l, t, d = heapq.heappop(que)
        if d == n:
            return l
        if t>fee[d]:
            continue
        fee[d] = t
        for dl, dt, next_d in g[d]:
            if t+dt <= k:
                heapq.heappush(que,(l+dl, t+dt, next_d))
    return -1
print(dijkstra(graph))
```

01182: 食物链

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

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。==

这次还是模版题居多，感觉做的..很开心?