Assignment #F: All-Killed 满分

Updated 1844 GMT+8 May 20, 2024

2024 spring, Complied by ==同学的姓名、院系==

罗瑞哲 生命科学学院

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：Windows 10

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

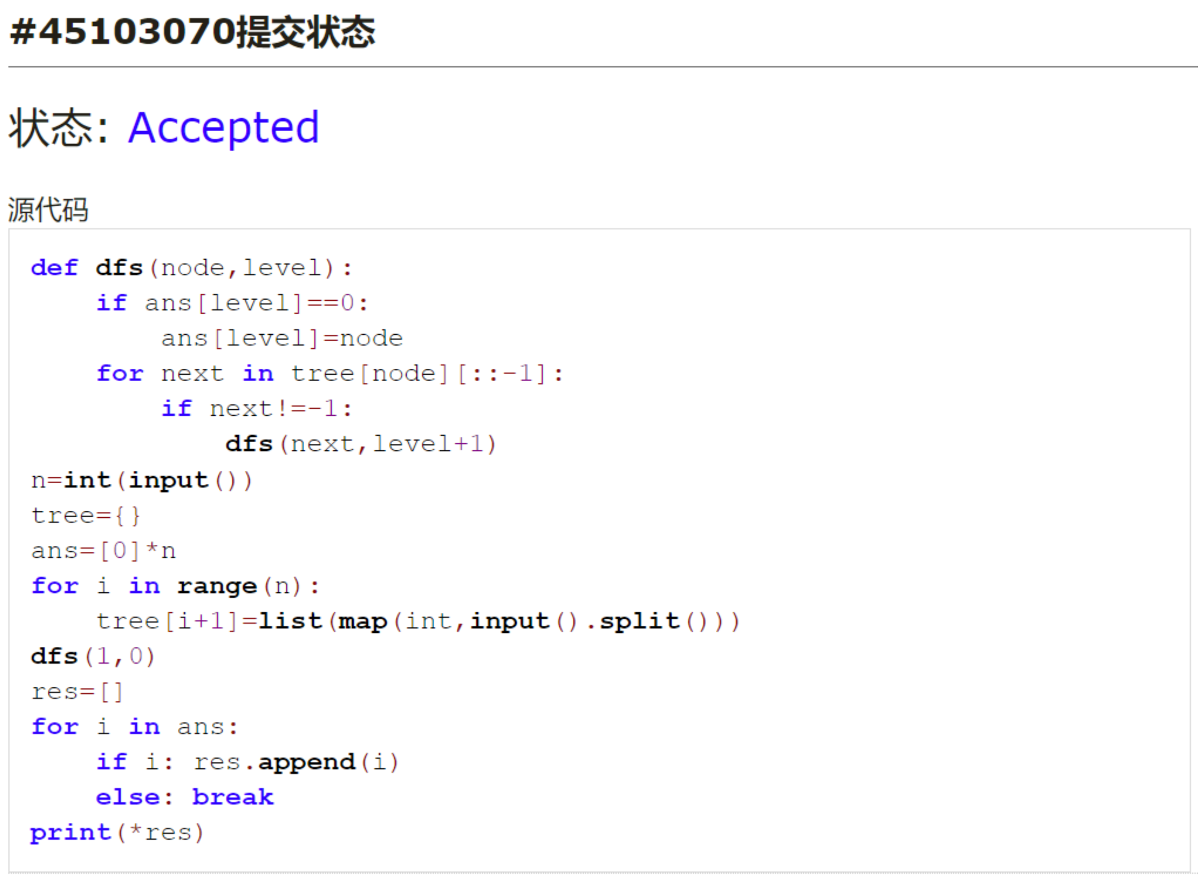
http://cs101.openjudge.cn/practice/22485/

思路：

代码

```
# 
def dfs(node,level):
    if ans[level]==0:
        ans[level]=node
    for next in tree[node][::-1]:
        if next!=-1:
            dfs(next,level+1)
n=int(input())
tree={}
ans=[0]*n
for i in range(n):
    tree[i+1]=list(map(int,input().split()))
dfs(1,0)
res=[]
for i in ans:
    if i: res.append(i)
    else: break
print(*res)
```

代码运行截图 ==（至少包含有"Accepted"）==



#45103070提交状态

状态: Accepted

源代码

```
def dfs(node,level):
    if ans[level]==0:
        ans[level]=node
    for next in tree[node][::-1]:
        if next!=-1:
            dfs(next,level+1)
n=int(input())
tree={}
ans=[0]*n
for i in range(n):
    tree[i+1]=list(map(int,input().split()))
dfs(1,0)
res=[]
for i in ans:
    if i: res.append(i)
    else: break
print(*res)
```

28203:【模板】单调栈

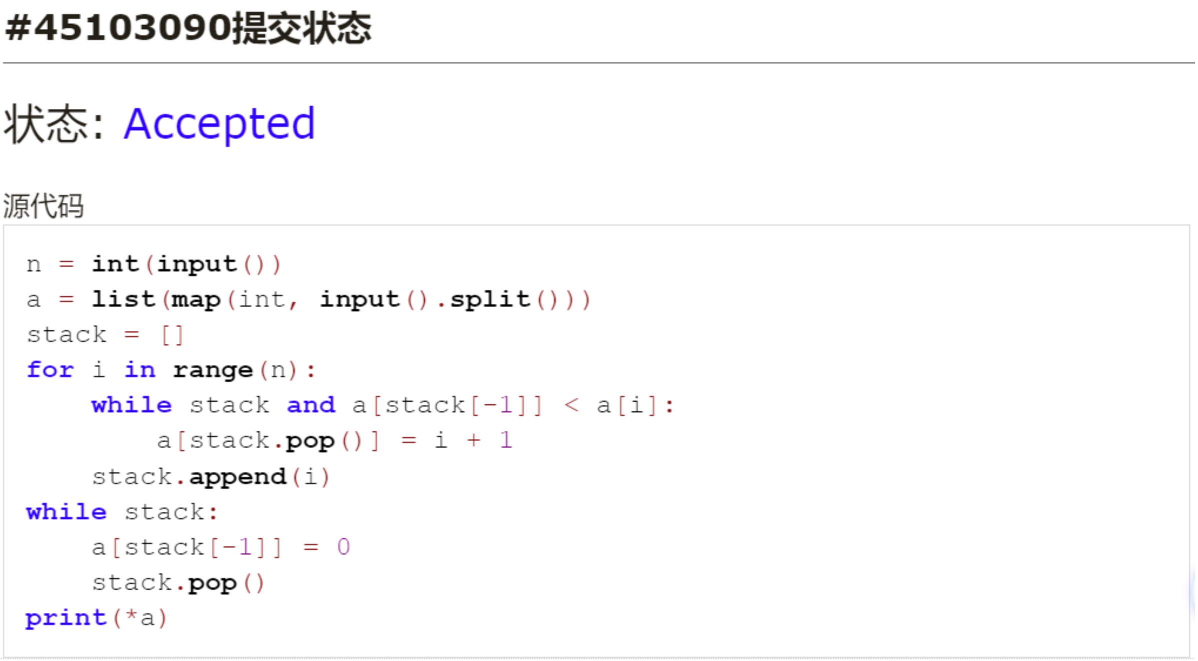
http://cs101.openjudge.cn/practice/28203/

思路：

代码

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

代码运行截图 ==（至少包含有"Accepted"）==



#45103090提交状态

状态: Accepted

源代码

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

09202: 舰队、海域出击！

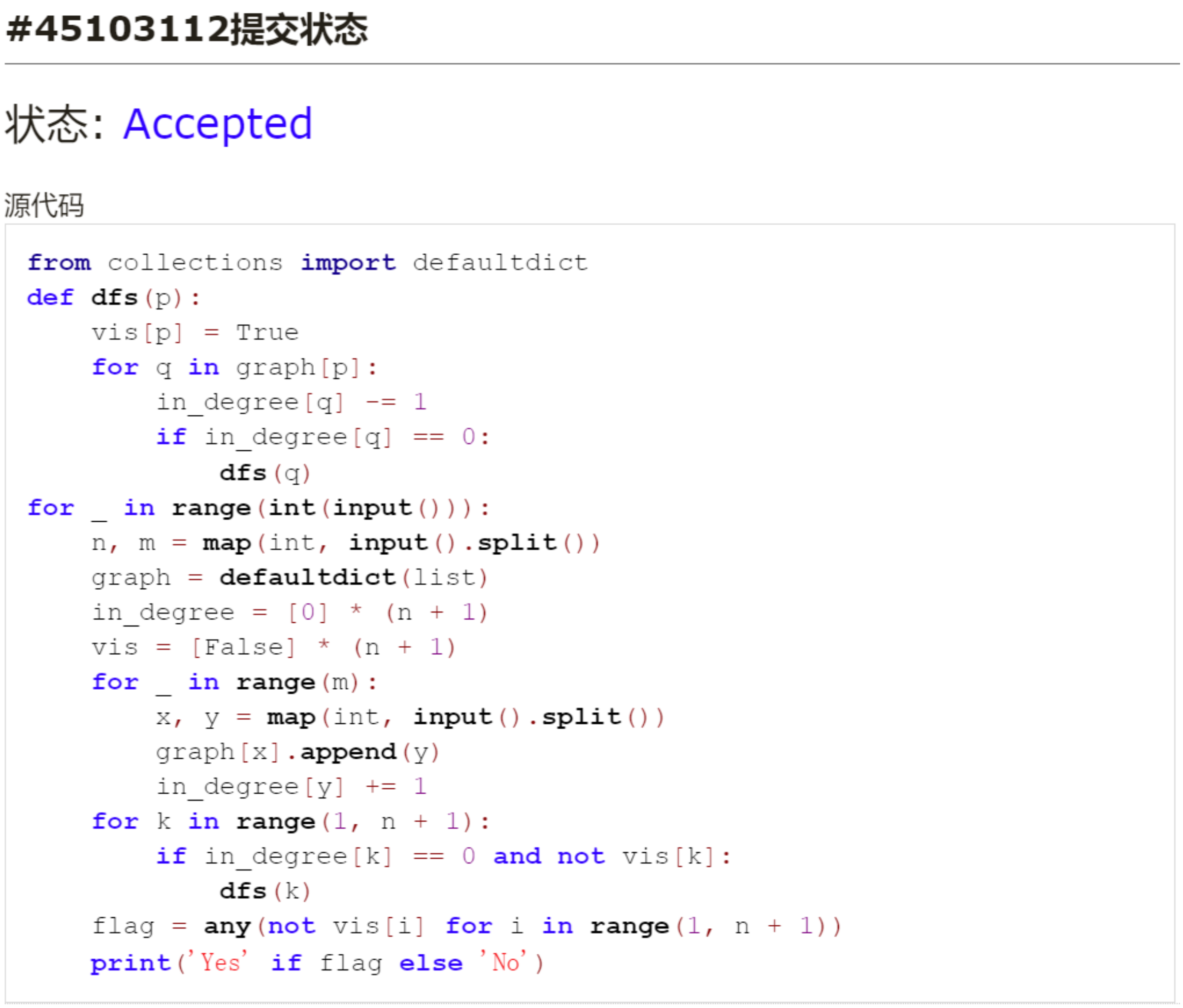
http://cs101.openjudge.cn/practice/09202/

思路：

代码

```
#
from collections import defaultdict
def dfs(p):
    vis[p] = True
    for q in graph[p]:
        in_degree[q] -= 1
        if in_degree[q] == 0:
            dfs(q)
for _ in range(int(input())):
    n, m = map(int, input().split())
    graph = defaultdict(list)
    in_degree = [0] * (n + 1)
    vis = [False] * (n + 1)
    for _ in range(m):
        x, y = map(int, input().split())
        graph[x].append(y)
        in_degree[y] += 1
    for k in range(1, n + 1):
        if in_degree[k] == 0 and not vis[k]:
            dfs(k)
    flag = any(not vis[i] for i in range(1, n + 1))
    print('Yes' if flag else 'No')
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==



#45103112提交状态

状态: Accepted

源代码

```
from collections import defaultdict
def dfs(p):
    vis[p] = True
    for q in graph[p]:
        in_degree[q] -= 1
        if in_degree[q] == 0:
            dfs(q)
for _ in range(int(input())):
    n, m = map(int, input().split())
    graph = defaultdict(list)
    in_degree = [0] * (n + 1)
    vis = [False] * (n + 1)
    for _ in range(m):
        x, y = map(int, input().split())
        graph[x].append(y)
        in_degree[y] += 1
    for k in range(1, n + 1):
        if in_degree[k] == 0 and not vis[k]:
            dfs(k)
    flag = any(not vis[i] for i in range(1, n + 1))
    print('Yes' if flag else 'No')
```

04135: 月度开销

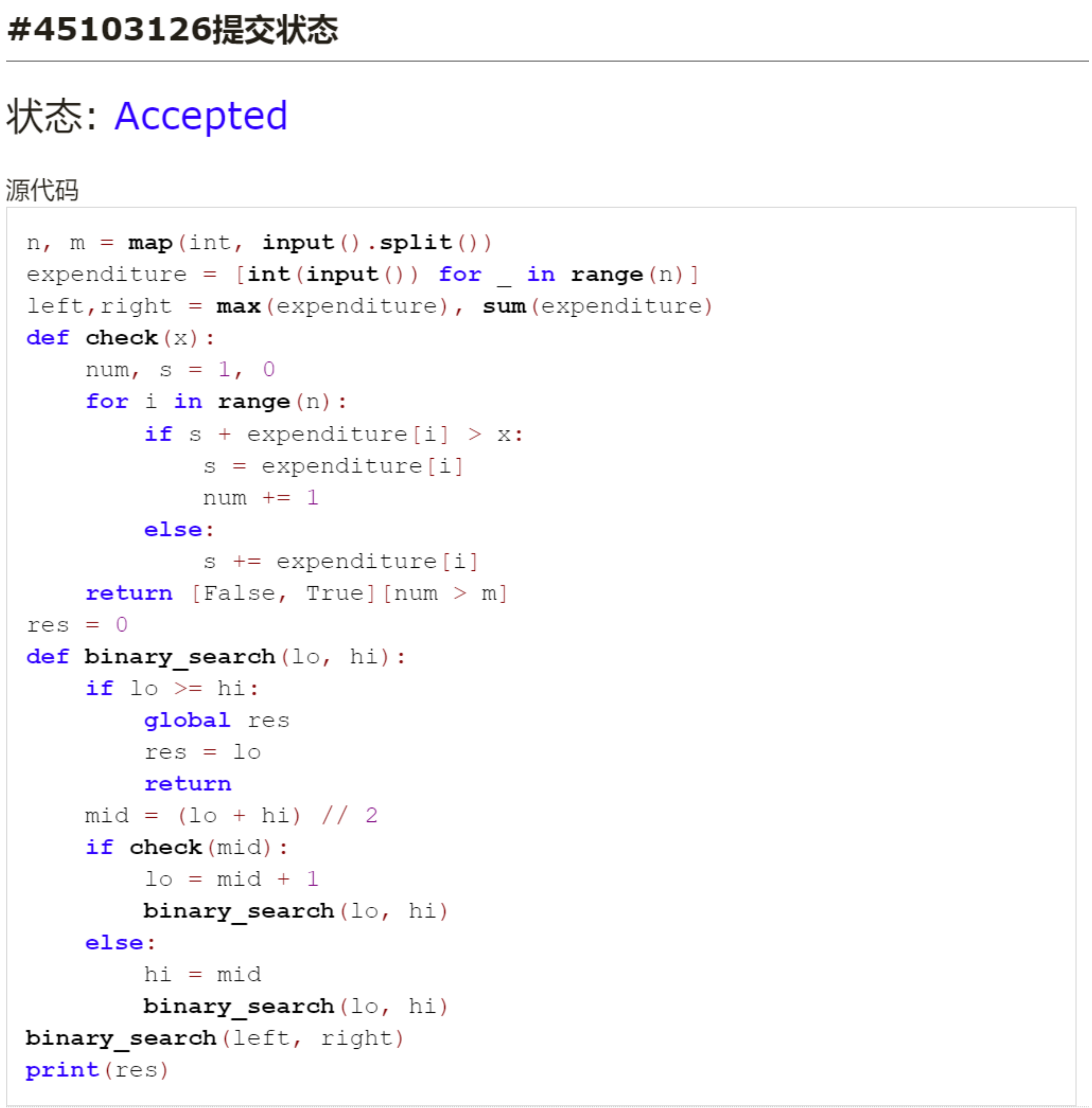
http://cs101.openjudge.cn/practice/04135/

思路：

代码

```python
#
n, m = map(int, input().split())
expenditure = [int(input()) for _ in range(n)]
left,right = max(expenditure), sum(expenditure)
def check(x):
    num, s = 1, 0
    for i in range(n):
        if s + expenditure[i] > x:
            s = expenditure[i]
            num += 1
        else:
            s += expenditure[i]
    return [False, True][num > m]
res = 0
def binary_search(lo, hi):
    if lo >= hi:
        global res
        res = lo
        return
    mid = (lo + hi) // 2
    if check(mid):
        lo = mid + 1
        binary_search(lo, hi)
    else:
        hi = mid
        binary_search(lo, hi)
binary_search(left, right)
print(res)
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==



#45103126提交状态

状态: Accepted

源代码

```
n, m = map(int, input().split())
expenditure = [int(input()) for _ in range(n)]
left,right = max(expenditure), sum(expenditure)
def check(x):
    num, s = 1, 0
    for i in range(n):
        if s + expenditure[i] > x:
            s = expenditure[i]
            num += 1
        else:
            s += expenditure[i]
    return [False, True][num > m]
res = 0
def binary_search(lo, hi):
    if lo >= hi:
        global res
        res = lo
        return
    mid = (lo + hi) // 2
    if check(mid):
        lo = mid + 1
        binary_search(lo, hi)
    else:
        hi = mid
        binary_search(lo, hi)
binary_search(left, right)
print(res)
```

07735: 道路

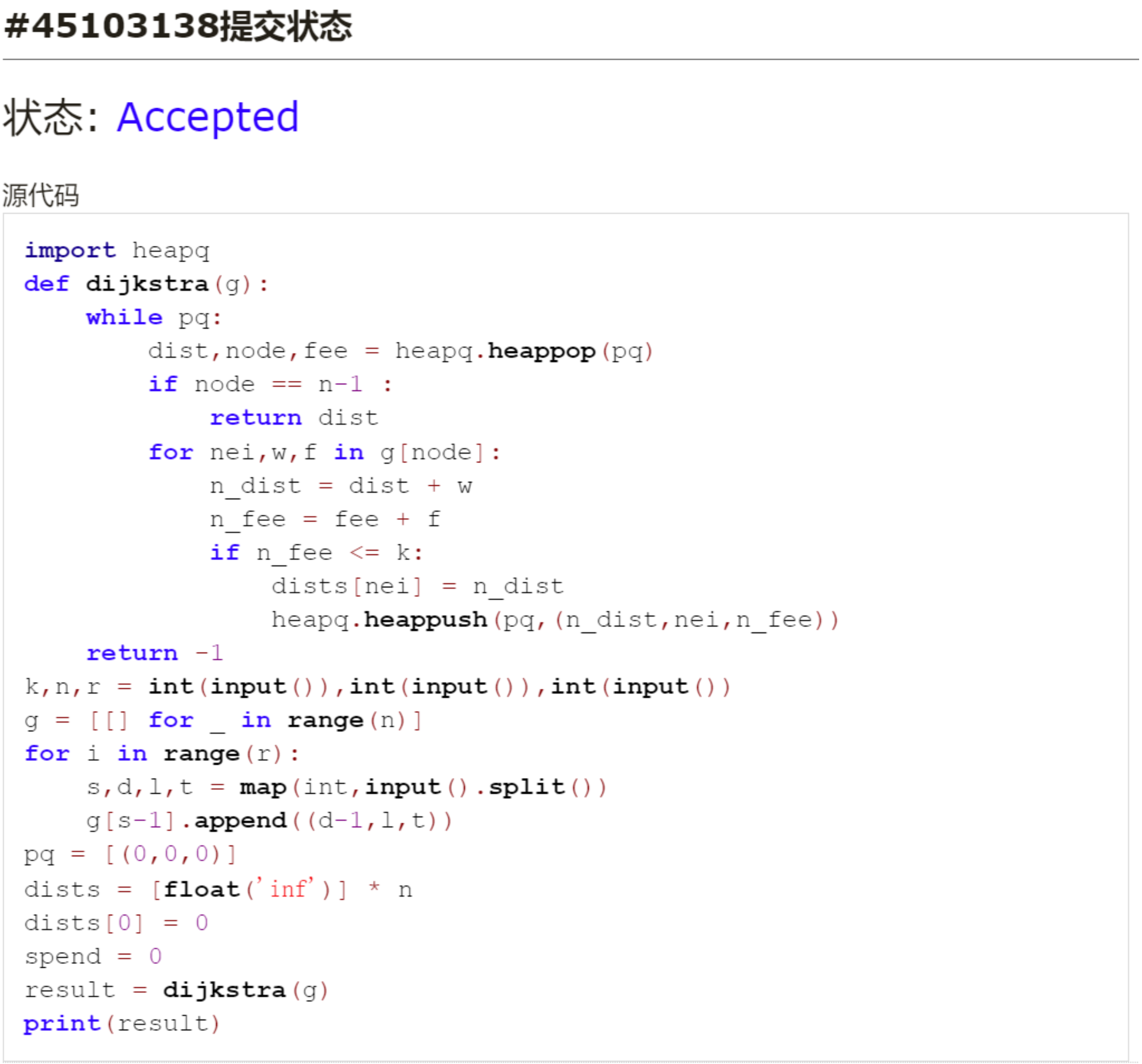
http://cs101.openjudge.cn/practice/07735/

思路：

代码

```
#
import heapq
def dijkstra(g):
    while pq:
        dist,node,fee = heapq.heappop(pq)
        if node == n-1 :
            return dist
        for nei,w,f in g[node]:
            n_dist = dist + w
            n_fee = fee + f
            if n_fee <= k:
                dists[nei] = n_dist
                heapq.heappush(pq,(n_dist,nei,n_fee))
    return -1
k,n,r = int(input()),int(input()),int(input())
g = [[] for _ in range(n)]
for i in range(r):
    s,d,l,t = map(int,input().split())
    g[s-1].append((d-1,l,t))
pq = [(0,0,0)]
dists = [float('inf')] * n
dists[0] = 0
spend = 0
result = dijkstra(g)
print(result)
```

代码运行截图 == （AC代码截图，至少包含有"Accepted"） ==



#45103138提交状态

状态: Accepted

源代码

```
import heapq
def dijkstra(g):
    while pq:
        dist,node,fee = heapq.heappop(pq)
        if node == n-1 :
            return dist
        for nei,w,f in g[node]:
            n_dist = dist + w
            n_fee = fee + f
            if n_fee <= k:
                dists[nei] = n_dist
                heapq.heappush(pq,(n_dist,nei,n_fee))
    return -1
k,n,r = int(input()),int(input()),int(input())
g = [[] for _ in range(n)]
for i in range(r):
    s,d,l,t = map(int,input().split())
    g[s-1].append((d-1,l,t))
pq = [(0,0,0)]
dists = [float(’inf’)] * n
dists[0] = 0
spend = 0
result = dijkstra(g)
print(result)
```

01182: 食物链

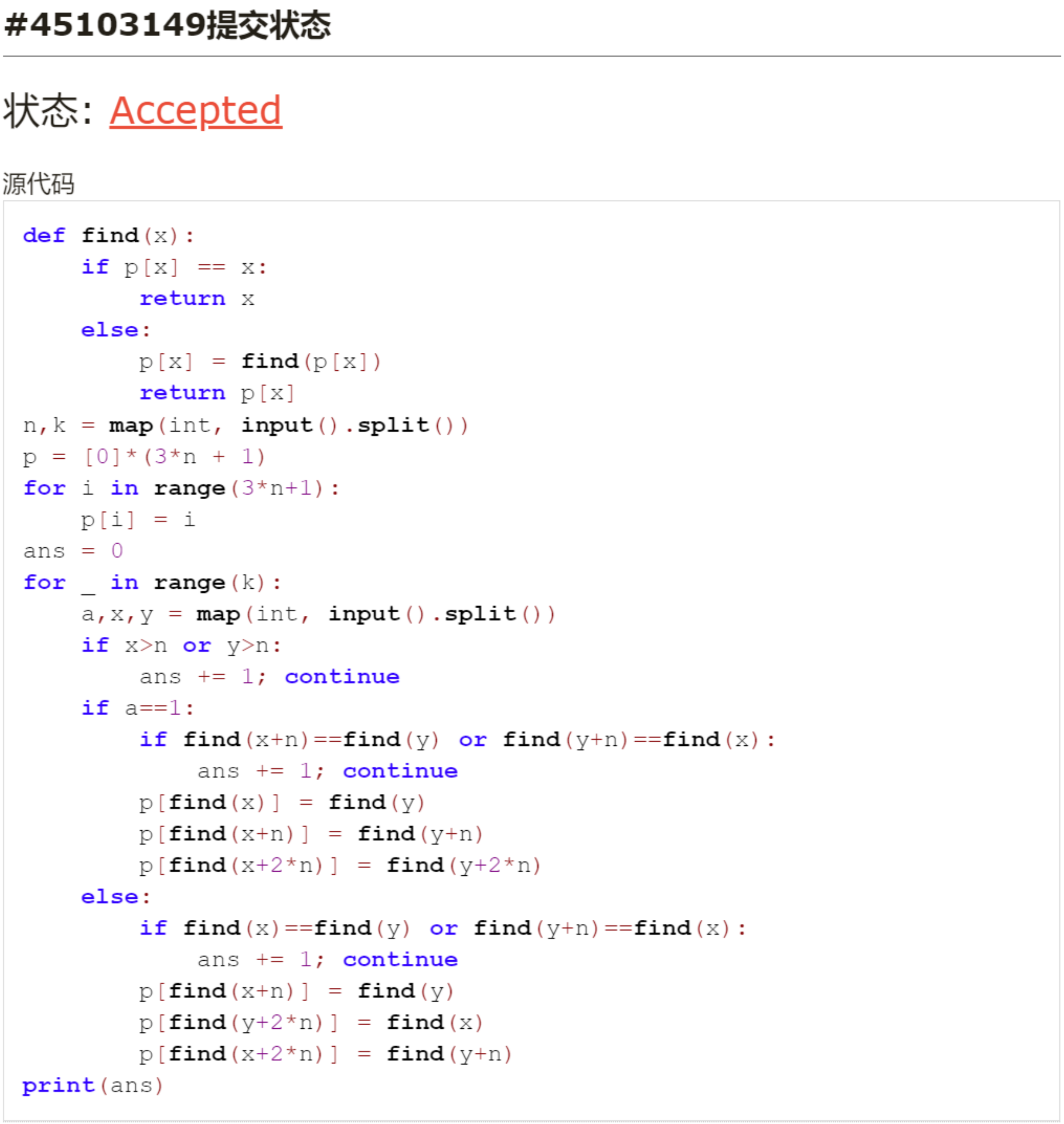
http://cs101.openjudge.cn/practice/01182/

思路：

代码

```python
#
def find(x):
    if p[x] == x:
        return x
    else:
        p[x] = find(p[x])
        return p[x]
n,k = map(int, input().split())
p = [0]*(3*n + 1)
for i in range(3*n+1):
    p[i] = i
ans = 0
for _ in range(k):
    a,x,y = map(int, input().split())
    if x>n or y>n:
        ans += 1; continue
    if a==1:
        if find(x+n)==find(y) or find(y+n)==find(x):
            ans += 1; continue
        p[find(x)] = find(y)
        p[find(x+n)] = find(y+n)
        p[find(x+2*n)] = find(y+2*n)
    else:
        if find(x)==find(y) or find(y+n)==find(x):
            ans += 1; continue
        p[find(x+n)] = find(y)
        p[find(y+2*n)] = find(x)
        p[find(x+2*n)] = find(y+n)
print(ans)
```

代码运行截图 == （AC代码截图，至少包含有"Accepted"） ==



#45103149提交状态

状态：Accepted

源代码

```python
def find(x):
    if p[x] == x:
        return x
    else:
        p[x] = find(p[x])
        return p[x]
n,k = map(int, input().split())
p = [0]*(3*n + 1)
for i in range(3*n+1):
    p[i] = i
ans = 0
for _ in range(k):
    a,x,y = map(int, input().split())
    if x>n or y>n:
        ans += 1; continue
    if a==1:
        if find(x+n)==find(y) or find(y+n)==find(x):
            ans += 1; continue
        p[find(x)] = find(y)
        p[find(x+n)] = find(y+n)
        p[find(x+2*n)] = find(y+2*n)
    else:
        if find(x)==find(y) or find(y+n)==find(x):
            ans += 1; continue
        p[find(x+n)] = find(y)
        p[find(y+2*n)] = find(x)
        p[find(x+2*n)] = find(y+n)
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。==

周三实验课上完了，我一定要在期末上机之前先去感受一下机房的电脑。