Assignment #B: 图论和树算

Updated 1709 GMT+8 Apr 28, 2024

2024 spring, Complied by =田济维 物理学院=

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

（python pycharm）

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

28170: 算鹰

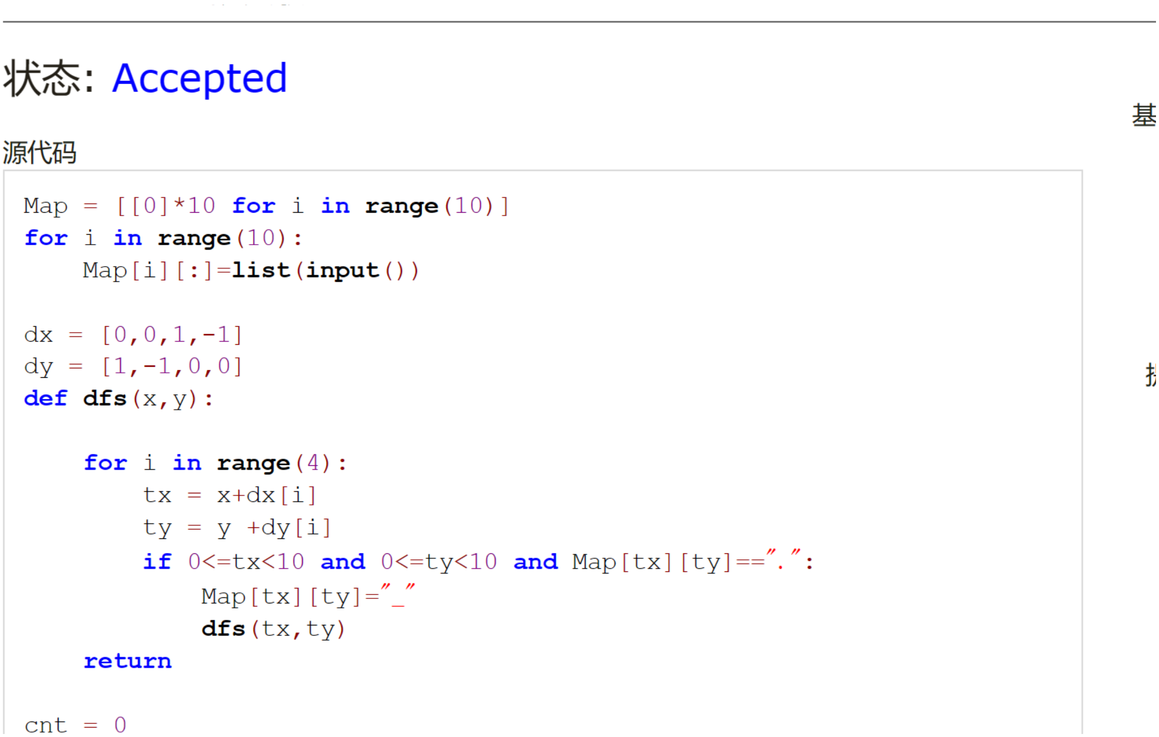
dfs, http://cs101.openjudge.cn/practice/28170/

思路：

代码

```python
#
Map = [[0]*10 for i in range(10)]
for i in range(10):
    Map[i][:]=list(input())

dx = [0,0,1,-1]
dy = [1,-1,0,0]
def dfs(x,y):

    for i in range(4):
        tx = x+dx[i]
        ty = y +dy[i]
        if 0<=tx<10 and 0<=ty<10 and Map[tx][ty]==".":
            Map[tx][ty]="_"
            dfs(tx,ty)
    return

cnt = 0
for i in range(10):
    for j in range(10):
        if Map[i][j]==".":
            dfs(i,j)
            cnt+=1
print(cnt)


```

代码运行截图 （至少包含有"Accepted"）



状态: Accepted

源代码

```
Map = [[0]*10 for i in range(10)]
for i in range(10):
    Map[i][:]=list(input())

dx = [0,0,1,-1]
dy = [1,-1,0,0]
def dfs(x,y):

    for i in range(4):
        tx = x+dx[i]
        ty = y +dy[i]
        if 0<=tx<10 and 0<=ty<10 and Map[tx][ty]==".":
            Map[tx][ty]="_"
            dfs(tx,ty)
    return

cnt = 0
```

02754: 八皇后

dfs, http://cs101.openjudge.cn/practice/02754/

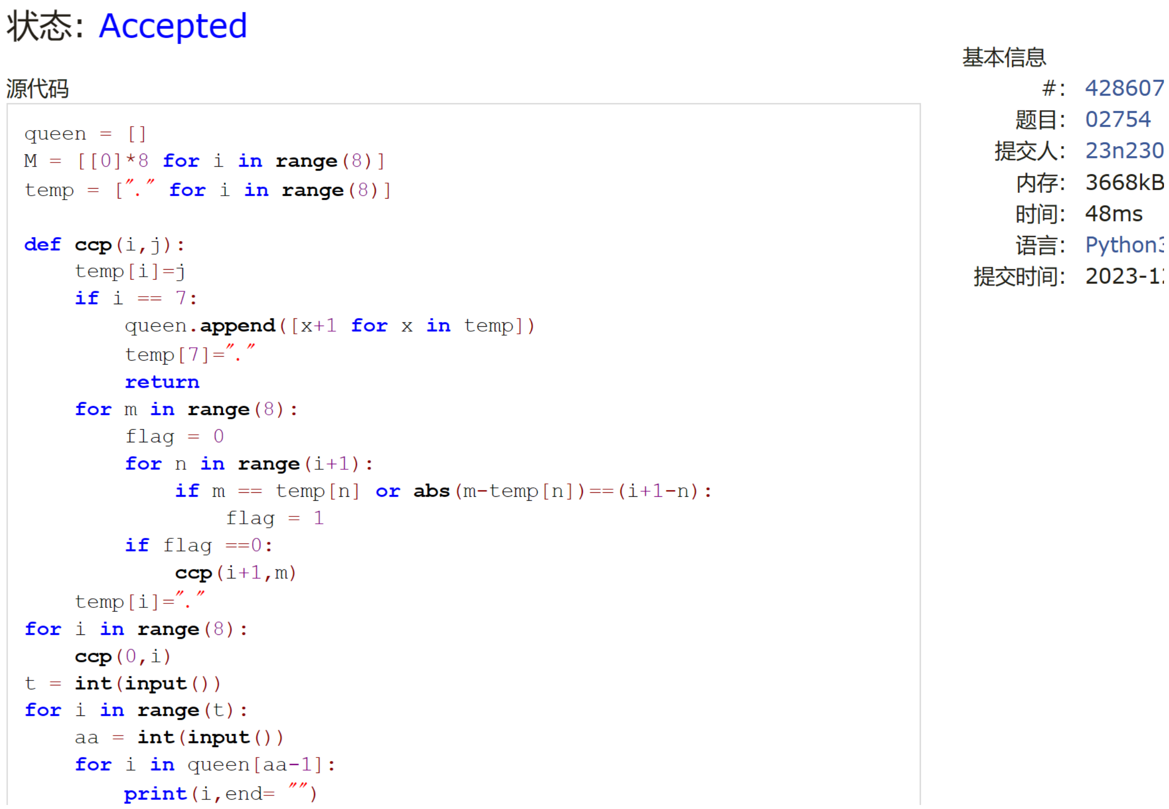
思路：

代码

```
#
```

```python
queen = []
M = [[0]*8 for i in range(8)]
temp = ["." for i in range(8)]

def ccp(i,j):
    temp[i]=j
    if i == 7:
        queen.append([x+1 for x in temp])
        temp[7]="."
        return
    for m in range(8):
        flag = 0
        for n in range(i+1):
            if m == temp[n] or abs(m-temp[n])==(i+1-n):
                flag = 1
        if flag ==0:
            ccp(i+1,m)
    temp[i]="."
for i in range(8):
    ccp(0,i)
t = int(input())
for i in range(t):
    aa = int(input())
    for i in queen[aa-1]:
        print(i,end= "")
    print()
```

代码运行截图 （至少包含有"Accepted"）



状态: Accepted

源代码

```python
queen = []
M = [[0]*8 for i in range(8)]
temp = ["." for i in range(8)]

def ccp(i,j):
    temp[i]=j
    if i == 7:
        queen.append([x+1 for x in temp])
        temp[7]="."
        return
    for m in range(8):
        flag = 0
        for n in range(i+1):
            if m == temp[n] or abs(m-temp[n])==(i+1-n):
                flag = 1
        if flag ==0:
            ccp(i+1,m)
    temp[i]="."
for i in range(8):
    ccp(0,i)
t = int(input())
for i in range(t):
    aa = int(input())
    for i in queen[aa-1]:
        print(i,end= "")
```

基本信息

#: 428607
题目: 02754
提交人: 23n230
内存: 3668kB
时间: 48ms
语言: Python3
提交时间: 2023-1

03151: Pots

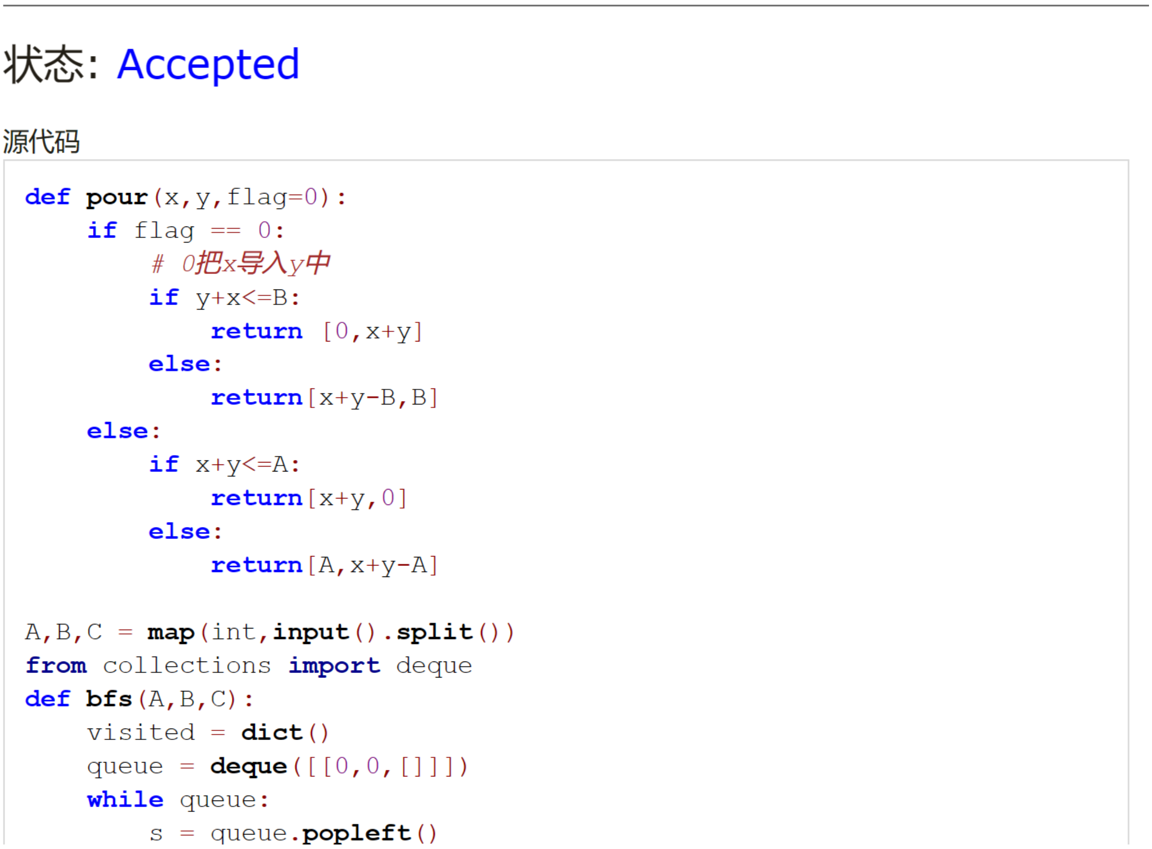
bfs, http://cs101.openjudge.cn/practice/03151/

思路：

代码

```python
#
def pour(x,y,flag=0):
    if flag == 0:
        # 0把x导入y中
        if y+x<=B:
            return [0,x+y]
        else:
            return[x+y-B,B]
    else:
        if x+y<=A:
            return[x+y,0]
        else:
            return[A,x+y-A]

A,B,C = map(int,input().split())
from collections import deque
def bfs(A,B,C):
    visited = dict()
    queue = deque([[0,0,[]]])
    while queue:
        s = queue.popleft()
        tA = s[0]
        tB = s[1]
        op = s[2]
        if C in [tA,tB]:
            print(len(op))
            for x in op:
                print(x)
            return
        if (0,tB) not in visited:
            op1 = op+["DROP(1)"]
            visited[(0,tB)]=1
            queue.append([0,tB,op1])

        if (tA,0) not in visited:
            op2 = op + ["DROP(2)"]
            visited[(tA,0)]=1
            queue.append([tA,0,op2])

        if (tA,B) not in visited:
            op3=op+["FILL(2)"]
            visited[(tA,B)]=1
            queue.append([tA,B,op3])


        if (A, tB) not in visited:
            op4 = op+["FILL(1)"]
            visited[(A, tB)] = 1
            queue.append([A, tB, op4])


        u1 = pour(tA,tB,0)
        u2 = pour(tA,tB,1)
        if tuple(u1) not in visited:
            op5=op+["POUR(1,2)"]
            visited[tuple(u1)]=1
            u1.append(op5)
            queue.append(u1)


        if tuple(u2) not in visited:
            op6=op+["POUR(2,1)"]
            visited[tuple(u2)]=1
            u2.append(op6)
            queue.append(u2)
    print("impossible")
bfs(A,B,C)

```

代码运行截图 （AC代码截图，至少包含有"Accepted"）



状态: Accepted

源代码

```
def pour(x,y,flag=0):
    if flag == 0:
        # 0把x导入y中
        if y+x<=B:
            return [0,x+y]
        else:
            return[x+y-B,B]
    else:
        if x+y<=A:
            return[x+y,0]
        else:
            return[A,x+y-A]

A,B,C = map(int,input().split())
from collections import deque
def bfs(A,B,C):
    visited = dict()
    queue = deque([[0,0,[]]])
    while queue:
        s = queue.popleft()
```

05907: 二叉树的操作

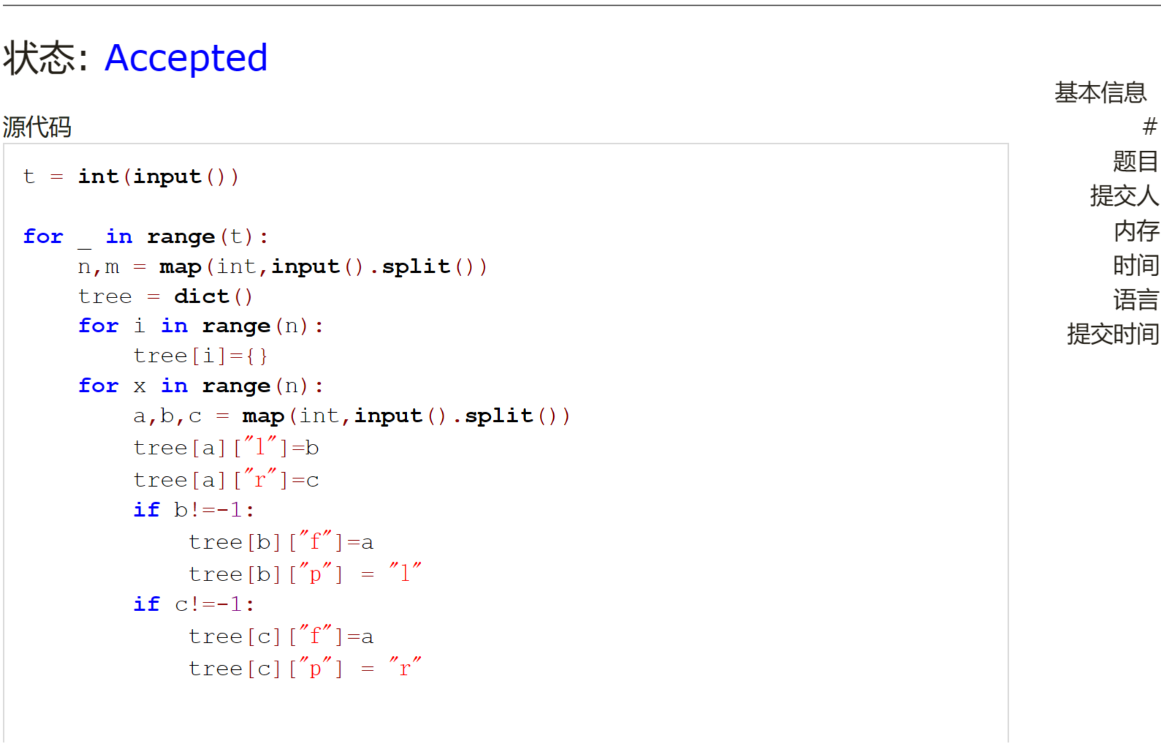
http://cs101.openjudge.cn/practice/05907/

思路：

代码

```python
t = int(input())

for _ in range(t):
    n,m = map(int,input().split())
    tree = dict()
    for i in range(n):
        tree[i]={}
    for x in range(n):
        a,b,c = map(int,input().split())
        tree[a]["l"]=b
        tree[a]["r"]=c
        if b!=-1:
            tree[b]["f"]=a
            tree[b]["p"] = "l"
        if c!=-1:
            tree[c]["f"]=a
            tree[c]["p"] = "r"


    for i in range(m):
        s = list(map(int,input().split()))
        if s[0]==1:
            t1 = tree[s[1]]["f"]
            t2 = tree[s[2]]["f"]
            tree[t1][tree[s[1]]["p"]]=s[2]
            tree[t2][tree[s[2]]["p"]]=s[1]
            tree[s[2]]["p"],tree[s[1]]["p"]=tree[s[1]]["p"],tree[s[2]]["p"]
            tree[s[2]]["f"]=t1
            tree[s[1]]["f"]=t2
        elif s[0]==2:
            t = s[1]
            while tree[t]["l"]!=-1:
                t = tree[t]["l"]
            print(t)

#

```

代码运行截图 （AC代码截图，至少包含有"Accepted"）



```python
t = int(input())

for _ in range(t):
    n,m = map(int,input().split())
    tree = dict()
    for i in range(n):
        tree[i]={}
    for x in range(n):
        a,b,c = map(int,input().split())
        tree[a]["l"]=b
        tree[a]["r"]=c
        if b!=-1:
            tree[b]["f"]=a
            tree[b]["p"] = "l"
        if c!=-1:
            tree[c]["f"]=a
            tree[c]["p"] = "r"
```

18250: 冰阔落 I

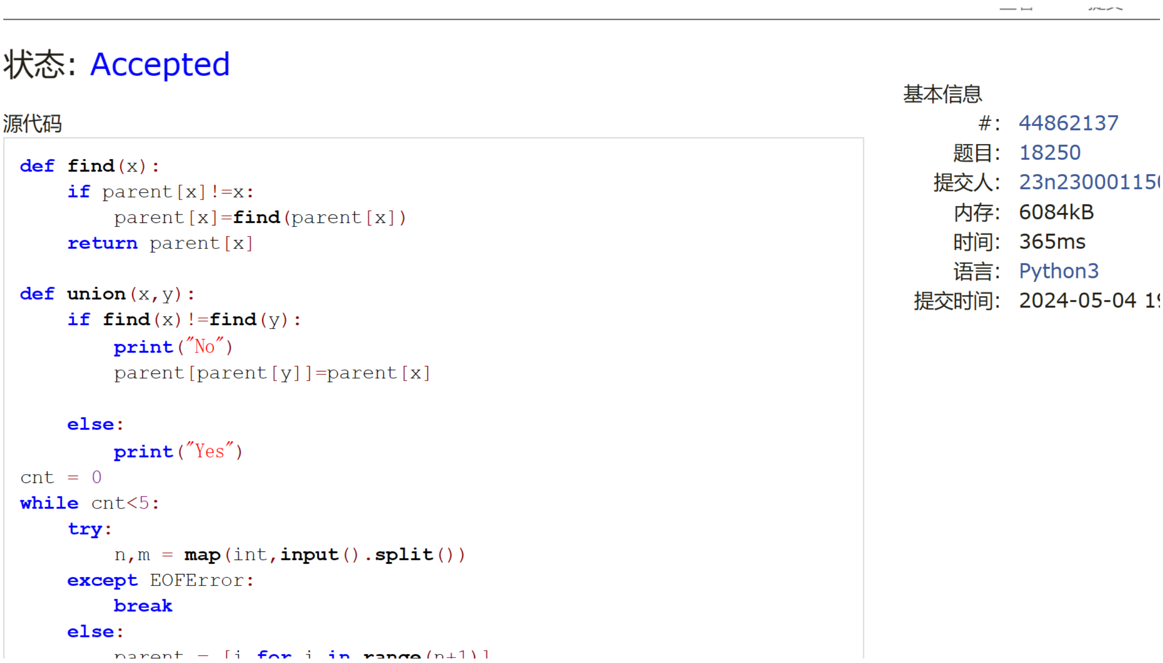
Disjoint set, http://cs101.openjudge.cn/practice/18250/

思路:

代码

```python
#
def find(x):
    if parent[x]!=x:
        parent[x]=find(parent[x])
    return parent[x]

def union(x,y):
    if find(x)!=find(y):
        print("No")
        parent[parent[y]]=parent[x]

    else:
        print("Yes")
cnt = 0
while cnt<5:
    try:
        n,m = map(int,input().split())
    except EOFError:
        break
    else:
        parent = [i for i in range(n+1)]
        for i in range(m):
            x,y = map(int,input().split())
            union(x,y)
        cc ={}
        for i in range(1,n+1):
            if find(i) not in cc:
                cc[parent[i]]=True
        print(len(list(cc.keys())))
        u = list(cc.keys())
        u.sort()
        for x in u:
            print(x,end=" ")
        print("")
        cnt+=1

```

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 parent[x]!=x:
        parent[x]=find(parent[x])
    return parent[x]

def union(x,y):
    if find(x)!=find(y):
        print("No")
        parent[parent[y]]=parent[x]

    else:
        print("Yes")
cnt = 0
while cnt<5:
    try:
        n,m = map(int,input().split())
    except EOFError:
        break
    else:
        parent = [i for i in range(n+1)]
```

基本信息

#:	44862137
题目:	18250
提交人:	23n23000115
内存:	6084kB
时间:	365ms
语言:	Python3
提交时间:	2024-05-04 1

05443: 兔子与樱花

http://cs101.openjudge.cn/practice/05443/

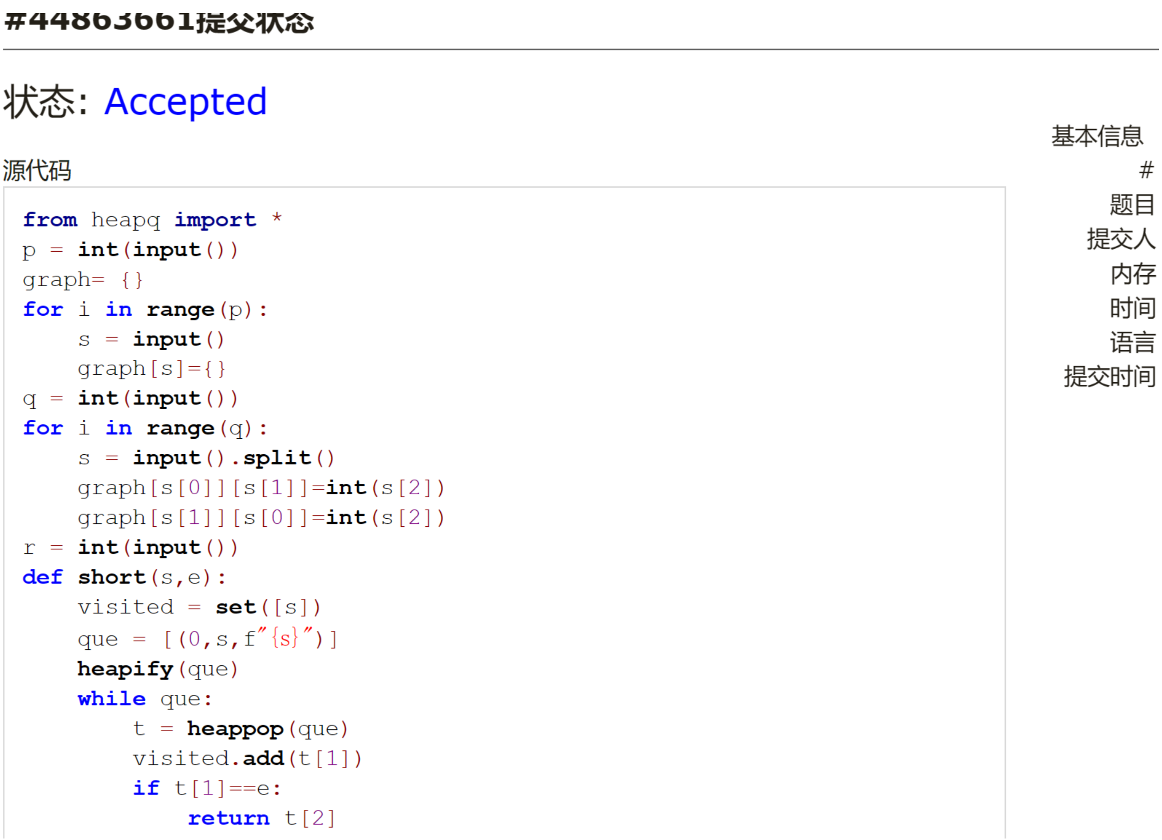
思路:

代码

```
#
from heapq import *
p = int(input())
graph= {}
```

```python
for i in range(p):
    s = input()
    graph[s]={}
q = int(input())
for i in range(q):
    s = input().split()
    graph[s[0]][s[1]]=int(s[2])
    graph[s[1]][s[0]]=int(s[2])
r = int(input())
def short(s,e):
    visited = set([s])
    que = [(0,s,f"{s}")]
    heapify(que)
    while que:
        t = heappop(que)
        visited.add(t[1])
        if t[1]==e:
            return t[2]
        for x in graph[t[1]].keys():
            if x not in visited:
                heappush(que,(t[0]+graph[t[1]][x],x,t[2]+f"->({graph[t[1]][x]})->{x}"))


for i in range(r):
    s,e = input().split()
    t=short(s,e)
    print(t)
```

代码运行截图 （AC代码截图，至少包含有"Accepted"）



#44863661提交状态

状态: Accepted

源代码

```python
from heapq import *
p = int(input())
graph= {}
for i in range(p):
    s = input()
    graph[s]={}
q = int(input())
for i in range(q):
    s = input().split()
    graph[s[0]][s[1]]=int(s[2])
    graph[s[1]][s[0]]=int(s[2])
r = int(input())
def short(s,e):
    visited = set([s])
    que = [(0,s,f"{s}")]
    heapify(que)
    while que:
        t = heappop(que)
        visited.add(t[1])
        if t[1]==e:
            return t[2]
```

基本信息

#

题目

提交人

内存

时间

语言

提交时间

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。

本次作业相对友好