

Assignment #B: 图论和树算

Updated 1709 GMT+8 Apr 28, 2024

2024 spring, Compiled 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

```
1  #
2  Map = [[0]*10 for i in range(10)]
3  for i in range(10):
4      Map[i][:]=list(input())
5
6  dx = [0,0,1,-1]
7  dy = [1,-1,0,0]
8  def dfs(x,y):
9
```

```

10     for i in range(4):
11         tx = x+dx[i]
12         ty = y +dy[i]
13         if 0<=tx<10 and 0<=ty<10 and Map[tx][ty]==".":
14             Map[tx][ty]="_"
15             dfs(tx,ty)
16     return
17
18 cnt = 0
19 for i in range(10):
20     for j in range(10):
21         if Map[i][j]==".":
22             dfs(i,j)
23             cnt+=1
24 print(cnt)
25
26

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

基

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

```

基

02754: 八皇后

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

思路:

代码

1 #

```

2 queen = []
3 M = [[0]*8 for i in range(8)]
4 temp = [ "." for i in range(8)]
5
6 def ccp(i,j):
7     temp[i]=j
8     if i == 7:
9         queen.append([x+1 for x in temp])
10        temp[7]="."
11        return
12    for m in range(8):
13        flag = 0
14        for n in range(i+1):
15            if m == temp[n] or abs(m-temp[n])== (i+1-n):
16                flag = 1
17        if flag ==0:
18            ccp(i+1,m)
19    temp[i]="."
20 for i in range(8):
21     ccp(0,i)
22 t = int(input())
23 for i in range(t):
24     aa = int(input())
25     for i in queen[aa-1]:
26         print(i,end= "")
27     print()

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```

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: 2023-10-10 14:10:10

03151: Pots

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

思路:

代码

```
1  #
2  def pour(x,y,flag=0):
3      if flag == 0:
4          # 0把x导入y中
5          if y+x<=B:
6              return [0,x+y]
7          else:
8              return [x+y-B,B]
9      else:
10         if x+y<=A:
11             return [x+y,0]
12         else:
13             return [A,x+y-A]
14
15 A,B,C = map(int,input().split())
16 from collections import deque
17 def bfs(A,B,C):
18     visited = dict()
19     queue = deque([[0,0,[]]])
20     while queue:
21         s = queue.popleft()
22         tA = s[0]
23         tB = s[1]
24         op = s[2]
25         if C in [tA,tB]:
26             print(len(op))
27             for x in op:
28                 print(x)
29             return
30         if (0,tB) not in visited:
31             op1 = op+["DROP(1)"]
32             visited[(0,tB)]=1
33             queue.append([0,tB,op1])
34
35         if (tA,0) not in visited:
36             op2 = op + ["DROP(2)"]
37             visited[(tA,0)]=1
38             queue.append([tA,0,op2])
39
40         if (tA,B) not in visited:
41             op3=op+["FILL(2)"]
42             visited[(tA,B)]=1
43             queue.append([tA,B,op3])
44
```

```

45
46         if (A, tB) not in visited:
47             op4 = op+["FILL(1)"]
48             visited[(A, tB)] = 1
49             queue.append([A, tB, op4])
50
51
52         u1 = pour(tA,tB,0)
53         u2 = pour(tA,tB,1)
54         if tuple(u1) not in visited:
55             op5=op+["POUR(1,2)"]
56             visited[tuple(u1)]=1
57             u1.append(op5)
58             queue.append(u1)
59
60
61         if tuple(u2) not in visited:
62             op6=op+["POUR(2,1)"]
63             visited[tuple(u2)]=1
64             u2.append(op6)
65             queue.append(u2)
66     print("impossible")
67     bfs(A,B,C)
68

```

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

```
1  t = int(input())
2
3  for _ in range(t):
4      n,m = map(int,input().split())
5      tree = dict()
6      for i in range(n):
7          tree[i]={}
8      for x in range(n):
9          a,b,c = map(int,input().split())
10         tree[a]["l"]=b
11         tree[a]["r"]=c
12         if b!=-1:
13             tree[b]["f"]=a
14             tree[b]["p"] = "l"
15         if c!=-1:
16             tree[c]["f"]=a
17             tree[c]["p"] = "r"
18
19
20     for i in range(m):
21         s = list(map(int,input().split()))
22         if s[0]==1:
23             t1 = tree[s[1]]["f"]
24             t2 = tree[s[2]]["f"]
25             tree[t1][tree[s[1]]["p"]]=s[2]
26             tree[t2][tree[s[2]]["p"]]=s[1]
27             tree[s[2]]["p"],tree[s[1]]["p"]=tree[s[1]]["p"],tree[s[2]]["p"]
28             tree[s[2]]["f"]=t1
29             tree[s[1]]["f"]=t2
30         elif s[0]==2:
31             t = s[1]
32             while tree[t]["l"]!=-1:
33                 t = tree[t]["l"]
34             print(t)
35
36 #
37
```

代码运行截图 (AC代码截图, 至少包含有"Accepted")

状态: Accepted

基本信息

#

题目

提交人

内存

时间

语言

提交时间

源代码

```
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

```
1  #
2  def find(x):
3      if parent[x]!=x:
4          parent[x]=find(parent[x])
5      return parent[x]
6
7  def union(x,y):
8      if find(x)!=find(y):
9          print("No")
10         parent[parent[y]]=parent[x]
11
12     else:
13         print("Yes")
14 cnt = 0
15 while cnt<5:
16     try:
17         n,m = map(int,input().split())
18     except EOFError:
19         break
20     else:
```

```

21     parent = [i for i in range(n+1)]
22     for i in range(m):
23         x,y = map(int,input().split())
24         union(x,y)
25     cc = {}
26     for i in range(1,n+1):
27         if find(i) not in cc:
28             cc[parent[i]] = True
29     print(len(list(cc.keys())))
30     u = list(cc.keys())
31     u.sort()
32     for x in u:
33         print(x,end=" ")
34     print("")
35     cnt+=1
36

```

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: 23n230001150
 内存: 6084kB
 时间: 365ms
 语言: Python3
 提交时间: 2024-05-04 19:00:00

05443: 兔子与樱花

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

思路:

代码

```

1  #
2  from heapq import *
3  p = int(input())
4  graph = {}

```



```

5  for i in range(p):
6      s = input()
7      graph[s]={}
8  q = int(input())
9  for i in range(q):
10     s = input().split()
11     graph[s[0]][s[1]]=int(s[2])
12     graph[s[1]][s[0]]=int(s[2])
13  r = int(input())
14  def short(s,e):
15     visited = set([s])
16     que = [(0,s,f"{s}")]
17     heapify(que)
18     while que:
19         t = heappop(que)
20         visited.add(t[1])
21         if t[1]==e:
22             return t[2]
23         for x in graph[t[1]].keys():
24             if x not in visited:
25                 heappush(que,(t[0]+graph[t[1]][x],x,t[2]+f"->({graph[t[1]]
[x]})->{x}"))
26
27
28  for i in range(r):
29     s,e = input().split()
30     t=short(s,e)
31     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

```

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