

Assignment #B: 图论和树算

Updated 1709 GMT+8 Apr 28, 2024

2024 spring, Compiled by 郑铭毅 数学科学学院

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

Windows 11

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

```

def count_eagles(board):
    rows = 10
    cols = 10

    directions = [(1, 0), (-1, 0), (0, 1), (0, -1)]

    def dfs(row, col):
        board[row][col] = '-'

        for dr, dc in directions:
            nr, nc = row + dr, col + dc

            if 0 <= nr < rows and 0 <= nc < cols and board[nr][nc] == '.':
                dfs(nr, nc)

    count = 0

    for i in range(rows):
        for j in range(cols):
            if board[i][j] == '.':
                dfs(i, j)
                count += 1

    return count

board = []

for _ in range(10):
    row = input().strip()
    board.append(list(row))

print(count_eagles(board))

```

代码运行截图

The screenshot shows the OpenJudge CS101 problem page. The submission status is "Accepted". The source code is displayed on the left, and the basic information is on the right.

源代码

```

def count_eagles(board):
    rows = 10
    cols = 10
    directions = [(1, 0), (-1, 0), (0, 1), (0, -1)]
    def dfs(row, col):
        board[row][col] = '-'
        for dr, dc in directions:
            nr, nc = row + dr, col + dc
            if 0 <= nr < rows and 0 <= nc < cols and board[nr][nc] == '.':
                dfs(nr, nc)
    count = 0
    for i in range(rows):
        for j in range(cols):
            if board[i][j] == '.':
                dfs(i, j)
                count += 1
    return count
board = []
for _ in range(10):
    row = input().strip()
    board.append(list(row))
print(count_eagles(board))

```

基本信息

- #: 4488830
- 题目: 28170
- 提交人: 2300010872
- 内存: 3660KB
- 时间: 21ms
- 语言: Python3
- 提交时间: 2024-05-07 16:26:08

©2002-2022 POJ 京ICP备20010980号-1 English 帮助 关于

02754: 八皇后

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

思路:

代码

```
result=[0]*8
s=[]
def isok(n,pos):
    for i in range(n):
        if result[i]==pos or abs(i-n)==abs(result[i]-pos):
            return False
    return True
def queen(i):
    global s
    if i==8:
        a=''
        for k in range(8):
            a+=str(result[k]+1)
        s.append(a)
        return
    for j in range(8):
        if isok(i,j):
            result[i]=j
            queen(i+1)
queen(0)
S=[int(i) for i in s]
S.sort()
n=int(input())
for i in range(n):
    w=int(input())
    print(s[w-1])
```

代码运行截图

OpenJudge 题目ID, 标题, 描述 2300010872 信箱 账号

CS101 / 题库 题目 排名 状态 提问

#44888421提交状态 查看 提交 统计 提问

状态: Accepted

源代码

```
result=[0]*8
s=[]
def isok(n,pos):
    for i in range(n):
        if result[i]==pos or abs(i-n)==abs(result[i]-pos):
            return False
    return True
def queen(i):
    global s
    if i==8:
        a=''
        for k in range(8):
            a+=str(result[k]+1)
        s.append(a)
        return
    for j in range(8):
        if isok(i,j):
            result[i]=j
            queen(i+1)
queen(0)
S=[int(i) for i in s]
S.sort()
n=int(input())
for i in range(n):
    w=int(input())
    print(s[w-1])
```

基本信息

```
#: 44888421
题目: 02754
提交人: 2300010872
内存: 3636kB
时间: 30ms
语言: Python3
提交时间: 2024-05-07 15:57:38
```

©2002-2022 POJ 京ICP备20010980号-1 English 帮助 关于

03151: Pots

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

思路:

代码

```

from collections import deque

def pour_water(a, b, c):
    visited = set()
    queue = deque([(0, 0, 0, [])])
    visited.add((0, 0))
    while queue:
        wa, wb, last_op, steps = queue.popleft()
        if wa == c or wb == c:
            return steps
        if wa < a and (a, wb) not in visited:
            visited.add((a, wb))
            queue.append((a, wb, 1, steps + ['FILL(1)']))
        if wb < b and (wa, b) not in visited:
            visited.add((wa, b))
            queue.append((wa, b, 2, steps + ['FILL(2)']))
        if wa > 0 and (0, wb) not in visited:
            visited.add((0, wb))
            queue.append((0, wb, 3, steps + ['DROP(1)']))
        if wb > 0 and (wa, 0) not in visited:
            visited.add((wa, 0))
            queue.append((wa, 0, 4, steps + ['DROP(2)']))
        if wa > 0 and wb < b:
            pour_amt = min(wa, b - wb)
            if (wa - pour_amt, wb + pour_amt) not in visited:
                visited.add((wa - pour_amt, wb + pour_amt))
                queue.append((wa - pour_amt, wb + pour_amt, 5, steps + ['POUR(1,2)']))
        if wb > 0 and wa < a:
            pour_amt = min(wb, a - wa)
            if (wa + pour_amt, wb - pour_amt) not in visited:
                visited.add((wa + pour_amt, wb - pour_amt))
                queue.append((wa + pour_amt, wb - pour_amt, 6, steps + ['POUR(2,1)']))
    return ["impossible"]

a, b, c = map(int, input().split())
result = pour_water(a, b, c)
if result[0] == "impossible":
    print(result[0])
else:
    print(len(result))
    for step in result:
        print(step)

```

代码运行截图



05907: 二叉树的操作

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

思路:

代码

```
1  #
2
```

代码运行截图 (AC代码截图, 至少包含有"Accepted")

18250: 冰阔落 I

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

思路:

```
def find(x):
    if uf[x]!=x:
        uf[x]=find(uf[x])
    return uf[x]
def union(x,y):
    x,y=find(x),find(y)
    if x==y:
        print('Yes')
    else:
        print('No')
        uf[y]=x
while True:
    try:
        n,m=map(int,input().split())
        uf=[i for i in range(n+1)]
        for _ in range(m):
            x,y=map(int,input().split())
            union(x,y)
        ans,num=[],0
        for i in range(1,n+1):
            if find(i)==i:
                ans.append(i)
                num+=1
        print(num)
        print(*ans)
    except EOFError:
        break
```

代码运行截图

CS101 / 题库

题目 排名 状态 提问

#44890475提交状态

查看 提交 统计 撤回

状态: Accepted

源代码

```
def find(x):
    if uf[x] != x:
        uf[x] = find(uf[x])
    return uf[x]
def union(x, y):
    x, y = find(x), find(y)
    if x == y:
        print('Yes')
    else:
        print('No')
    uf[y] = x
while True:
    try:
        n, m = map(int, input().split())
        uf = [i for i in range(n+1)]
        for _ in range(m):
            x, y = map(int, input().split())
            union(x, y)
        ans, num = [], 0
        for i in range(1, n+1):
            if find(i) == i:
                ans.append(i)
                num += 1
        print(num)
        print('ans')
    except EOFError:
        break
```

基本信息

#: 44890475

题目: 18250

提交人: 2300010872

内存: 5668kB

时间: 364ms

语言: Python3

提交时间: 2024-05-07 19:35:14

©2002-2022 POJ 前ICPC备20010980号-1

English 帮助 关于

05443: 兔子与樱花

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

思路:

代码


```
1 #  
2
```

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

逐渐熟悉了关于bfs和dfs的一些算法, 当然也还有很多需要提升的地方。