

Assignment B

姓名: 尹廖丹华

学号: 2000092113

院系: 心理与认知科学学院

一、作业题目

1、28170: 算鹰

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

代码:

```
def dfs(x,y):
    graph[x][y] = "-"
    for dx,dy in [(1,0),(-1,0),(0,1),(0,-1)]:
        if 0<=x+dx<10 and 0<=y+dy<10 and graph[x+dx][y+dy] == ".":
            dfs(x+dx,y+dy)
graph = []
result = 0
for i in range(10):
    graph.append(list(input()))
for i in range(10):
    for j in range(10):
        if graph[i][j] == ".":
            result += 1
            dfs(i,j)
print(result)
```

OpenJudge

题目ID, 标题, 描述

22n2000092113

信箱

账号

CS101 / 题库

题目

排名

状态

提问

#44902302提交状态

查看

提交

统计

提问

状态: Accepted

源代码

```
def dfs(x,y):
    graph[x][y] = "-"
    for dx,dy in [(1,0),(-1,0),(0,1),(0,-1)]:
        if 0<=x+dx<10 and 0<=y+dy<10 and graph[x+dx][y+dy] == ".":
            dfs(x+dx,y+dy)
graph = []
result = 0
for i in range(10):
    graph.append(list(input()))
for i in range(10):
    for j in range(10):
        if graph[i][j] == ".":
            result += 1
            dfs(i,j)
print(result)
```

基本信息

#: 44902302

题目: 28170

提交人: 22n2000092113

内存: 3604kB

时间: 20ms

语言: Python3

提交时间: 2024-05-08 19:56:53

©2002-2022 POJ 京ICP备20010980号-1

English

帮助

关于

2、03151: Pots

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

代码:

```
def bfs(A, B, C):
    start = (0, 0)
    visited = set()
    visited.add(start)
    queue = [(start, [])]

    while queue:
        (a, b), actions = queue.pop(0)

        if a == C or b == C:
            return actions

        next_states = [(A, b), (a, B), (0, b), (a, 0), (min(a + b, A), \
            max(0, a + b - A)), (max(0, a + b - B), min(a + b, B))]

        for i in next_states:
            if i not in visited:
                visited.add(i)
                new_actions = actions + [get_action(a, b, i)]
                queue.append((i, new_actions))

    return ["impossible"]

def get_action(a, b, next_state):
    if next_state == (A, b):
        return "FILL(1)"
    elif next_state == (a, B):
        return "FILL(2)"
    elif next_state == (0, b):
        return "DROP(1)"
    elif next_state == (a, 0):
        return "DROP(2)"
    elif next_state == (min(a + b, A), max(0, a + b - A)):
        return "POUR(2,1)"
    else:
        return "POUR(1,2)"

A, B, C = map(int, input().split())
solution = bfs(A, B, C)

if solution == ["impossible"]:
    print(solution[0])
else:
    print(len(solution))
    for i in solution:
        print(i)
```

#44902328提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
def bfs(A, B, C):
    start = (0, 0)
    visited = set()
    visited.add(start)
    queue = [(start, [])]

    while queue:
        (a, b), actions = queue.pop(0)

        if a == C or b == C:
            return actions

        next_states = [(A, b), (a, B), (0, b), (a, 0), (min(a + b, A), \
            max(0, a + b - A)), (max(0, a + b - B), min(a + b, B))]

        for i in next_states:
            if i not in visited:
```

基本信息

#: 44902328
题目: 03151
提交人: 22n2000092113
内存: 3712kB
时间: 22ms
语言: Python3
提交时间: 2024-05-08 19:58:12

05907: 二叉树的操作

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

代码:

```
def swap(x, y):
    tree[loc[x][0]][loc[x][1]] = y
    tree[loc[y][0]][loc[y][1]] = x
    loc[x], loc[y] = loc[y], loc[x]

for _ in range(int(input())):
    n, m = map(int, input().split())
    tree = {}
    loc = [[] for _ in range(n)]
    for _ in range(n):
        a, b, c = map(int, input().split())
        tree[a] = [b, c]
        loc[b], loc[c] = [a, 0], [a, 1]
    for _ in range(m):
        op = list(map(int, input().split()))
        if op[0] == 1:
            swap(op[1], op[2])
        else:
            cur = op[1]
            while tree[cur][0] != -1:
                cur = tree[cur][0]
            print(cur)
```

#44902369提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
def swap(x, y):
    tree[loc[x][0]][loc[x][1]] = y
    tree[loc[y][0]][loc[y][1]] = x
    loc[x], loc[y] = loc[y], loc[x]

for _ in range(int(input())):
    n, m = map(int, input().split())
    tree = {}
    loc = [[] for _ in range(n)]
    for _ in range(n):
        a, b, c = map(int, input().split())
        tree[a] = [b, c]
        loc[b], loc[c] = [a, 0], [a, 1]
    for _ in range(m):
        op = list(map(int, input().split()))
        if op[0] == 1:
```

基本信息

#: 44902369
题目: 05907
提交人: 22n2000092113
内存: 3820kB
时间: 73ms
语言: Python3
提交时间: 2024-05-08 19:59:49

18250: 冰阔落 I

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

代码:

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

def union(x, y):
    root_x = find(x)
    root_y = find(y)
    if root_x != root_y:
        parent[root_y] = root_x

while True:
    try:
        n, m = map(int, input().split())
        parent = list(range(n + 1))

        for _ in range(m):
            a, b = map(int, input().split())
            if find(a) == find(b):
                print('Yes')
            else:
                print('No')
                union(a, b)

        unique_parents = set(find(x) for x in range(1, n + 1))
        ans = sorted(unique_parents)
        print(len(ans))
        print(*ans)

    except EOFError:
        break
```

#44902401提交状态

查看 提交 统计 提问

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):
    root_x = find(x)
    root_y = find(y)
    if root_x != root_y:
        parent[root_y] = root_x

while True:
    try:
        n, m = map(int, input().split())
        parent = list(range(n + 1))

        for _ in range(m):
```

基本信息

#: 44902401
题目: 18250
提交人: 22n2000092113
内存: 6100kB
时间: 387ms
语言: Python3
提交时间: 2024-05-08 20:01:09

二、 学习总结和收获

假期的作业实在是太多了，所以作业没能及时上交，非常抱歉，还花了很长时间理解，感觉知识点都很抽象 T-T，希望能快点赶上进度。