Assignment #10: dp & bfs

Updated 2 GMT+8 Nov 25, 2024

2024 fall, Complied by 任宇桐 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。

1. 题目

LuoguP1255 数楼梯

dp, bfs, https://www.luogu.com.cn/problem/P1255

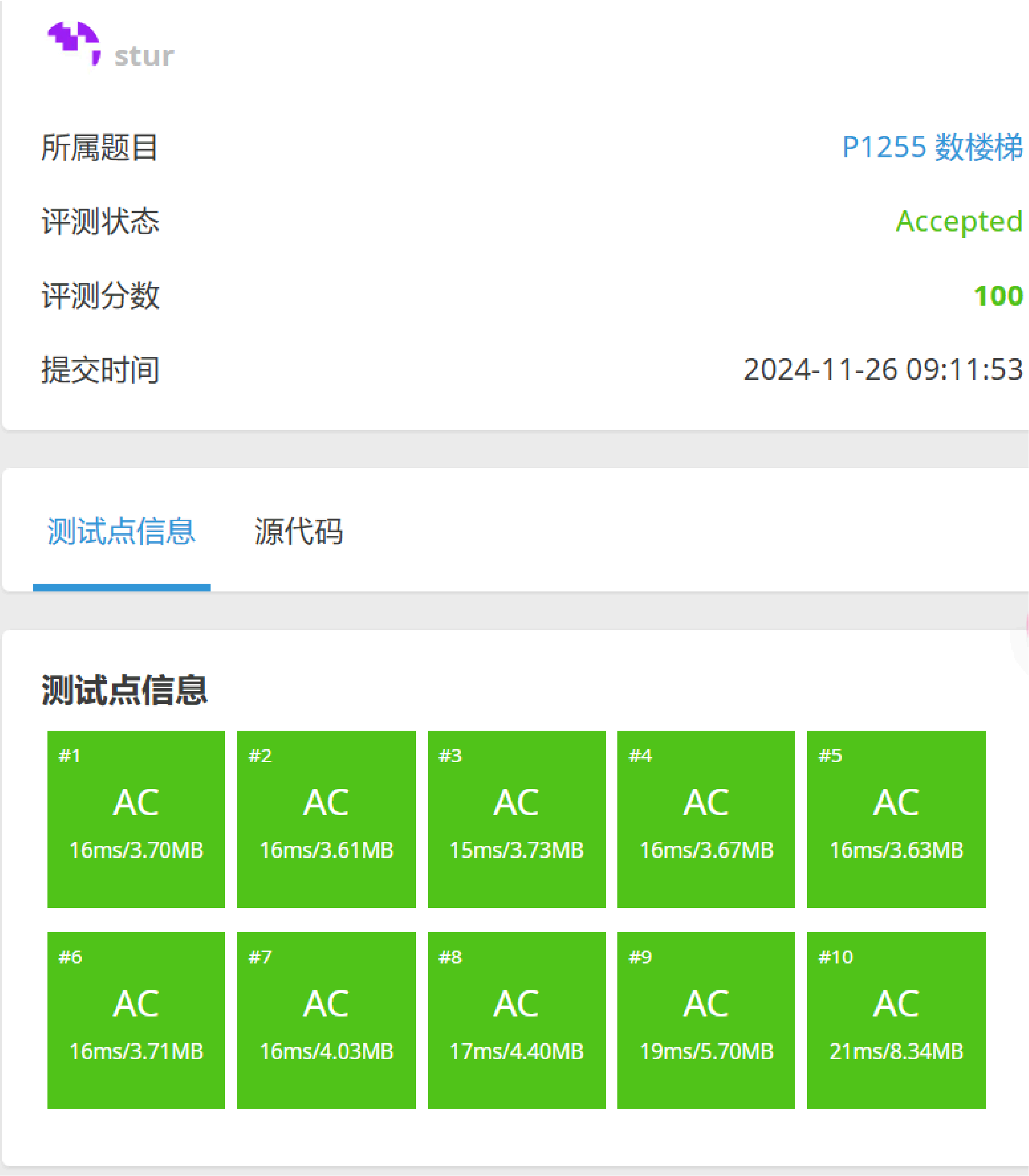
思路：

如果不使用优化直接递归应该不能通过（？）

代码：

```python
import sys
from functools import lru_cache
sys.setrecursionlimit(1<<30)
n = int(input())
@lru_cache(maxsize=None)
def f(n):
    if n == 0:
        return 1
    if n == 1:
        return 1
    return f(n-1)+f(n-2)
print(f(n))
```

代码运行截图 （至少包含有"Accepted"）



所属题目	P1255 数楼梯
评测状态	Accepted
评测分数	100
提交时间	2024-11-26 09:11:53

测试点信息

#1 AC 16ms/3.70MB	#2 AC 16ms/3.61MB	#3 AC 15ms/3.73MB	#4 AC 16ms/3.67MB	#5 AC 16ms/3.63MB
#6 AC 16ms/3.71MB	#7 AC 16ms/4.03MB	#8 AC 17ms/4.40MB	#9 AC 19ms/5.70MB	#10 AC 21ms/8.34MB

27528: 跳台阶

dp, http://cs101.openjudge.cn/practice/27528/

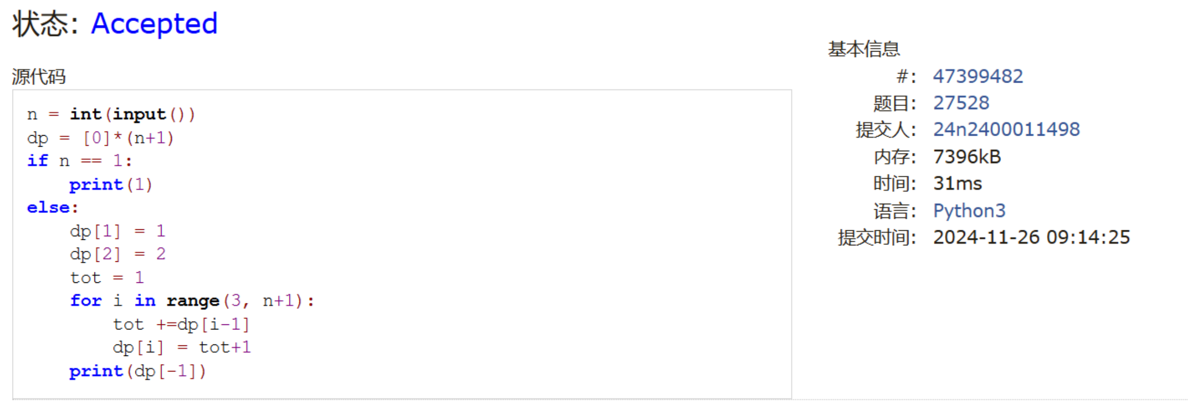
思路：

注意与上题不同，需要加上前面所有的情形。

代码：

```
n = int(input())
dp = [0]*(n+1)
if n == 1:
    print(1)
else:
    dp[1] = 1
    dp[2] = 2
    tot = 1
    for i in range(3, n+1):
        tot +=dp[i-1]
        dp[i] = tot+1
    print(dp[-1])
```

代码运行截图 （至少包含有"Accepted"）



状态: Accepted

源代码

```
n = int(input())
dp = [0]*(n+1)
if n == 1:
    print(1)
else:
    dp[1] = 1
    dp[2] = 2
    tot = 1
    for i in range(3, n+1):
        tot +=dp[i-1]
        dp[i] = tot+1
    print(dp[-1])
```

基本信息

#:	47399482
题目:	27528
提交人:	24n2400011498
内存:	7396kB
时间:	31ms
语言:	Python3
提交时间:	2024-11-26 09:14:25

474D. Flowers

dp, https://codeforces.com/problemset/problem/474/D

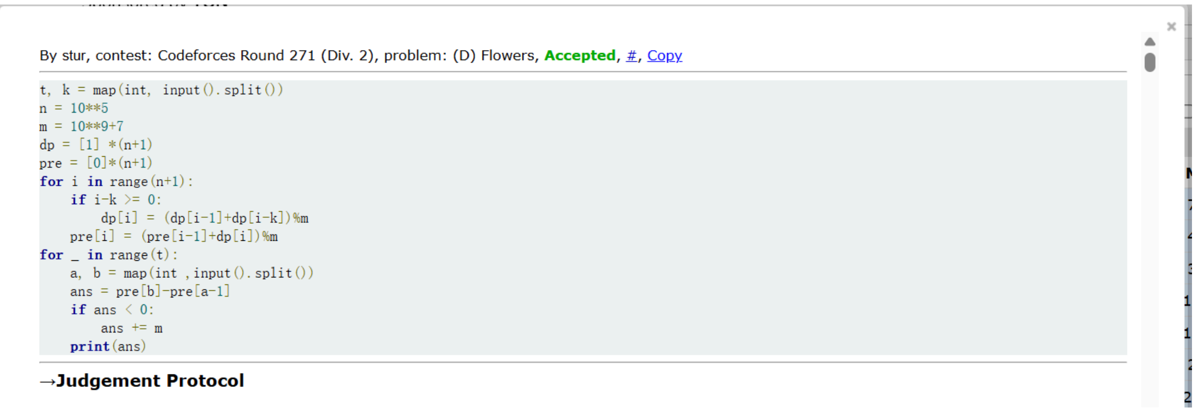
思路：

一开始没有看懂题目，不知道a, b是什么意思，看懂以后就简单多了。按照题目的提示dp即可，但是应该要反复取模，否则会超时。

代码：

```
t, k = map(int, input().split())
n = 10**5
m = 10**9+7
dp = [1] *(n+1)
pre = [0]*(n+1)
for i in range(n+1):
    if i-k >= 0:
        dp[i] = (dp[i-1]+dp[i-k])%m
    pre[i] = (pre[i-1]+dp[i])%m
for _ in range(t):
    a, b = map(int ,input().split())
    ans = pre[b]-pre[a-1]
    if ans < 0:
        ans += m
    print(ans)
```

代码运行截图 （至少包含有"Accepted"）



By stur, contest: Codeforces Round 271 (Div. 2), problem: (D) Flowers, **Accepted**, #, Copy

```
t, k = map(int, input().split())
n = 10**5
m = 10**9+7
dp = [1] *(n+1)
pre = [0]*(n+1)
for i in range(n+1):
    if i-k >= 0:
        dp[i] = (dp[i-1]+dp[i-k])%m
    pre[i] = (pre[i-1]+dp[i])%m
for _ in range(t):
    a, b = map(int ,input().split())
    ans = pre[b]-pre[a-1]
    if ans < 0:
        ans += m
    print(ans)
```

→**Judgement Protocol**

LeetCode5.最长回文子串

dp, two pointers, string, https://leetcode.cn/problems/longest-palindromic-substring/

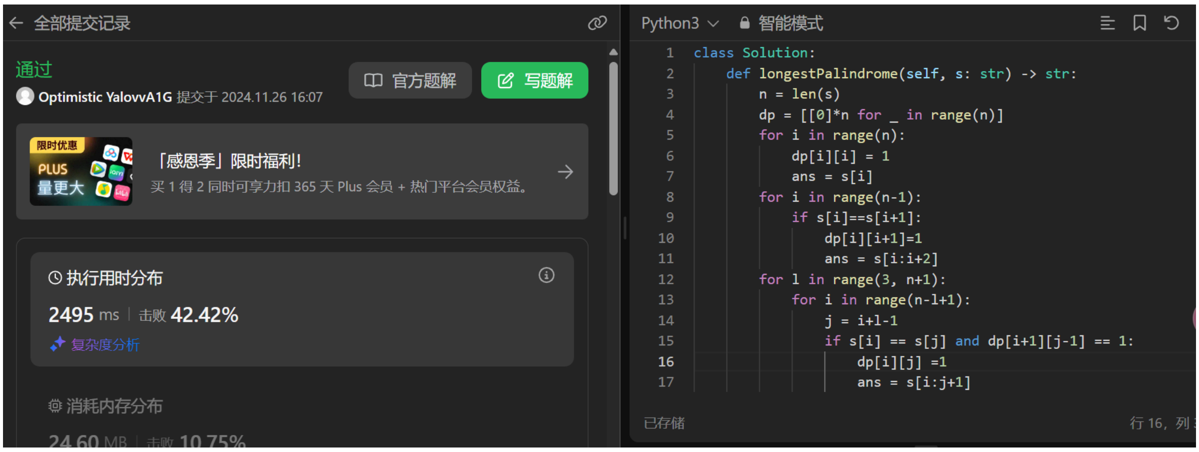
思路：

自己大概能意识到需要讨论length-2的dp，但是开始时怎么也不知道如何处理length=1 and length=2的情形，参考了晴问上的类似题目才知道如何处理。

代码：

```
class Solution:
    def longestPalindrome(self, s: str) -> str:
        n = len(s)
        dp = [[0]*n for _ in range(n)]
        for i in range(n):
            dp[i][i] = 1
            ans = s[i]
        for i in range(n-1):
            if s[i]==s[i+1]:
                dp[i][i+1]=1
                ans = s[i:i+2]
        for l in range(3, n+1):
            for i in range(n-l+1):
                j = i+l-1
                if s[i] == s[j] and dp[i+1][j-1] == 1:
                    dp[i][j] =1
                    ans = s[i:j+1]
        return ans
```

代码运行截图 （至少包含有"Accepted"）



通过

Optimistic YalovvA1G 提交于 2024.11.26 16:07

「感恩季」限时福利！
买 1 得 2 同时可享力扣 365 天 Plus 会员 + 热门平台会员权益。

执行用时分布

2495 ms | 击败 42.42%

复杂度分析

消耗内存分布

24.60 MB | 击败 10.75%

```python
class Solution:
    def longestPalindrome(self, s: str) -> str:
        n = len(s)
        dp = [[0]*n for _ in range(n)]
        for i in range(n):
            dp[i][i] = 1
            ans = s[i]
        for i in range(n-1):
            if s[i]==s[i+1]:
                dp[i][i+1]=1
                ans = s[i:i+2]
        for l in range(3, n+1):
            for i in range(n-l+1):
                j = i+l-1
                if s[i] == s[j] and dp[i+1][j-1] == 1:
                    dp[i][j] =1
                    ans = s[i:j+1]
```

已存储　行 16，列

12029: 水淹七军

bfs, dfs, http://cs101.openjudge.cn/practice/12029/

用时：60min

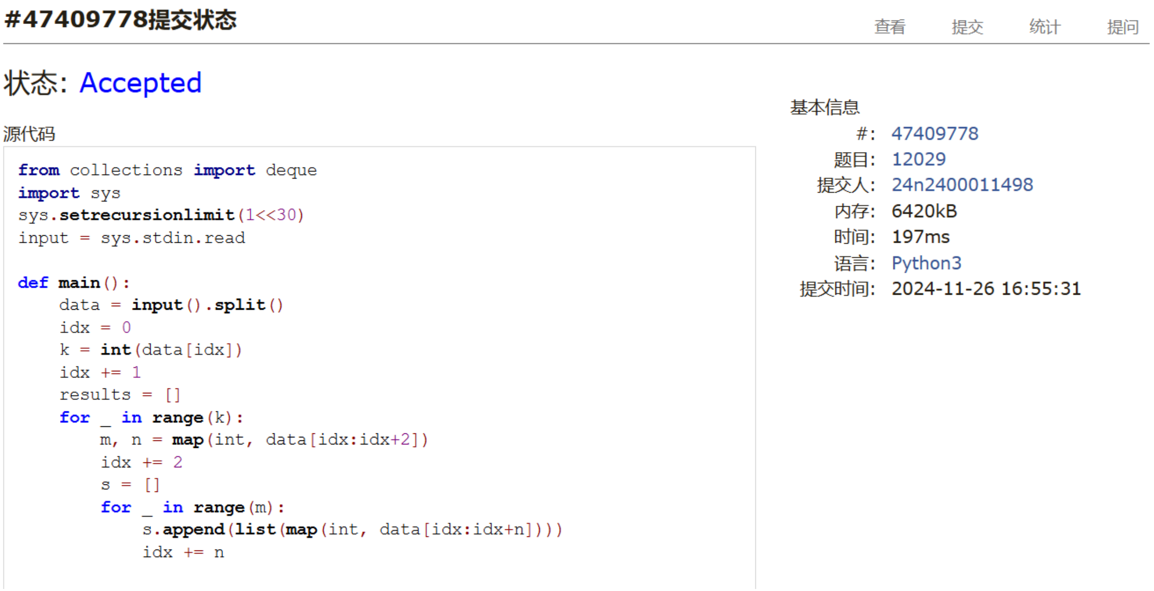
思路：

这道题目初看没有什么难度，做起来被难了很久...首先是数据读取的问题，半天没有弄懂为什么RE，然后查找了聊天记录才明白，不小心看到了上下文提到的坑点，不然还不知道需要多少时间才能AC...

代码：

```python
from collections import deque
import sys
sys.setrecursionlimit(1<<30)
input = sys.stdin.read

def main():
    data = input().split()
    idx = 0
    k = int(data[idx])
    idx += 1
    results = []
    for _ in range(k):
        m, n = map(int, data[idx:idx+2])
        idx += 2
        s = []
        for _ in range(m):
            s.append(list(map(int, data[idx:idx+n])))
            idx += n

        i, j = map(int, data[idx:idx+2])
        idx += 2
        i, j = i-1, j-1

        p = int(data[idx])
        idx += 1
        inQueue = [[False]*n for _ in range(m)]
        q = deque([])
        for _ in range(p):
            x, y = map(int, data[idx:idx+2])
            idx += 2
            x -= 1
            y -= 1
            inQueue[x][y] = True
            q.append((x, y))

        def scope(a, b):
            return  0<=a<m and 0<=b<n

        while q:
            l = len(q)
            for _ in range(l):
                x, y = q.popleft()
                for dx, dy in [(-1, 0), (1, 0), (0, 1), (0, -1)]:
                    nx = x+dx
                    ny = y+dy
                    if scope(nx, ny):
                        if s[nx][ny] < s[x][y]:
                            q.append((nx, ny))
                            s[nx][ny] = s[x][y]
                            inQueue[nx][ny] = True

        results.append('Yes' if inQueue[i][j] else 'No')
    sys.stdout.write('\n'.join(results)+'\n')
if __name__ == '__main__':
    main()
```

代码运行截图 （至少包含有"Accepted"）



#47409778提交状态

状态: Accepted

源代码

```
from collections import deque
import sys
sys.setrecursionlimit(1<<30)
input = sys.stdin.read

def main():
    data = input().split()
    idx = 0
    k = int(data[idx])
    idx += 1
    results = []
    for _ in range(k):
        m, n = map(int, data[idx:idx+2])
        idx += 2
        s = []
        for _ in range(m):
            s.append(list(map(int, data[idx:idx+n])))
            idx += n
```

基本信息

#:	47409778
题目:	12029
提交人:	24n2400011498
内存:	6420kB
时间:	197ms
语言:	Python3
提交时间:	2024-11-26 16:55:31

02802: 小游戏

bfs, http://cs101.openjudge.cn/practice/02802/

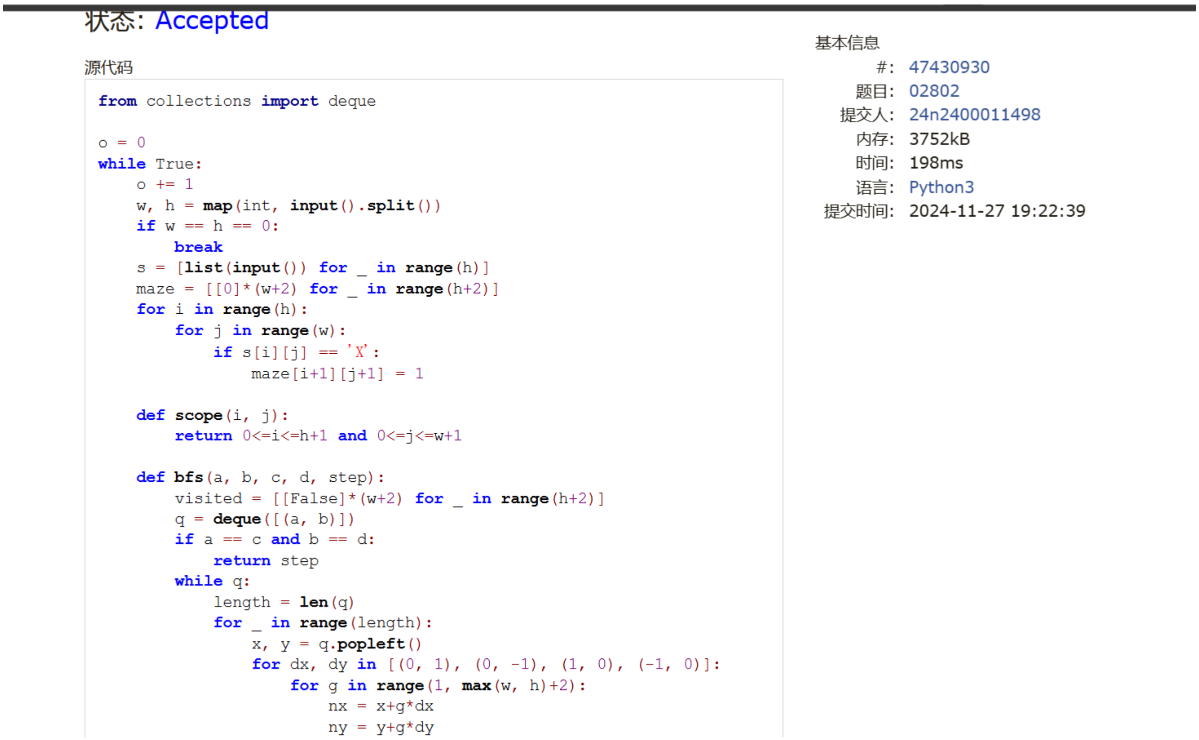
思路：

一开始没有看到tag，然后直接想dfs，无论如何不能AC，然后才开始想bfs，因为和模板有一些不同之处，所以小错误不断，最后求助以后才成功AC。基本思路是，先尝试沿着某个方向尝试所有可能的步长，加到同一层中，然后进行搜索。

代码：

```
from collections import deque

o = 0
while True:
    o += 1
    w, h = map(int, input().split())
    if w == h == 0:
        break
    s = [list(input()) for _ in range(h)]
    maze = [[0]*(w+2) for _ in range(h+2)]
    for i in range(h):
        for j in range(w):
            if s[i][j] == 'X':
                maze[i+1][j+1] = 1

    def scope(i, j):
        return 0<=i<=h+1 and 0<=j<=w+1

    def bfs(a, b, c, d, step):
        visited = [[False]*(w+2) for _ in range(h+2)]
        q = deque([(a, b)])
        if a == c and b == d:
            return step
        while q:
            length = len(q)
            for _ in range(length):
                x, y = q.popleft()
                for dx, dy in [(0, 1), (0, -1), (1, 0), (-1, 0)]:
                    for g in range(1, max(w, h)+2):
                        nx = x+g*dx
                        ny = y+g*dy
                        if nx == c and ny == d:
                            return step+1
                        if not scope(nx, ny):
                            break
                        if maze[nx][ny] == 1:
                            break
                        if scope(nx, ny) and maze[nx][ny] == 0 and not
visited[nx][ny]:
                            visited[nx][ny] = True
                            q.append((nx, ny))

            step += 1

    print(f'Board #{o}:')
    t = 0
    while True:
        t += 1
        x1, y1, x2, y2 = map(int, input().split())
        if x1 == y1 == x2 == y2 == 0:
            print()
            break
        ans = bfs(y1, x1, y2, x2, 0)
        if ans != None:
            print(f'Pair {t}: {ans} segments.')
        else:
            print(f'Pair {t}: impossible.')
```

代码运行截图 （至少包含有"Accepted"）



状态: Accepted

源代码

```
from collections import deque

o = 0
while True:
    o += 1
    w, h = map(int, input().split())
    if w == h == 0:
        break
    s = [list(input()) for _ in range(h)]
    maze = [[0]*(w+2) for _ in range(h+2)]
    for i in range(h):
        for j in range(w):
            if s[i][j] == 'X':
                maze[i+1][j+1] = 1

    def scope(i, j):
        return 0<=i<=h+1 and 0<=j<=w+1

    def bfs(a, b, c, d, step):
        visited = [[False]*(w+2) for _ in range(h+2)]
        q = deque([(a, b)])
        if a == c and b == d:
            return step
        while q:
            length = len(q)
            for _ in range(length):
                x, y = q.popleft()
                for dx, dy in [(0, 1), (0, -1), (1, 0), (-1, 0)]:
                    for g in range(1, max(w, h)+2):
                        nx = x+g*dx
                        ny = y+g*dy
```

基本信息

#:	47430930
题目:	02802
提交人:	24n2400011498
内存:	3752kB
时间:	198ms
语言:	Python3
提交时间:	2024-11-27 19:22:39

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“计概2024fall每日选做”、CF、LeetCode、洛谷等网站题目。

发现作业难度方差较大，前两题非常简单，快速AC，第三题感觉题目的意思表述的不是很能搞明白（也许是英文表述所以理解存在差异？），第4题的dp如果没有见过感觉有一定思维难度，5、6题看起来是模板，但是变数还是有不少的，发现如果能够理解模板背后的逻辑，应对变式还是比较容易写出思路正确的代码的， 但是关于种种细节还是需要勤加练习才能掌握。另外，感觉不能一直在一道题目上死磕，有时就是想法不对（比如最后一题的dfs），花了太多时间，导致每日选做拉下很多没有时间思考。