Assignment #A: dp & bfs

Updated 2 GMT+8 Nov 25, 2024

2024 fall, Complied by 陈张涵、工学院

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：简单dp

代码：

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

代码运行截图 （至少包含有"Accepted"）

题目提供者 洛谷

难度 普及−

历史分数 100

27528: 跳台阶

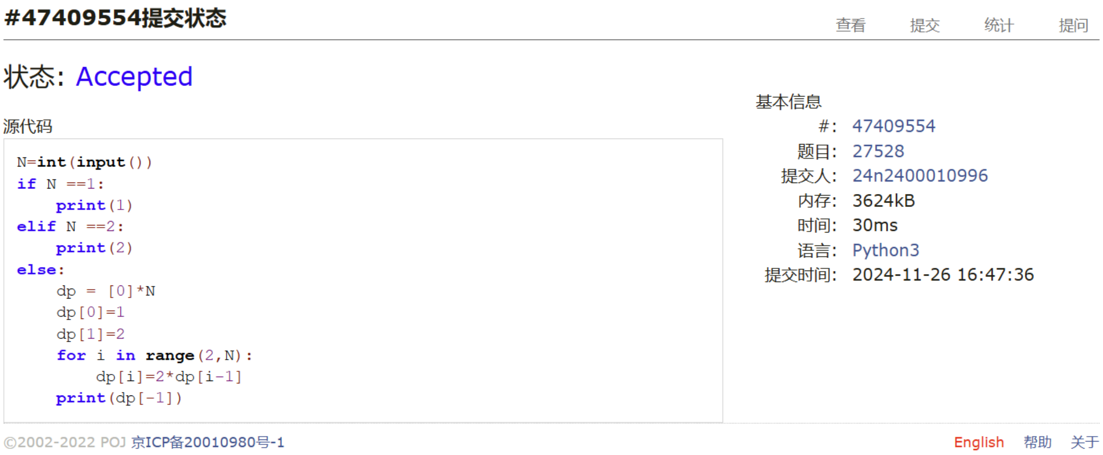
dp, http://cs101.openjudge.cn/practice/27528/

思路：简单dp

代码：

```
N=int(input())
if N ==1:
    print(1)
elif N ==2:
    print(2)
else:
    dp = [0]*N
    dp[0]=1
    dp[1]=2
    for i in range(2,N):
        dp[i]=2*dp[i-1]
    print(dp[-1])
```

代码运行截图 ==（至少包含有"Accepted"）==



#47409554提交状态

状态: Accepted

源代码

```
N=int(input())
if N ==1:
    print(1)
elif N ==2:
    print(2)
else:
    dp = [0]*N
    dp[0]=1
    dp[1]=2
    for i in range(2,N):
        dp[i]=2*dp[i-1]
    print(dp[-1])
```

基本信息

#:	47409554
题目:	27528
提交人:	24n2400010996
内存:	3624kB
时间:	30ms
语言:	Python3
提交时间:	2024-11-26 16:47:36

©2002-2022 POJ 京ICP备20010980号-1

474D. Flowers

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

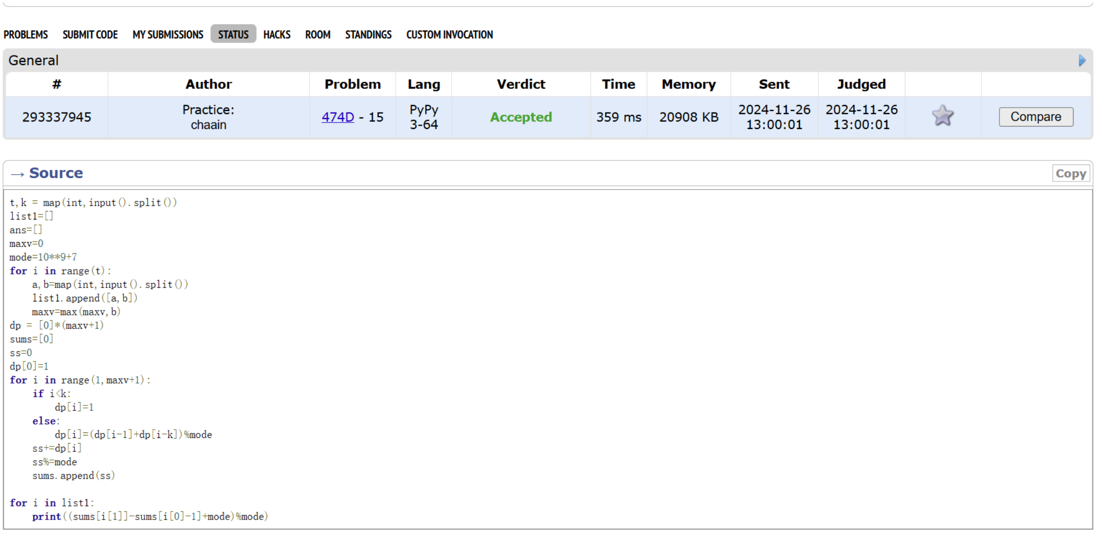
思路：dp 但稍微有点难想，不超时还需要一些小技巧

代码：

```
t,k = map(int,input().split())
list1=[]
ans=[]
maxv=0
mode=10**9+7
for i in range(t):
    a,b=map(int,input().split())
    list1.append([a,b])
    maxv=max(maxv,b)
dp = [0]*(maxv+1)
sums=[0]
ss=0
dp[0]=1
for i in range(1,maxv+1):
    if i<k:
        dp[i]=1
    else:
        dp[i]=(dp[i-1]+dp[i-k])%mode
    ss+=dp[i]
    ss%=mode
    sums.append(ss)

for i in list1:
    print((sums[i[1]]-sums[i[0]-1]+mode)%mode)
```

代码运行截图 （至少包含有"Accepted"）


PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS ROOM STANDINGS CUSTOM INVOCATION
General
Copy

#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
293337945	Practice: chaain	474D - 15	PyPy 3-64	Accepted	359 ms	20908 KB	2024-11-26 13:00:01	2024-11-26 13:00:01	☆	Compare

→ Source

```
t,k = map(int,input().split())
list1=[]
ans=[]
maxv=0
mode=10**9+7
for i in range(t):
    a,b=map(int,input().split())
    list1.append([a,b])
    maxv=max(maxv,b)
dp = [0]*(maxv+1)
sums=[0]
ss=0
dp[0]=1
for i in range(1,maxv+1):
    if i<k:
        dp[i]=1
    else:
        dp[i]=(dp[i-1]+dp[i-k])%mode
    ss+=dp[i]
    ss%=mode
    sums.append(ss)

for i in list1:
    print((sums[i[1]]-sums[i[0]-1]+mode)%mode)
```

LeetCode5.最长回文子串

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

思路：难想，询问ai才有的思路

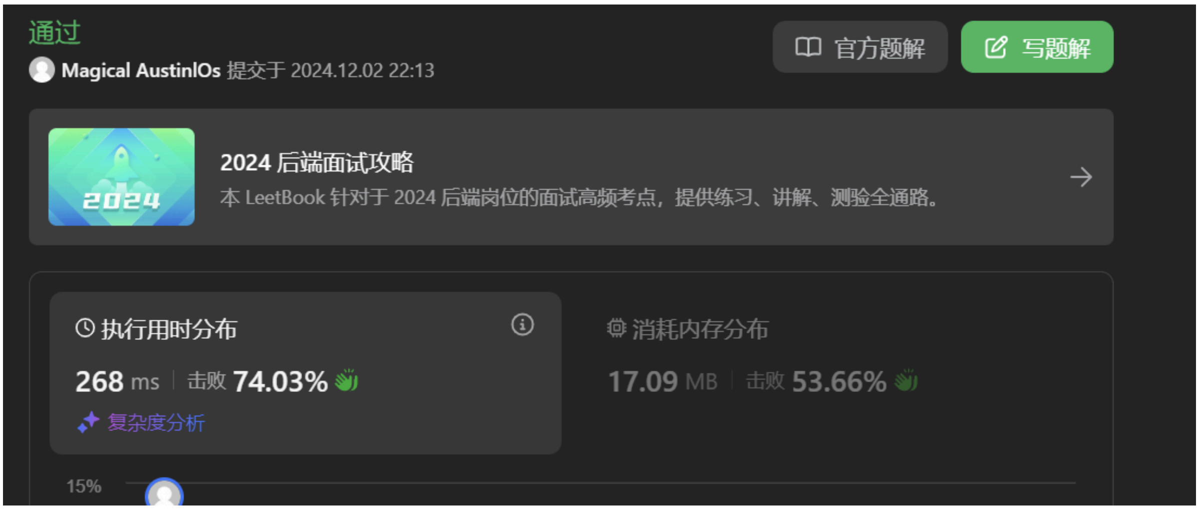
代码：

```
class Solution(object):
    def longestPalindrome(s:str)->str:
        """
        :type s: str
        :rtype: str
        """
        if not s:
            return ''
        n = len(s)
        start = 0
        mlength = 1

        def expansion(left, right):
            nonlocal start, mlength
            while left >= 0 and right < n and s[left] == s[right]:
                length = right - left + 1
                left -= 1
                right += 1
            if length >= mlength:
                mlength = length
                start = left + 1

        for i in range(n - 1):
            expansion(i, i)
            expansion(i, i + 1)
        expansion(n - 1, n - 1)
        return s[start:start + mlength]
```

代码运行截图 （至少包含有"Accepted"）



通过

Magical AustinIOs 提交于 2024.12.02 22:13

2024 后端面试攻略

本 LeetBook 针对于 2024 后端岗位的面试高频考点，提供练习、讲解、测验全通路。

执行用时分布

268 ms | 击败 74.03%

复杂度分析

消耗内存分布

17.09 MB | 击败 53.66%

15%

12029: 水淹七军

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

思路：太复杂了，其实题目本身倒没什么难度，代码难写

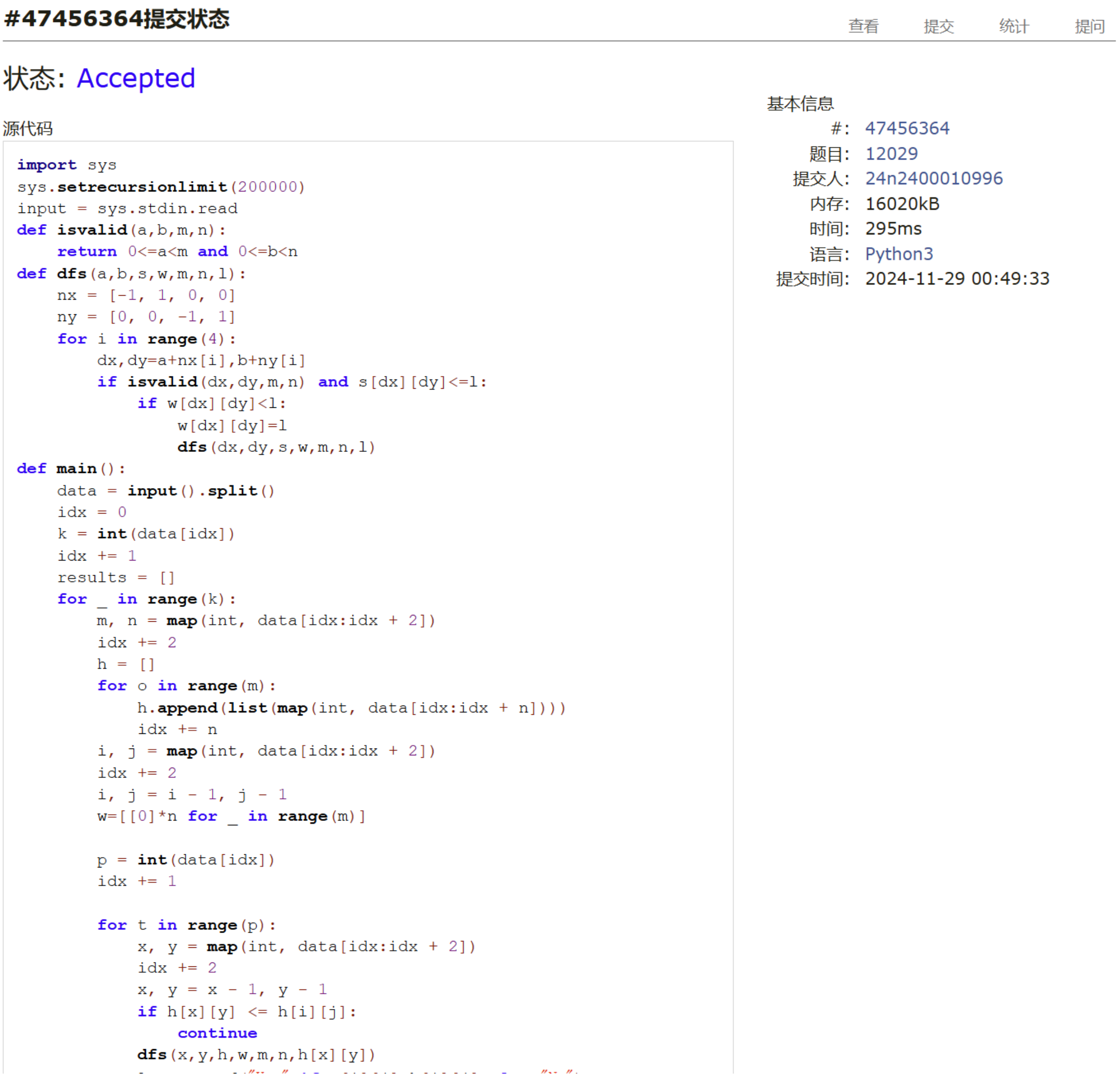
代码：

```python
import sys
sys.setrecursionlimit(200000)
input = sys.stdin.read
def isvalid(a,b,m,n):
    return 0<=a<m and 0<=b<n
def dfs(a,b,s,w,m,n,l):
    nx = [-1, 1, 0, 0]
    ny = [0, 0, -1, 1]
    for i in range(4):
        dx,dy=a+nx[i],b+ny[i]
        if isvalid(dx,dy,m,n) and s[dx][dy]<=l:
            if w[dx][dy]<l:
                w[dx][dy]=l
                dfs(dx,dy,s,w,m,n,l)
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
        h = []
        for o in range(m):
            h.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
        w=[[0]*n for _ in range(m)]

        p = int(data[idx])
        idx += 1

        for t in range(p):
            x, y = map(int, data[idx:idx + 2])
            idx += 2
            x, y = x - 1, y - 1
            if h[x][y] <= h[i][j]:
                continue
            dfs(x,y,h,w,m,n,h[x][y])
        results.append("Yes" if w[i][j]>h[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"）



#47456364提交状态

状态: Accepted

源代码

```
import sys
sys.setrecursionlimit(200000)
input = sys.stdin.read
def isvalid(a,b,m,n):
    return 0<=a<m and 0<=b<n
def dfs(a,b,s,w,m,n,l):
    nx = [-1, 1, 0, 0]
    ny = [0, 0, -1, 1]
    for i in range(4):
        dx,dy=a+nx[i],b+ny[i]
        if isvalid(dx,dy,m,n) and s[dx][dy]<=l:
            if w[dx][dy]<l:
                w[dx][dy]=l
                dfs(dx,dy,s,w,m,n,l)
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
        h = []
        for o in range(m):
            h.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
        w=[[0]*n for _ in range(m)]

        p = int(data[idx])
        idx += 1

        for t in range(p):
            x, y = map(int, data[idx:idx + 2])
            idx += 2
            x, y = x - 1, y - 1
            if h[x][y] <= h[i][j]:
                continue
            dfs(x,y,h,w,m,n,h[x][y])
```

基本信息

#:	47456364
题目:	12029
提交人:	24n2400010996
内存:	16020kB
时间:	295ms
语言:	Python3
提交时间:	2024-11-29 00:49:33

02802: 小游戏

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

思路：思路容易有，写对代码难

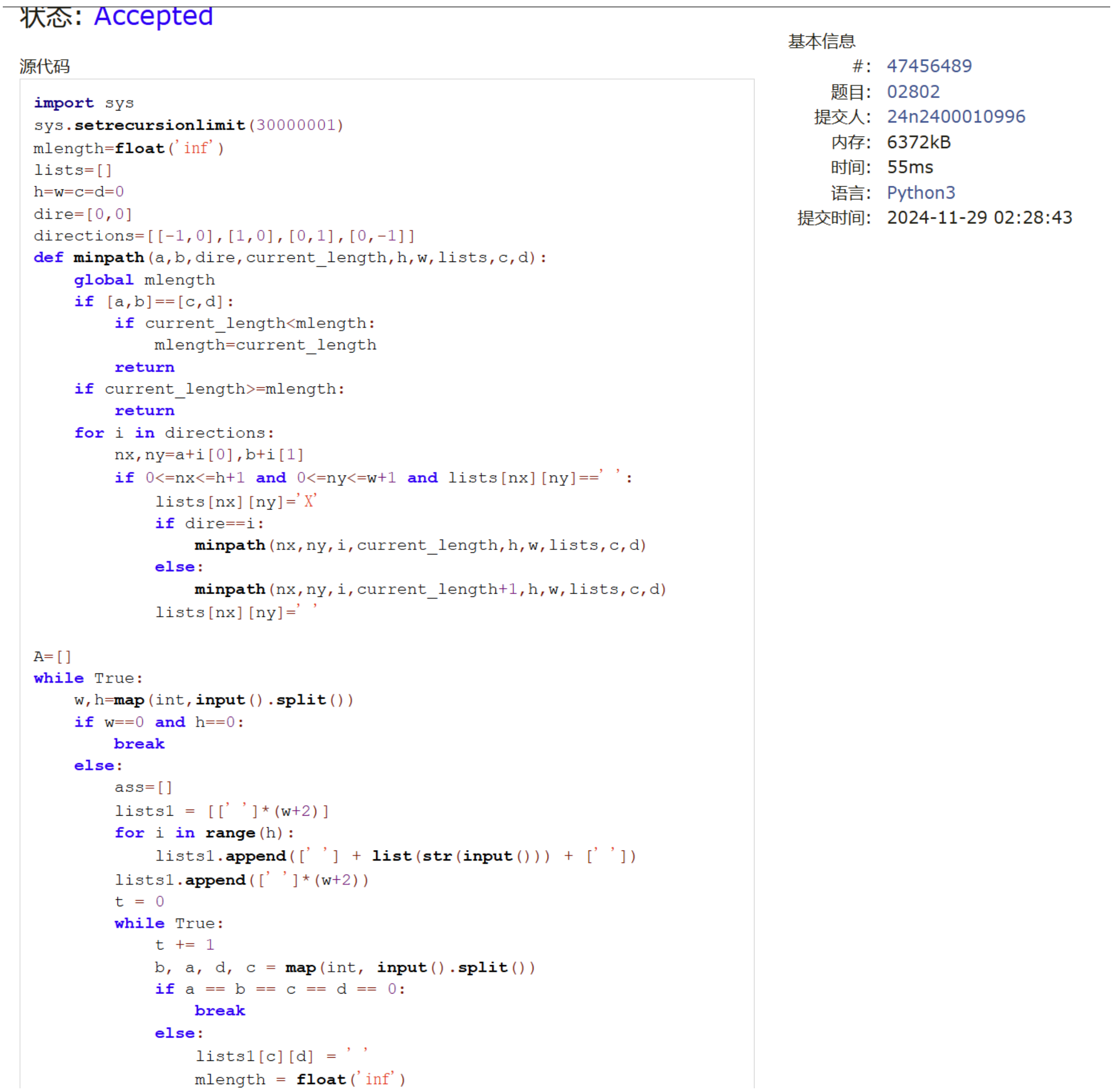
代码：

```python
import sys
sys.setrecursionlimit(30000001)
mlength=float('inf')
lists=[]
h=w=c=d=0
dire=[0,0]
directions=[[-1,0],[1,0],[0,1],[0,-1]]
def minpath(a,b,dire,current_length,h,w,lists,c,d):
    global mlength
    if [a,b]==[c,d]:
        if current_length<mlength:
            mlength=current_length
        return
    if current_length>=mlength:
        return
    for i in directions:
        nx,ny=a+i[0],b+i[1]
        if 0<=nx<=h+1 and 0<=ny<=w+1 and lists[nx][ny]==' ':
            lists[nx][ny]='X'
            if dire==i:
                minpath(nx,ny,i,current_length,h,w,lists,c,d)
            else:
                minpath(nx,ny,i,current_length+1,h,w,lists,c,d)
            lists[nx][ny]=' '

A=[]
while True:
    w,h=map(int,input().split())
    if w==0 and h==0:
        break
    else:
        ass=[]
        lists1 = [[' ']*(w+2)]
        for i in range(h):
            lists1.append([' '] + list(str(input())) + [' '])
        lists1.append([' ']*(w+2))
        t = 0
        while True:
            t += 1
            b, a, d, c = map(int, input().split())
            if a == b == c == d == 0:
                break
            else:
                lists1[c][d] = ' '
                mlength = float('inf')
                minpath(a, b,[0,0], 0,h,w,lists1,c,d)
                lists1[c][d] = 'X'
                if mlength == float('inf'):
                    ass.append(f'Pair {t}: impossible.')
                else:
                    ass.append(f'Pair {t}: {mlength} segments.')
    A.append(ass)

for i in range(len(A)):
    print(f'Board #{i + 1}:')
    for z in A[i]:
        print(z)
    print()
```

代码运行截图 （至少包含有"Accepted"）



状态: Accepted

源代码

```
import sys
sys.setrecursionlimit(30000001)
mlength=float('inf')
lists=[]
h=w=c=d=0
dire=[0,0]
directions=[[-1,0],[1,0],[0,1],[0,-1]]
def minpath(a,b,dire,current_length,h,w,lists,c,d):
    global mlength
    if [a,b]==[c,d]:
        if current_length<mlength:
            mlength=current_length
        return
    if current_length>=mlength:
        return
    for i in directions:
        nx,ny=a+i[0],b+i[1]
        if 0<=nx<=h+1 and 0<=ny<=w+1 and lists[nx][ny]==' ':
            lists[nx][ny]='X'
            if dire==i:
                minpath(nx,ny,i,current_length,h,w,lists,c,d)
            else:
                minpath(nx,ny,i,current_length+1,h,w,lists,c,d)
            lists[nx][ny]=' '

A=[]
while True:
    w,h=map(int,input().split())
    if w==0 and h==0:
        break
    else:
        ass=[]
        lists1 = [[' ']*(w+2)]
        for i in range(h):
            lists1.append([' '] + list(str(input())) + [' '])
        lists1.append([' ']*(w+2))
        t = 0
        while True:
            t += 1
            b, a, d, c = map(int, input().split())
            if a == b == c == d == 0:
                break
            else:
                lists1[c][d] = ' '
                mlength = float('inf')
```

基本信息

#:	47456489
题目:	02802
提交人:	24n2400010996
内存:	6372kB
时间:	55ms
语言:	Python3
提交时间:	2024-11-29 02:28:43

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。

其实这周作业除了回文是个新类型的题，其他都是之前见过的思路，但是这周对代码复杂性的要求一下子就上来了，发现其实题目很容易

就有思路，核心代码其实都不难，但把整个代码写好写对就很困难了，很容易就会出一些小错误，而且改错往往耗费大量时间，很让人抓狂，现在最担心的就是考试也出现这样的问题