Assignment #9: dfs, bfs, & dp

Updated 2107 GMT+8 Nov 19, 2024

2024 fall, Complied by 宋宇宸 元培学院

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

18160: 最大连通域面积

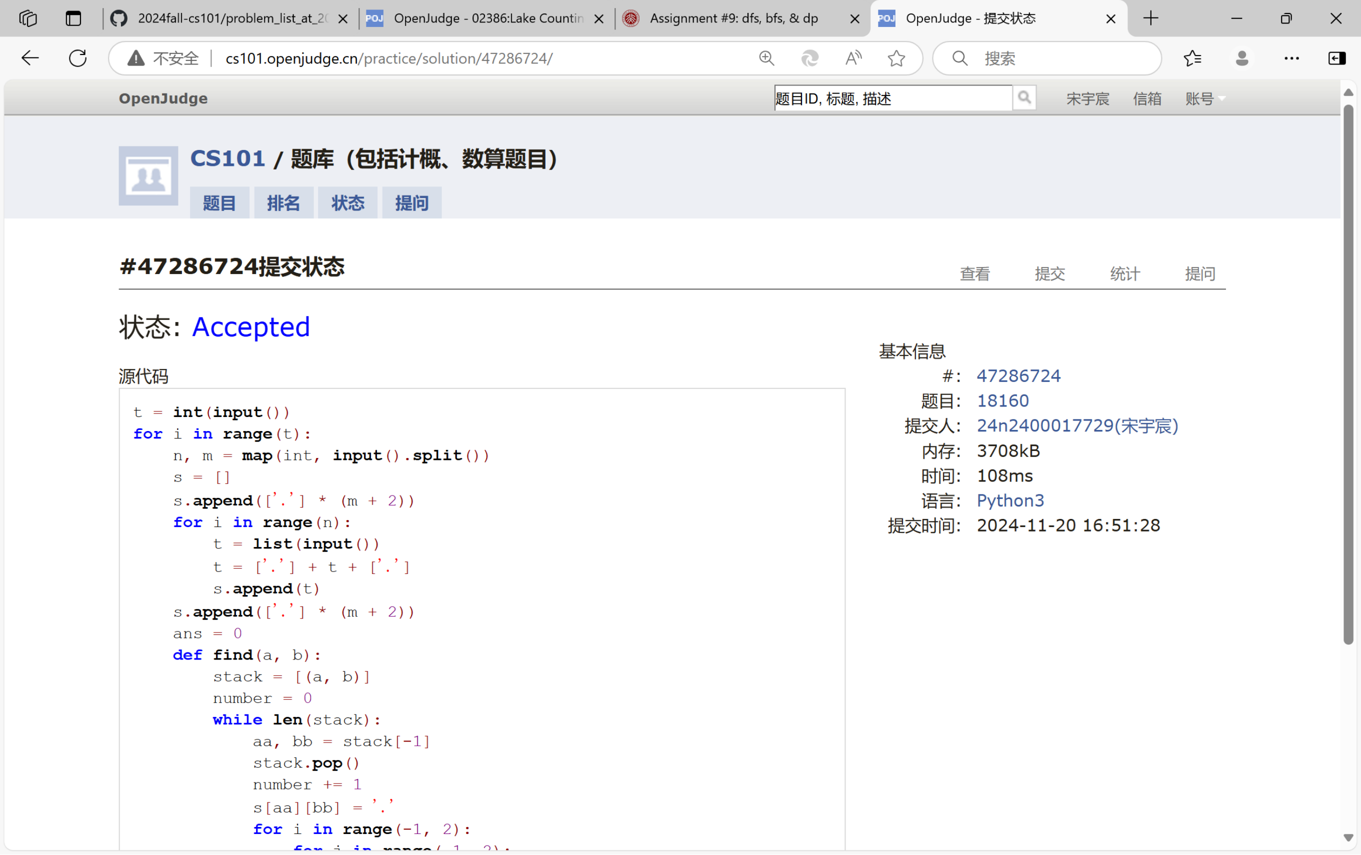
dfs similar, http://cs101.openjudge.cn/practice/18160

思路：

代码：

```
t = int(input())
for i in range(t):
    n, m = map(int, input().split())
    s = []
    s.append(['.'] * (m + 2))
    for i in range(n):
        t = list(input())
        t = ['.'] + t + ['.']
        s.append(t)
    s.append(['.'] * (m + 2))
    ans = 0
    def find(a, b):
        stack = [(a, b)]
        number = 0
        while len(stack):
            aa, bb = stack[-1]
            stack.pop()
            number += 1
            s[aa][bb] = '.'
            for i in range(-1, 2):
                for j in range(-1, 2):
                    if s[aa + i][bb + j] == "W":
                        stack.append((aa + i, bb + j))
                        s[aa + i][bb + j] = '.'
        return number
    for i in range(1, n + 1):
        for j in range(1, m + 1):
            if s[i][j] == "W":
                ans = max(ans, find(i, j))
    print(ans)
```

代码运行截图 （至少包含有"Accepted"）


2024fall-cs101/problem_list_at_2
OpenJudge - 02386:Lake Countin
Assignment #9: dfs, bfs, & dp
OpenJudge - 提交状态
不安全
cs101.openjudge.cn/practice/solution/47286724/
搜索
OpenJudge
题目ID, 标题, 描述
宋宇宸
信箱
账号
CS101 / 题库（包括计概、数算题目）
题目
排名
状态
提问
查看
提交
统计
提问

#47286724提交状态

状态: Accepted

源代码

```
t = int(input())
for i in range(t):
    n, m = map(int, input().split())
    s = []
    s.append(['.'] * (m + 2))
    for i in range(n):
        t = list(input())
        t = ['.'] + t + ['.']
        s.append(t)
    s.append(['.'] * (m + 2))
    ans = 0
    def find(a, b):
        stack = [(a, b)]
        number = 0
        while len(stack):
            aa, bb = stack[-1]
            stack.pop()
            number += 1
            s[aa][bb] = '.'
            for i in range(-1, 2):
                for j in range(-1, 2):
```

基本信息

#:	47286724
题目:	18160
提交人:	24n2400017729(宋宇宸)
内存:	3708kB
时间:	108ms
语言:	Python3
提交时间:	2024-11-20 16:51:28

19930: 寻宝

bfs, http://cs101.openjudge.cn/practice/19930

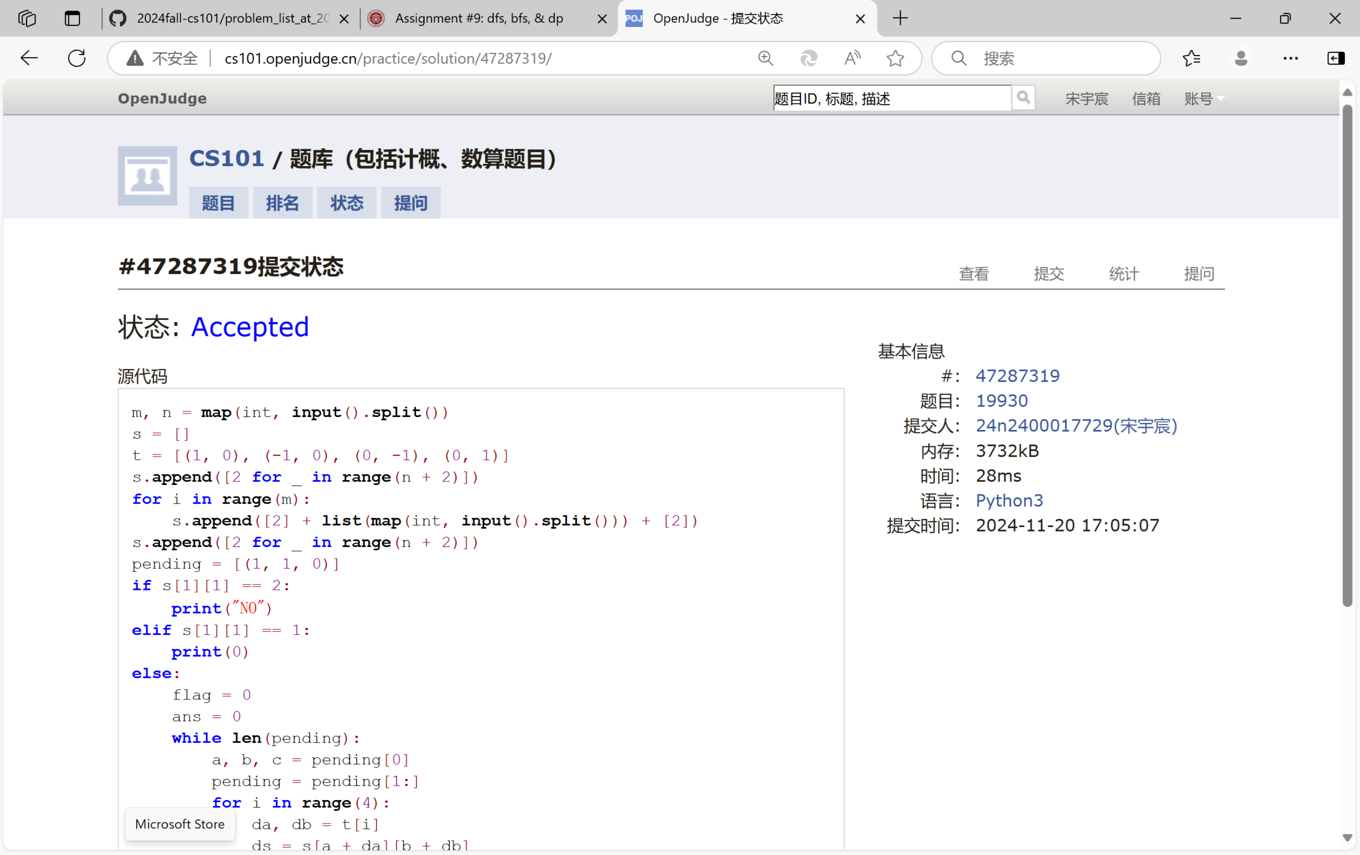
思路：

代码：

```
m, n = map(int, input().split())
s = []
t = [(1, 0), (-1, 0), (0, -1), (0, 1)]
s.append([2 for _ in range(n + 2)])
for i in range(m):
    s.append([2] + list(map(int, input().split())) + [2])
s.append([2 for _ in range(n + 2)])
pending = [(1, 1, 0)]
if s[1][1] == 2:
    print("NO")
elif s[1][1] == 1:
```

```python
    print(0)
else:
    flag = 0
    ans = 0
    while len(pending):
        a, b, c = pending[0]
        pending = pending[1:]
        for i in range(4):
            da, db = t[i]
            ds = s[a + da][b + db]
            if ds == 0:
                pending.append((a + da, b + db, c + 1))
                s[a + da][b + db] = 2
            elif ds == 1:
                flag = 1
                ans = c + 1
                break
        if flag == 1:
            break
    if flag == 1:
        print(ans)
    else:
        print("NO")
```

代码运行截图 ==（至少包含有"Accepted"）==


2024fall-cs101/problem_list_at_20
Assignment #9: dfs, bfs, & dp
OpenJudge - 提交状态
不安全
cs101.openjudge.cn/practice/solution/47287319/
搜索
OpenJudge
题目ID, 标题, 描述
宋宇宸
信箱
账号
CS101 / 题库（包括计概、数算题目）
题目
排名
状态
提问
查看
提交
统计
提问
Microsoft Store

#47287319提交状态

状态: Accepted

源代码

```python
m, n = map(int, input().split())
s = []
t = [(1, 0), (-1, 0), (0, -1), (0, 1)]
s.append([2 for _ in range(n + 2)])
for i in range(m):
    s.append([2] + list(map(int, input().split())) + [2])
s.append([2 for _ in range(n + 2)])
pending = [(1, 1, 0)]
if s[1][1] == 2:
    print("NO")
elif s[1][1] == 1:
    print(0)
else:
    flag = 0
    ans = 0
    while len(pending):
        a, b, c = pending[0]
        pending = pending[1:]
        for i in range(4):
            da, db = t[i]
            ds = s[a + da][b + db]
```

基本信息

#: 47287319
题目: 19930
提交人: 24n2400017729(宋宇宸)
内存: 3732kB
时间: 28ms
语言: Python3
提交时间: 2024-11-20 17:05:07

04123: 马走日

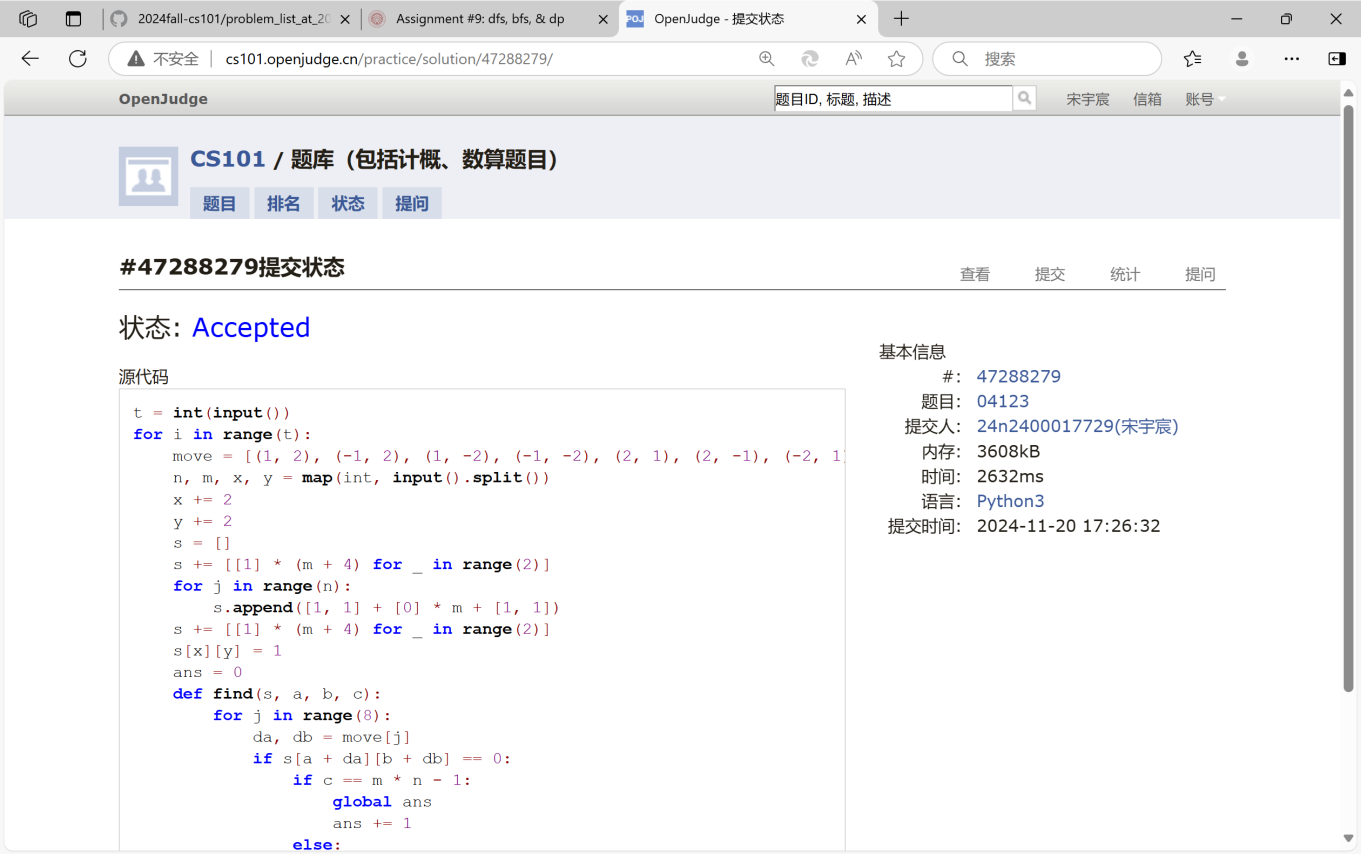
dfs, http://cs101.openjudge.cn/practice/04123

思路：

代码：

```
t = int(input())
for i in range(t):
    move = [(1, 2), (-1, 2), (1, -2), (-1, -2), (2, 1), (2, -1), (-2, 1), (-2, -1)]
    n, m, x, y = map(int, input().split())
    x += 2
    y += 2
    s = []
    s += [[1] * (m + 4) for _ in range(2)]
    for j in range(n):
        s.append([1, 1] + [0] * m + [1, 1])
    s += [[1] * (m + 4) for _ in range(2)]
    s[x][y] = 1
    ans = 0
    def find(s, a, b, c):
        for j in range(8):
            da, db = move[j]
            if s[a + da][b + db] == 0:
                if c == m * n - 1:
                    global ans
                    ans += 1
                else:
                    s[a + da][b + db] = 1
                    find(s.copy(), a + da, b + db, c + 1)
                    s[a + da][b + db] = 0
    find(s.copy(), x, y, 1)
    print(ans)
```

代码运行截图 （至少包含有"Accepted"）


2024fall-cs101/problem_list_at_20
Assignment #9: dfs, bfs, & dp
OpenJudge - 提交状态
不安全 | cs101.openjudge.cn/practice/solution/47288279/
搜索
OpenJudge
题目ID, 标题, 描述
宋宇宸
信箱
账号
CS101 / 题库（包括计概、数算题目）
题目
排名
状态
提问
查看
提交
统计
提问

#47288279提交状态

状态: Accepted

源代码

```
t = int(input())
for i in range(t):
    move = [(1, 2), (-1, 2), (1, -2), (-1, -2), (2, 1), (2, -1), (-2, 1
    n, m, x, y = map(int, input().split())
    x += 2
    y += 2
    s = []
    s += [[1] * (m + 4) for _ in range(2)]
    for j in range(n):
        s.append([1, 1] + [0] * m + [1, 1])
    s += [[1] * (m + 4) for _ in range(2)]
    s[x][y] = 1
    ans = 0
    def find(s, a, b, c):
        for j in range(8):
            da, db = move[j]
            if s[a + da][b + db] == 0:
                if c == m * n - 1:
                    global ans
                    ans += 1
                else:
```

基本信息
#: 47288279
题目: 04123
提交人: 24n2400017729(宋宇宸)
内存: 3608kB
时间: 2632ms
语言: Python3
提交时间: 2024-11-20 17:26:32

sy316: 矩阵最大权值路径

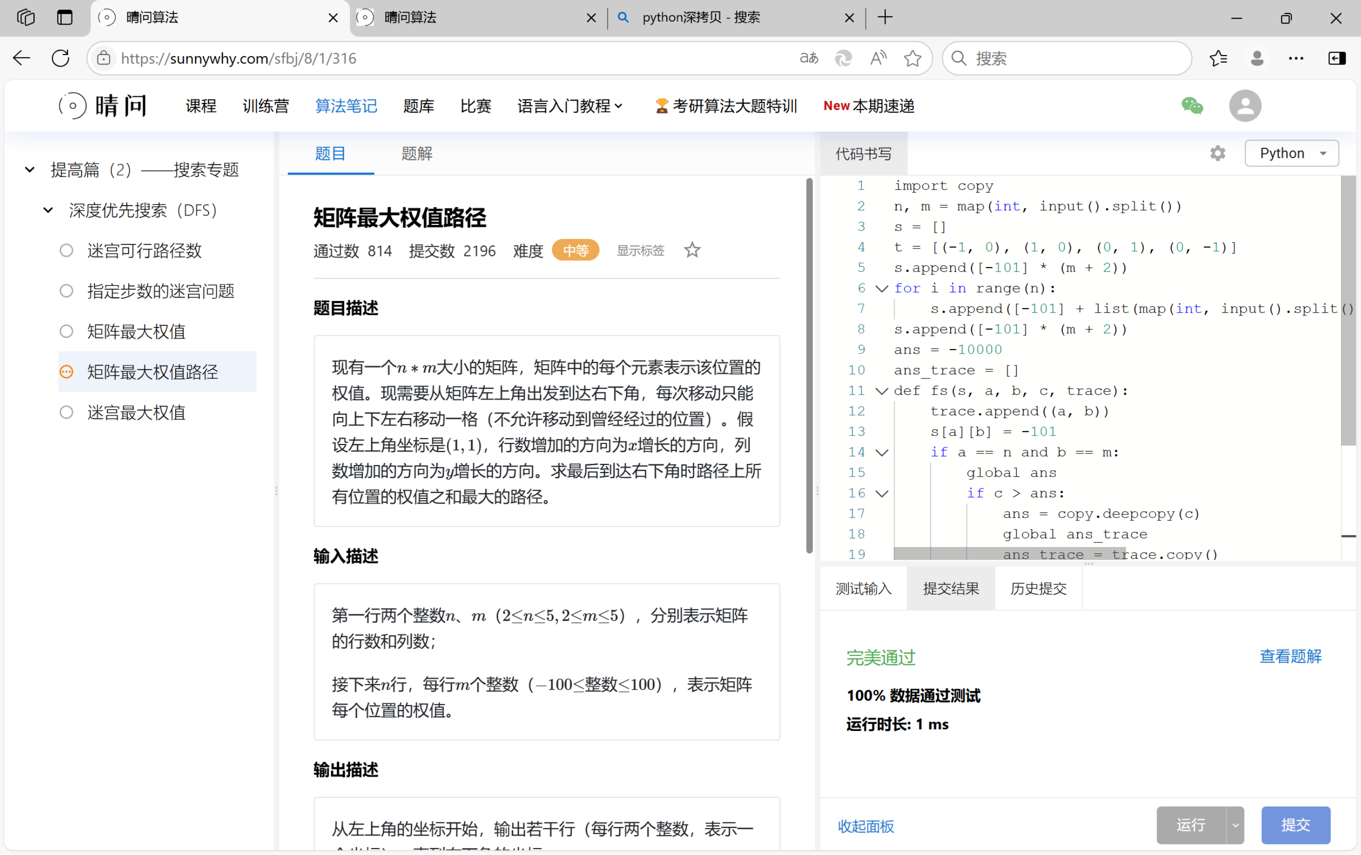
dfs, https://sunnywhy.com/sfbj/8/1/316

思路：

代码：

```python
import copy
n, m = map(int, input().split())
s = []
t = [(-1, 0), (1, 0), (0, 1), (0, -1)]
s.append([-101] * (m + 2))
for i in range(n):
    s.append([-101] + list(map(int, input().split())) + [-101])
s.append([-101] * (m + 2))
ans = -10000
ans_trace = []
def fs(s, a, b, c, trace):
    trace.append((a, b))
    s[a][b] = -101
    if a == n and b == m:
        global ans
        if c > ans:
            ans = copy.deepcopy(c)
            global ans_trace
            ans_trace = trace.copy()
    for i in range(4):
        da, db = t[i]
        if s[a + da][b + db] != -101:
            fs(copy.deepcopy(s), a + da, b + db, c + s[a + da][b + db], 
copy.deepcopy(trace))
fs(s.copy(), 1, 1, s[1][1], [])
for i, j in ans_trace:
    print(i, j)
```

代码运行截图 （至少包含有"Accepted"）



矩阵最大权值路径

通过数 814 提交数 2196 难度 中等 显示标签

题目描述

现有一个$n*m$大小的矩阵，矩阵中的每个元素表示该位置的权值。现需要从矩阵左上角出发到达右下角，每次移动只能向上下左右移动一格（不允许移动到曾经经过的位置）。假设左上角坐标是$(1,1)$，行数增加的方向为x增长的方向，列数增加的方向为y增长的方向。求最后到达右下角时路径上所有位置的权值之和最大的路径。

输入描述

第一行两个整数n、m（$2 \le n \le 5, 2 \le m \le 5$），分别表示矩阵的行数和列数；

接下来n行，每行m个整数（$-100 \le 整数 \le 100$），表示矩阵每个位置的权值。

输出描述

从左上角的坐标开始，输出若干行（每行两个整数，表示一

```
import copy
n, m = map(int, input().split())
s = []
t = [(-1, 0), (1, 0), (0, 1), (0, -1)]
s.append([-101] * (m + 2))
for i in range(n):
    s.append([-101] + list(map(int, input().split()
s.append([-101] * (m + 2))
ans = -10000
ans_trace = []
def fs(s, a, b, c, trace):
    trace.append((a, b))
    s[a][b] = -101
    if a == n and b == m:
        global ans
        if c > ans:
            ans = copy.deepcopy(c)
            global ans_trace
            ans_trace = trace.copy()
```

完美通过 查看题解

100% 数据通过测试

运行时长: 1 ms

LeetCode62.不同路径

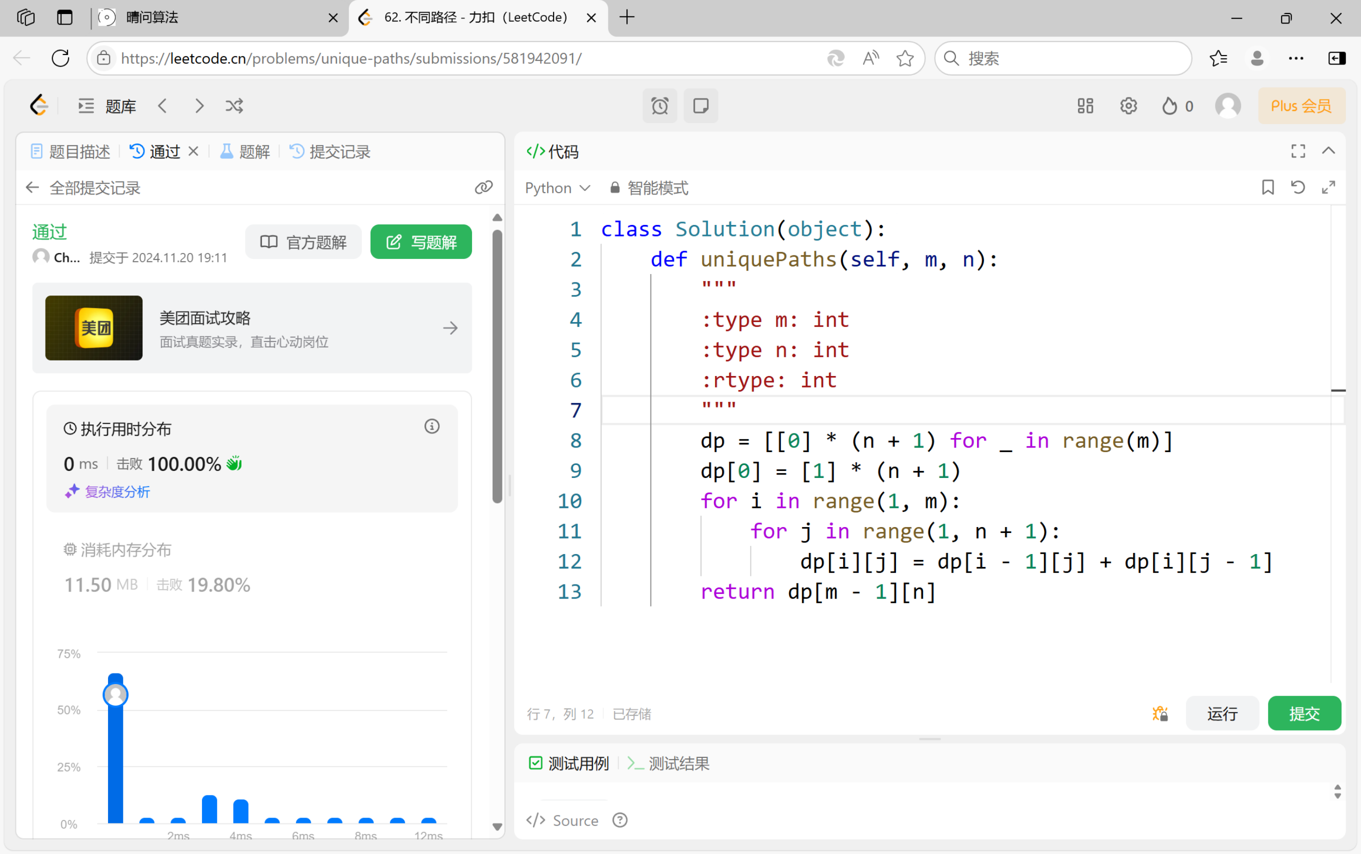
dp, https://leetcode.cn/problems/unique-paths/

思路：

代码：

```
class Solution(object):
    def uniquePaths(self, m, n):
        """
        :type m: int
        :type n: int
        :rtype: int
        """
        dp = [[0] * (n + 1) for _ in range(m)]
        dp[0] = [1] * (n + 1)
        for i in range(1, m):
            for j in range(1, n + 1):
                dp[i][j] = dp[i - 1][j] + dp[i][j - 1]
        return dp[m - 1][n]
```

代码运行截图 （至少包含有"Accepted"）



通过
Ch... 提交于 2024.11.20 19:11

执行用时分布
0 ms | 击败 100.00%

消耗内存分布
11.50 MB | 击败 19.80%

```python
class Solution(object):
    def uniquePaths(self, m, n):
        """
        :type m: int
        :type n: int
        :rtype: int
        """
        dp = [[0] * (n + 1) for _ in range(m)]
        dp[0] = [1] * (n + 1)
        for i in range(1, m):
            for j in range(1, n + 1):
                dp[i][j] = dp[i - 1][j] + dp[i][j - 1]
        return dp[m - 1][n]
```

sy358: 受到祝福的平方

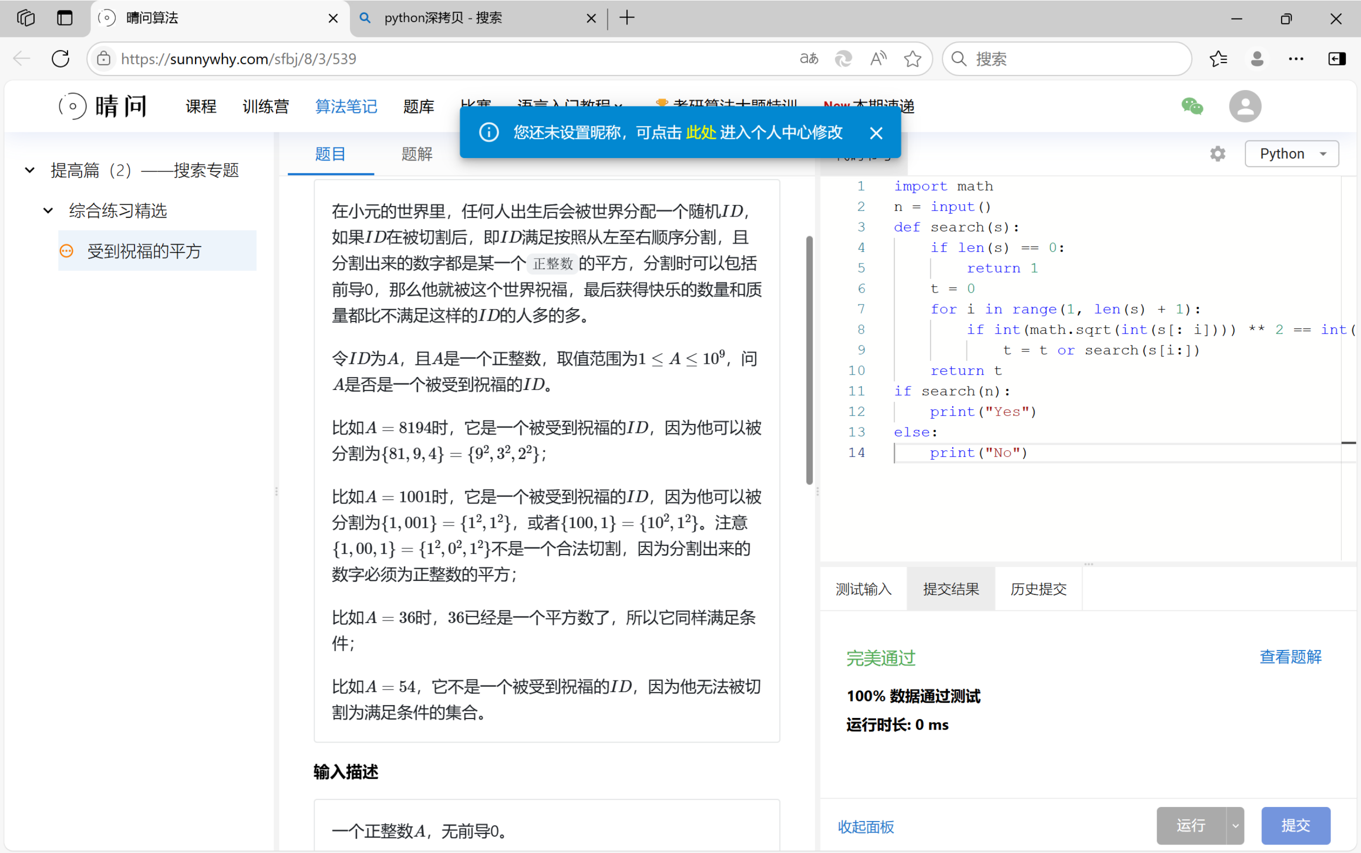
dfs, dp, https://sunnywhy.com/sfbj/8/3/539

思路：

代码：

```python
import math
n = input()
def search(s):
    if len(s) == 0:
        return 1
    t = 0
    for i in range(1, len(s) + 1):
        if int(math.sqrt(int(s[: i]))) ** 2 == int(s[: i]) and int(s[: i]) != 0:
            t = t or search(s[i:])
    return t
if search(n):
    print("Yes")
else:
    print("No")
```

代码运行截图 （至少包含有"Accepted"）



在小元的世界里，任何人出生后会被世界分配一个随机ID，如果ID在被切割后，即ID满足按照从左至右顺序分割，且分割出来的数字都是某一个 正整数 的平方，分割时可以包括前导0，那么他就被这个世界祝福，最后获得快乐的数量和质量都比不满足这样的ID的人多的多。

令ID为A，且A是一个正整数，取值范围为$1 \le A \le 10^9$，问A是否是一个被受到祝福的ID。

比如$A = 8194$时，它是一个被受到祝福的ID，因为他可以被分割为$\{81, 9, 4\} = \{9^2, 3^2, 2^2\}$；

比如$A = 1001$时，它是一个被受到祝福的ID，因为他可以被分割为$\{1, 001\} = \{1^2, 1^2\}$，或者$\{100, 1\} = \{10^2, 1^2\}$。注意$\{1, 00, 1\} = \{1^2, 0^2, 1^2\}$不是一个合法切割，因为分割出来的数字必须为正整数的平方；

比如$A = 36$时，36已经是一个平方数了，所以它同样满足条件；

比如$A = 54$，它不是一个被受到祝福的ID，因为他无法被切割为满足条件的集合。

输入描述

一个正整数A，无前导0。

```python
import math
n = input()
def search(s):
    if len(s) == 0:
        return 1
    t = 0
    for i in range(1, len(s) + 1):
        if int(math.sqrt(int(s[: i]))) ** 2 == int(
            t = t or search(s[i:])
    return t
if search(n):
    print("Yes")
else:
    print("No")
```

完美通过

100% 数据通过测试

运行时长: 0 ms

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。

感觉搜索还需要多多练习