Assignment #8: 田忌赛马来了

Updated 1021 GMT+8 Nov 12, 2024

2024 fall, Complied by 佟永鑫元培学院

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

12558: 岛屿周长

matices, http://cs101.openjudge.cn/practice/12558/

思路：

代码：

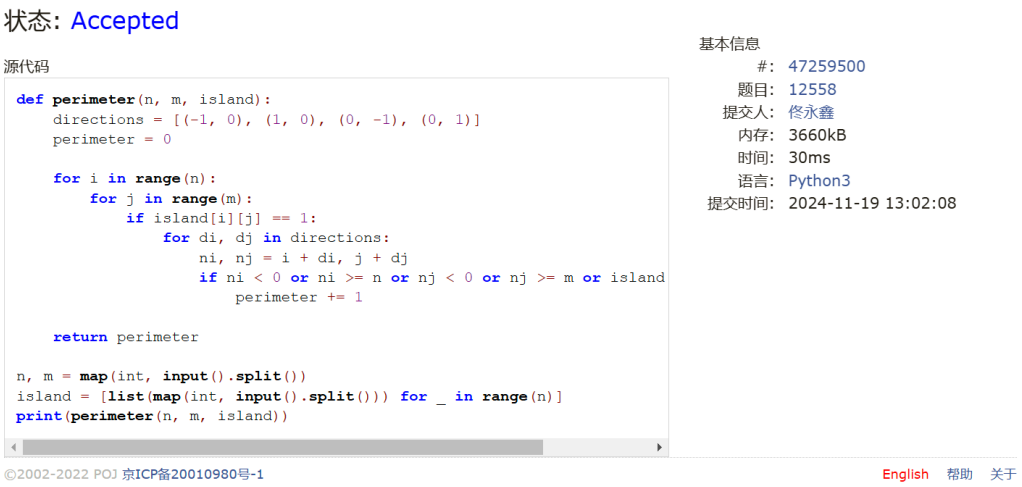
```
def perimeter(n, m, island):
    directions = [(-1, 0), (1, 0), (0, -1), (0, 1)]
    perimeter = 0

    for i in range(n):
        for j in range(m):
            if island[i][j] == 1:
                for di, dj in directions:
                    ni, nj = i + di, j + dj
                    if ni < 0 or ni >= n or nj < 0 or nj >= m or island[ni][nj]
== 0:
                        perimeter += 1

    return perimeter

n, m = map(int, input().split())
island = [list(map(int, input().split())) for _ in range(n)]
print(perimeter(n, m, island))
```

代码运行截图 （至少包含有"Accepted"）



LeetCode54.螺旋矩阵

matrice, https://leetcode.cn/problems/spiral-matrix/

与OJ这个题目一样的 18106: 螺旋矩阵，http://cs101.openjudge.cn/practice/18106

思路:

正常是用模拟写的，但看题解有递归写法，写了个相似的

代码:

```
class Solution:
    def spiralOrder(self, matrix: List[List[int]]) -> List[int]:
        def dfs(top, bottom, left, right):
            if top > bottom or left > right:
                return

            for j in range(left, right + 1):
                res.append(matrix[top][j])
            for i in range(top + 1, bottom + 1):
                res.append(matrix[i][right])
            if top < bottom:
                for j in range(right - 1, left - 1, -1):
                    res.append(matrix[bottom][j])
            if left < right:
                for i in range(bottom - 1, top, -1):
                    res.append(matrix[i][left])
            dfs(top + 1, bottom - 1, left + 1, right - 1)

        if not matrix or not matrix[0]:
            return []

        res = []
        dfs(0, len(matrix) - 1, 0, len(matrix[0]) - 1)
        return res
```

代码运行截图 ==（至少包含有"Accepted"）==

04133:垃圾炸弹

matrices, http://cs101.openjudge.cn/practice/04133/

思路：

遍历每个垃圾点，每个为中心2d+1的正方形赋值垃圾数，最后找到赋值相加的最大值

代码：

```
def garbage_bomb(d, n, locations):
    Map_size = 1025
    Map = [[0] * Map_size for _ in range(Map_size)]

    for x, y, garbage in locations:
        for i in range(max(0, x - d), min(Map_size, x + d + 1)):
            for j in range(max(0, y - d), min(Map_size, y + d + 1)):
                Map[i][j] += garbage
    max_garbage = 0
```

```
    max_count = 0
    max_garbage = max(max(row) for row in Map)

    max_count = sum(row.count(max_garbage) for row in Map)
    return max_count, max_garbage
d = int(input())
n = int(input())
locations = [tuple(map(int, input().split())) for _ in range(n)]

count, garbage = garbage_bomb(d, n, locations)
print(count, garbage)
```

代码运行截图 （至少包含有"Accepted"）

状态: Accepted

源代码

```
def garbage_bomb(d, n, locations):
    Map_size = 1025
    Map = [[0] * Map_size for _ in range(Map_size)]

    for x, y, garbage in locations:
        for i in range(max(0, x - d), min(Map_size, x + d + 1)):
            for j in range(max(0, y - d), min(Map_size, y + d + 1)):
                Map[i][j] += garbage
    max_garbage = 0
    max_count = 0
    max_garbage = max(max(row) for row in Map)

    max_count = sum(row.count(max_garbage) for row in Map)
    return max_count, max_garbage
d = int(input())
n = int(input())
locations = [tuple(map(int, input().split())) for _ in range(n)]

count, garbage = garbage_bomb(d, n, locations)
print(count, garbage)
```

基本信息

#: 47260196
题目: 04133
提交人: 佟永鑫
内存: 11876kB
时间: 59ms
语言: Python3
提交时间: 2024-11-19 13:41:18

京ICP备20010980号-1

English 帮助 关于

LeetCode376.摆动序列

greedy, dp, https://leetcode.cn/problems/wiggle-subsequence/

与OJ这个题目一样的，26976:摆动序列, http://cs101.openjudge.cn/routine/26976/

思路:

找到极值点

代码:

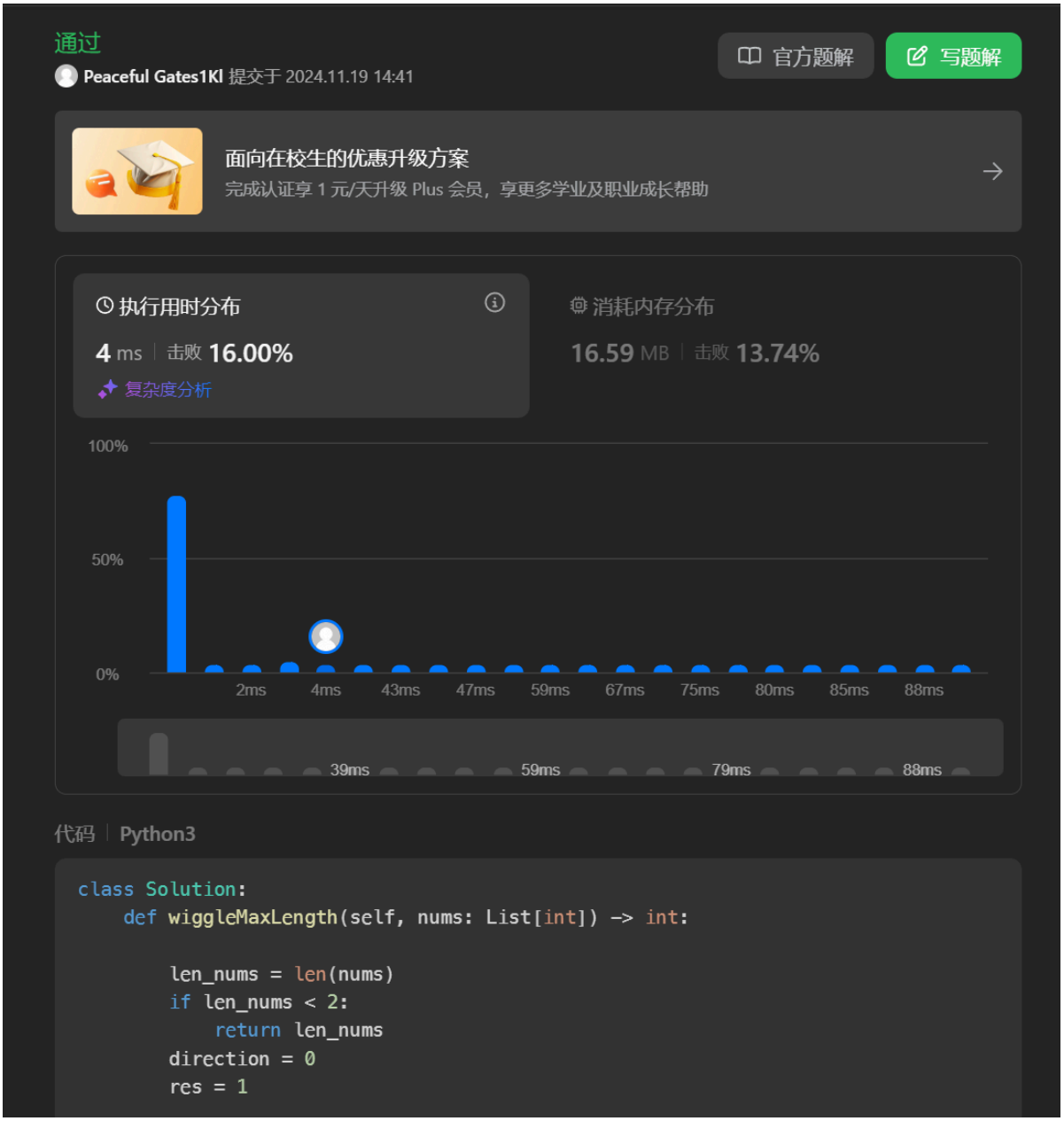
```
class Solution:
    def wiggleMaxLength(self, nums: List[int]) -> int:

        len_nums = len(nums)
        if len_nums < 2:
            return len_nums
        direction = 0
        res = 1
        for i in range(1, len_nums):
```

```
            diff = nums[i] - nums[i - 1]
            if diff > 0 and direction != 1:
                direction = 1
                res += 1
            elif diff < 0 and direction != -1:
                direction = -1
                res += 1

        return res
```

代码运行截图 （至少包含有"Accepted"）



CF455A: Boredom

dp, 1500, https://codeforces.com/contest/455/problem/A

思路：

经典DP

代码：

```
def Boredom(n, arr):
    max_val = max(arr)
    count = [0] * (max_val + 1)
    for num in arr:
        count[num] += 1
    dp = [0] * (max_val + 1)
    dp[1] = count[1]
    for i in range(2, max_val + 1):
        dp[i] = max(dp[i - 1], dp[i - 2] + i * count[i])
    return dp[max_val]
n = int(input())
arr = list(map(int, input().split()))
print(Boredom(n, arr))
```

代码运行截图 （至少包含有"Accepted"）

292280053	Nov/19/2024 14:04 UTC+8	tongyongxin	A - Boredom	Python 3	Accepted	155 ms	13100 KB

02287: Tian Ji -- The Horse Racing

greedy, dfs http://cs101.openjudge.cn/practice/02287

思路：

这题开始的思路是两方的马倒序排序，田忌的马去匹配齐王马里更慢但最接近的，避免浪费，但这么写出来发现题目给的还有群里讨论的样例都没问题，就是不AC。最后去翻老师给的测试文件，发现

10

3 5 7 9 2 4 6 8 10 11
11 9 8 7 6 5 4 3 2 1

这组有问题，按照我的思路11找9,10找8一直赢到最后2输11，但其实11和11打平后其他每组都能赢。然后参考题解的思路做了修改。

代码：

```
def tianji_racing():
    while True:
        n = int(input().strip())
        if n == 0:
            break
        tian_horses = list(map(int, input().split()))
        king_horses = list(map(int, input().split()))
```

```
        tian_horses.sort(reverse=True)
        king_horses.sort(reverse=True)
        answer = 0
        for _ in range(n):
            if tian_horses[0] > king_horses[0]:  # Compare strongest horses
                answer += 1
                tian_horses.pop(0)
                king_horses.pop(0)
            else:
                if tian_horses[-1] > king_horses[-1]:
                    answer += 1
                    tian_horses.pop(-1)
                    king_horses.pop(-1)
                else:
                    if tian_horses[-1] < king_horses[0]:
                        answer -= 1
                    tian_horses.pop(-1)
                    king_horses.pop(0)
        print(200 * answer)

tianji_racing()
```

代码运行截图 （至少包含有"Accepted"）

状态: Accepted

源代码

```
def tianji_racing():
    while True:
        n = int(input().strip())
        if n == 0:
            break
        tian_horses = list(map(int, input().split()))
        king_horses = list(map(int, input().split()))
        tian_horses.sort(reverse=True)
        king_horses.sort(reverse=True)
        answer = 0
        for _ in range(n):
            if tian_horses[0] > king_horses[0]:  # Compare strongest ho
                answer += 1
                tian_horses.pop(0)
                king_horses.pop(0)
            else:
                if tian_horses[-1] > king_horses[-1]:
                    answer += 1
                    tian_horses.pop(-1)
                    king_horses.pop(-1)
                else:
                    if tian_horses[-1] < king_horses[0]:
                        answer -= 1
                    tian_horses.pop(-1)
                    king_horses.pop(0)
        print(200 * answer)

tianji_racing()
```

基本信息

#:	47266797
题目:	02287
提交人:	佟永鑫
内存:	4072kB
时间:	57ms
语言:	Python3
提交时间:	2024-11-19 17:16:00

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。

前5题没什么问题，田忌赛马参考了题解，没有独立做出来很是遗憾