Assignment #8: 田忌赛马来了

Updated 1021 GMT+8 Nov 12, 2024

2024 fall, Complied by 同学的姓名、院系

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：

如果旁边同为岛屿模块，其接壤边就无法算作周长，所以4-四周岛屿数就可以

代码：

```
m, n = map(int, input().split())
spots = [[0]*(n + 2)]
C = 0
for i in range(m):
    spots.append([0] + list(map(int, input().split())) + [0])
spots.append([0]*(n + 2))
for i in range(1, m + 1):
    for j in range(1, n + 1):
        if spots[i][j] == 1:
            C += 4 - spots[i][j - 1] - spots[i][j + 1] - spots[i - 1][j] -
spots[i + 1][j]
print(C)
```

代码运行截图 （至少包含有"Accepted"）

状态: Accepted

源代码

```
m, n = map(int, input().split())
spots = [[0]*(n + 2)]
C = 0
for i in range(m):
    spots.append([0] + list(map(int, input().split())) + [0])
spots.append([0]*(n + 2))
for i in range(1, m + 1):
    for j in range(1, n + 1):
        if spots[i][j] == 1:
            C += 4 - spots[i][j - 1] - spots[i][j + 1] - spots[i - 1][
print(C)
```

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:

从1或2开始递归处理，对n - 2的结果添加边界即可

代码:

```
def generate_blocks(n):
    if n == 1:
        return [[1]]
    elif n == 2:
        return [[1, 2], [4, 3]]
    block = [[0] * n for _ in range(n)]

    block[0] = list(range(1, n + 1))
    block[n - 1] = list(range(3 * n - 2, 2 * n - 2, -1))
    for i in range(1, n - 1):
        block[i][0] = 4 * n - 3 - i
        block[i][n - 1] = n + i
    if n > 2:
        inner_block = generate_blocks(n - 2)
        offset = 4 * (n - 1)
        for i in range(n - 2):
            for j in range(n - 2):
                block[i + 1][j + 1] = inner_block[i][j] + offset
    return block

n = int(input())
block = generate_blocks(n)

for row in block:
    print(' '.join(map(str, row)))
```

代码运行截图 ==（至少包含有"Accepted"）==

状态：Accepted

基本

源代码

提交

```
def generate_blocks(n):
    if n == 1:
        return [[1]]
    elif n == 2:
        return [[1, 2], [4, 3]]
    block = [[0] * n for _ in range(n)]

    block[0] = list(range(1, n + 1))
    block[n - 1] = list(range(3 * n - 2, 2 * n - 2, -1))
    for i in range(1, n - 1):
        block[i][0] = 4 * n - 3 - i
        block[i][n - 1] = n + i
    if n > 2:
        inner_block = generate_blocks(n - 2)
        offset = 4 * (n - 1)
        for i in range(n - 2):
            for j in range(n - 2):
                block[i + 1][j + 1] = inner_block[i][j] + offset
    return block

n = int(input())
block = generate_blocks(n)

for row in block:
    print(' '.join(map(str, row)))
```

04133:垃圾炸弹

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

思路：

不断滚动投放地点，存储最优值

代码：

```
d = int(input())
n = int(input())
screen = [[0]*1025 for _ in range(1025)]
for i in range(n):
    x, y, v = map(int, input().split())
    for j in range(max(x - d, 0), min(x + d, 1024) + 1):
        for k in range(max(y - d, 0), min(y + d, 1024) + 1):
            screen[j][k] += v
res_max = 0
count = 0
for l in range(1025):
    for m in range(1025):
        if screen[l][m] > res_max:
            res_max = screen[l][m]
            count = 1
        elif screen[l][m] == res_max:
```

```
            count += 1
print('%d %d' %(count, res_max))
```

代码运行截图 （至少包含有"Accepted"）

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：

一个一个比较过去，摆动了就计数，不摆就跳过（记得处理==0情况）

代码：

```
def judge(x):
    if x == 0:
        return 0
    else:
        return x//abs(x)

n = int(input())
nums = list(map(int,input().split()))
change = [judge(nums[i+1]-nums[i]) for i in range(n-1)]

result = 1
pre = 0
for i in range(n-1):
    if (pre == 0 and change[i] != 0) or change[i]*pre < 0:
        result += 1
        pre = change[i]
print(result)
```

代码运行截图 （至少包含有"Accepted"）

状态：Accepted

源代码

```
def judge(x):
    if x == 0:
        return 0
    else:
        return(x//abs(x))


n = int(input())
nums = list(map(int,input().split()))
change = [judge(nums[i+1]-nums[i]) for i in range(n-1)]

result = 1
pre = 0
for i in range(n-1):
    if (pre == 0 and change[i] != 0) or change[i]*pre < 0:
        result += 1
        pre = change[i]
print(result)
```

CF455A: Boredom

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

思路：

因为存储卡在0的五次方比较好操作，先把相同的数归纳在一起，再从0开始观察一个数能不能删

代码：

```
n = int(input())
numbers = list(map(int, input().split()))
num = [0]*(10**5 + 1)
dp = [0]*(10**5 + 1)
for i in range(n):
    num[numbers[i]] += numbers[i]
for i in range(1, 10**5 + 1):
    dp[i] = max(dp[i - 1], dp[i - 2] + num[i])
print(max(dp))
```

代码运行截图 （至少包含有"Accepted"）

#	When	Who	Problem	Lang	Verdict	Time	Memory
291168426	Nov/12/2024 18:03UTC+8	lyralee	A - Boredom	Python 3	Accepted	171 ms	10700 KB

02287: Tian Ji -- The Horse Racing

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

思路：

找到恰好能赢的马，如果找不到就拿他对决对方最强的

代码：

```
while True:
    n = int(input())
    if n==0:
        break

    Tian = sorted(list(map(int, input().split())))
    King = sorted(list(map(int, input().split())))

    lTian = 0; rTian = n - 1
    lKing = 0; rKing = n - 1
    ans = 0
    while lTian <= rTian:
        if Tian[lTian] > King[lKing]:
            ans += 1
            lTian += 1
            lKing += 1
        elif Tian[rTian] > King[rKing]:
            ans += 1
            rTian -= 1
            rKing -= 1
        else:
            if Tian[lTian] < King[rKing]:
                ans -= 1

            lTian += 1
            rKing -= 1

    print(ans*200)
```

代码运行截图 （至少包含有"Accepted"）

#47128684提交状态

状态：Accepted

源代码

```
while True:
    n = int(input())
    if n==0:
        break
```

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。

本次作业除了田忌赛马理清思路用了很久都还算可以，且发现手写思维导图还是非常有用的

继续努力赶每日一题进度ing