Assignment #8: 田忌赛马来了

Updated 1021 GMT+8 Nov 12, 2024

2024 fall, Complied by <mark>任宇桐 物理学院</mark>

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：

出现岛屿就进行计数，但是要注意减去相邻的边。

代码：

```python
n, m = map(int, input().split())
island = [[0]*(m+2)]
for _ in range(n):
    s = [int(x) for x in input().split()]
    s.reverse()
    s.append(0)
    s.reverse()
    s.append(0)
    island.append(s)
island.append([0]*(m+2))
circle = 0
for i in range(1,n+1):
    for j in range(1,m+1):
        if island[i][j] == 1:
            circle += 4
            if island[i+1][j] ==1:
                circle -= 1
            if island[i-1][j] == 1:
                circle -= 1
            if island[i][j-1] == 1:
                circle -= 1
            if island[i][j+1] == 1:
                circle -= 1
```

```
print(circle)
```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

源代码
```
n, m = map(int, input().split())
island = [[0]*(m+2)]
for _ in range(n):
    s = [int(x) for x in input().split()]
    s.reverse()
    s.append(0)
    s.reverse()
    s.append(0)
    island.append(s)
island.append([0]*(m+2))
circle = 0
for i in range(1,n+1):
    for j in range(1,m+1):
        if island[i][j] == 1:
            circle += 4
            if island[i+1][j] ==1:
                circle -= 1
            if island[i-1][j] == 1:
                circle -= 1
            if island[i][j-1] == 1:
                circle -= 1
            if island[i][j+1] == 1:
                circle -= 1
print(circle)
```

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:

先沿着原来的方向移动，如果不行，就按照先向右，再向下，再向左，再向上的方式逐个尝试。

代码:

```
n = int(input())
matrix = [[0]*n for _ in range(n)]
temp = 0
i, j = 0, -1
step = [(0, 1), (1, 0), (0, -1), (-1, 0)]

flag = 0

while True:
    temp += 1
    ref = [j + 1 <= n - 1 and matrix[i][j + 1] == 0, i + 1 <= n - 1 and matrix[i
+ 1][j] == 0,
           j - 1 >= 0 and matrix[i][j - 1] == 0, i - 1 >= 0 and matrix[i - 1][j]
== 0]

    if ref[flag]:
        di, dj = step[flag]
        i = i+di
        j = j+dj
        matrix[i][j] = temp
```

```
    else:
        for k in range(4):
            if ref[k]:
                di, dj = step[k]
                i = i + di
                j = j + dj
                matrix[i][j] = temp
                flag = k

    if temp == n**2:
        break

for i in range(n):
    print(*matrix[i])
```

代码运行截图 <mark>(至少包含有"Accepted")</mark>

04133:垃圾炸弹

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

思路:

开始时候准备直接暴力枚举,然后发现很快会超时,后来发现可以手动降低一下数据的范围,然后就成功了,同时还需要注意枚举不能超出索引的范围。

代码:

```
d = int(input())
n = int(input())
s = []
for _ in range(n):
    s.append(list(map(int, input().split())))
x = 0
y = 0
maxv = 0
num = 1
minx = min(s[i][0] for i in range(n))
```

```python
    maxx = max(s[i][0] for i in range(n))
    miny = min(s[i][1] for i in range(n))
    maxy = max(s[i][1] for i in range(n))
    for x in range(max(0,minx-d), min(1025,maxx+1+d)): ### 注意加范围限制
        for y in range(max(0,miny-d), min(maxy+1+d,1025)):
            cnt = 0
            for i in range(n):
                if abs(x-s[i][0])<= d and abs(y-s[i][1]) <= d:
                    cnt += s[i][2]
            if cnt > maxv:
                maxv = cnt
                num = 1
            elif cnt == maxv:
                num += 1
            else:
                continue
print(num, maxv)
```

代码运行截图 <mark>(至少包含有"Accepted")</mark>

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：

如果序列两项差等于前两项的差，则将中间的数据去掉；如果相反，则可以直接加入。注意讨论常数列的情形。

代码：

```python
n = int(input())
s = list(map(int, input().split()))
if n == 1:
```

```
    print(1)
    exit(0)
cnt = 1
temp = s[1] - s[0]
pre = s[1]
for i in range(2, n):
    if temp*(s[i]-pre) < 0:
        temp = s[i] - pre
        pre = s[i]
        cnt += 1
    else:
        pre = s[i]


cnt += 1


if temp == 0:
    cnt -= 1
print(cnt)
```

代码运行截图 <mark>(至少包含有"Accepted")</mark>

CF455A: Boredom

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

思路:

注意到如果将值相同的数合并在一起可以很好的简化程序。

代码:

```
from collections import Counter

int(input())
s = list(map(int, input().split()))
count = Counter(s)
num = list(count.keys())
num.sort()
n = len(num)
dp = [[0, 0] for _ in range(n)]
```

```
dp[0] = [0, num[0]*count[num[0]]]
for i in range(1, n):
    if num[i] - num[i-1] > 1:
        dp[i][0] = max(dp[i-1])
        dp[i][1] = max(dp[i-1])+count[num[i]]*num[i]
    else:
        dp[i][0] = max(dp[i-1])
        dp[i][1] = dp[i-1][0]+count[num[i]]*num[i]

print(max(dp[-1]))
```

代码运行截图 <mark>(至少包含有"Accepted")</mark>

```
By stur, contest: Codeforces Round 260 (Div. 1), problem: (A) Boredom, Accepted, #, Copy

from collections import Counter

int(input())
s = list(map(int, input().split()))
count = Counter(s)
num = list(count.keys())
num.sort()
n = len(num)
dp = [[0, 0] for _ in range(n)]
dp[0] = [0, num[0]*count[num[0]]]
for i in range(1, n):
    if num[i] - num[i-1] > 1:
        dp[i][0] = max(dp[i-1])
        dp[i][1] = max(dp[i-1])+count[num[i]]*num[i]
    else:
        dp[i][0] = max(dp[i-1])
        dp[i][1] = dp[i-1][0]+count[num[i]]*num[i]

print(max(dp[-1]))

→Judgement Protocol

Test: #1, time: 77 ms., memory: 0 KB, exit code: 0, checker exit code: 0, verdict: OK
Input
```

02287: Tian Ji -- The Horse Racing

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

思路:

一开始没有觉得多难,后来反复提交都是WA,通过群聊中的记录才知道原来是因为没有考虑清楚相等的情况,最后也没能独立搞定,还是参考了题解才勉强过关。

代码:

```
while True:
    n = int(input())
    if n == 0:
        break
    t = list(map(int, input().split()))
    k = list(map(int, input().split()))

    t.sort()
    k.sort()
    cnt = 0
    i = 0
    j = 0
    m = n-1
    l = n-1

    while True:
```

```
        if i > m or j > l:
            break
        if t[i] > k[j]:
            i += 1
            j += 1
            cnt += 1
        elif t[m] > k[l]:
            m -= 1
            l -= 1
            cnt += 1
        else:
            if t[i] < k[l]:
                cnt -= 1

            i += 1
            l -= 1

    print(cnt*200)
```

代码运行截图 <mark>(至少包含有"Accepted")</mark>

2. 学习总结和收获

<mark>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。</mark>

田忌赛马断断续续做了很久，半天也没有搞定，看来确实还是有一定难度的。

最近一周期中考试有点多，没有及时跟进每日选做，接下来要尽量赶上。同时发现自己还有一些模板题目没有掌握，如最大子序列和等等，要尽快掌握了。