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:

```
n,m=map(int,input().split())
wait_list=[False]*m
wai=False
C=0
for _ in range(n):
    l=list(map(int,input().split()))
    for i in range(len(1)):
        if l[i]==1:
            if wait_list[i]:
                 if wai:
                     C=C+0
                 else:
                    c=c+2
            else:
                 if wai:
                    c=c+2
                 else:
                     C=C+4
            wait_list[i]=True
            wai=True
        else:
            wait_list[i]=False
            wai=False
```

```
wai=False
print(c)
```

代码运行截图 (至少包含有"Accepted")

#47164444提交状态

状态: Accepted

源代码

```
n,m=map(int,input().split())
wait_list=[False]*m
wai=False
c=0
for _ in range(n):
    l=list(map(int,input().split()))
    for i in range(len(l)):
```

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

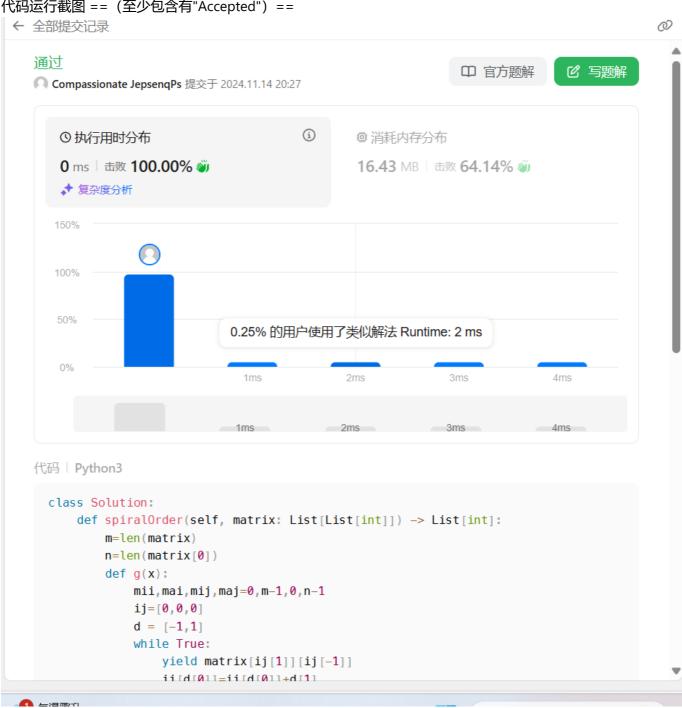
思路:

```
class Solution:
    def spiralOrder(self, matrix: List[List[int]]) -> List[int]:
        m=len(matrix)
        n=len(matrix[0])
        def g(x):
            mii, mai, mij, maj=0, m-1, 0, n-1
            ij = [0, 0, 0]
            d = [-1,1]
            while True:
                 yield matrix[ij[1]][ij[-1]]
                 ij[d[0]]=ij[d[0]]+d[1]
                 if mii<=ij[1]<=mai and mij<=ij[-1]<=maj:
                     pass
                 elif mii>ij[1] or ij[1]>mai:
                     if d[0] == 1 and d[1] == 1:
                         maj=maj-1
                     elif d[0] == 1 and d[1] == -1:
                         mij=mij+1
                     ij[d[0]]=ij[d[0]]-d[1]
                     d[0] = -d[0]
                     d[1] = -d[1]
                     ij[d[0]] = ij[d[0]] + d[1]
                 else:
```

2024-11-18 assignment8.md

```
if d[0] == -1 and d[1] == 1:
                mii = mii + 1
            elif d[0] == -1 and d[1] == -1:
                mai = mai - 1
            ij[d[0]]=ij[d[0]]-d[1]
            d[0] = -d[0]
            ij[d[0]] = ij[d[0]] + d[1]
ge=g(₀)
key=[]
for i in range(m*n):
    key.append(next(ge))
return key
```

代码运行截图 == (至少包含有"Accepted") ==



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

思路:

代码:

```
d=int(input())
n=int(input())
l=[[0]*1025 \text{ for i in range}(1025)]
for _ in range(n):
    x,y,i=map(int,input().split())
    for m in range(max(0,x-d), min(1025,x+d+1)):
        for n in range(max(0,y-d), min(1025,y+d+1)):
             1[m][n]=1[m][n]+i
ma=0
c=1
for i in range(1025):
    for j in range (1025):
        if l[i][j]>ma:
             ma=l[i][j]
             c=1
        elif l[i][j] == ma:
             C=C+1
print(c,ma)
```

代码运行截图 (至少包含有"Accepted")

#47184267提交状态

状态: Accepted

源代码

```
d=int(input())
n=int(input())
l=[[0]*1025 for i in range(1025)]
for in range(n):
    x, y, i=map(int,input().split())
    for m in range(max(0,x-d),min(1025,x+d+1)):
                                                                               提
        for n in range(max(0,y-d),min(1025,y+d+1)):
            l[m][n]=l[m][n]+i
ma=0
c=1
for i in range (1025):
    for j in range(1025):
        if l[i][j]>ma:
            ma=l[i][j]
            c=1
        -1:f 1[:1[:1==ma.
```

基2

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:

```
class Solution:
    def wiggleMaxLength(self, nums: List[int]) -> int:
        l=[[i,1,1] for i in nums]
        1[0][1]=1
        1[0][2]=1
        def g(x,y):
            if y[0]-x[0]>0:
                y[2]=max(y[2],x[1]+1)
            elif y[0]-x[0]<0:
                 y[1]=max(y[1],x[2]+1)
        for i in range(1,len(nums)):
            for j in range(0,i):
                 g(l[j],l[i])
        \#n=\max(\max(l[i][1] \text{ for i in range}(len(nums))),\max(l[i][2] \text{ for i in}
range(len(nums))))
        n=max(l[i][k]for i in range(len(nums)) for k in range(1,3))
        return(n)
        #print(1)
```

代码运行截图 (至少包含有"Accepted")

```
← 全部提交记录
  诵讨
                                                            口 官方题解
                                                                          2 写题解
  Compassionate JepsenqPs 提交于 2024.11.15 19:24
                                       (i)
      ① 执行用时分布
                                              ∰ 消耗内存分布
      111 ms | 击败 5.91%
                                              16.65 MB | 击败 5.01%
      ★ 复杂度分析
     100%
      50%
      0%
                  2ms
                               43ms
                                      47ms
                                            59ms
                                                   73ms
                                                         79ms
                                                                83ms
                                                                       87ms
                                                                             89ms
                     4ms 43ms 47ms 59ms 73ms 79ms 83ms 87ms 89ms
  代码 | Python3
    class Solution:
        def wiggleMaxLength(self, nums: List[int]) -> int:
            l=[[i,1,1] \text{ for } i \text{ in nums}]
            1[0][1]=1
            1[0][2]=1
            def g(x,y):
                if y[0]-x[0]>0:
                    y[2]=max(y[2],x[1]+1)
                Alif v[0]_y[0]<0:
```

CF455A: Boredom

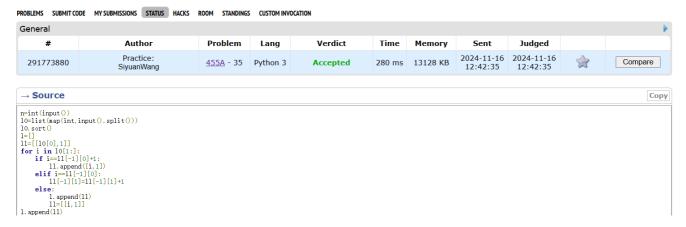
dp, 1500, https://codeforces.com/contest/455/problem/A

思路:

```
n=int(input())
10=list(map(int,input().split()))
10.sort()
l=[]
11=[[10[0],1]]
```

```
for i in 10[1:]:
    if i = = 11[-1][0] + 1:
        11.append([i,1])
    elif i==ll[-1][0]:
        11[-1][1]=11[-1][1]+1
    else:
        1.append(11)
        ll=[[i,1]]
1.append(11)
def g(x):
    if len(x)>2:
        lis=[x[0][0]*x[0][1], \ x[1][0]*x[1][1], \ x[0][0]*x[0][1]+x[2][0]*x[2][1]]
        for i in x[min(len(x),3):]:
            lis.append(\max(lis[-2]+i[0]*i[1], lis[-3]+i[0]*i[1]))
        #print(lis)
        return max(lis[-1],lis[-2])
    else:
        return max(i[0]*i[1] for i in x)
print(sum(g(i) for i in 1))
#print(1)
```

代码运行截图 (至少包含有"Accepted")



02287: Tian Ji -- The Horse Racing

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

思路:

```
while True:
    n=int(input())
    if n==0:
        break
    tl=list(map(int,input().split()))
    kl=list(map(int,input().split()))
    kl.sort(reverse=True)
    tl.sort(reverse=True)
    c=0
```

```
kst,tst=0,0
ked, ted=n-1, n-1
while ted>=tst:
    if tl[tst]>kl[kst]:
        tst=tst+1
        kst=kst+1
        c = c + 200
    elif tl[ted]>kl[ked]:
        ted=ted-1
        ked=ked-1
        c = c + 200
    else:
        if kl[kst]>tl[ted]:
            c=c-200
        ted = ted - 1
        kst = kst + 1
print(c)
```

代码运行截图 (至少包含有"Accepted")

#47208136提交状态

查看

状态: Accepted

```
基本信息
源代码
                                                                                  #: 472
                                                                                题目: 022
 while True:
                                                                              提交人: 24n
     n=int(input())
                                                                                内存: 386
     if n==0:
         break
                                                                                时间: 60㎡
     tl=list(map(int,input().split()))
                                                                                语言: Pyth
     kl=list(map(int,input().split()))
                                                                             提交时间: 202
     kl.sort (reverse=True)
     tl.sort (reverse=True)
     c=0
     kst, tst=0, 0
     ked, ted=n-1, n-1
     while ted>=tst:
         if tl[tqtl>kl[kqtl.
```

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。 这次题确有难度,但感觉不是这几周最难的,摆动序列做出来发现太慢了, 看题解后发现思路确实复杂了, 重写了之后发现没什么可以优化的了没有再提交, leetcode能看到各种运行情况的代码挺好的,就是一开始不太会用, 田忌赛马确实有难度,自己写的代码改了又改,打了一堆补丁之后问了GPT, 但是GPT写的不太对,最后是学了GPT的思路自己改的。感觉收获还是不小的。