# Assignment #8: 田忌赛马来了

Updated 1021 GMT+8 Nov 12, 2024

2024 fall, Complied by <mark>同学的姓名、院系</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: 岛屿周⻓

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

思路：给整个矩阵加一层边，求周长就是求每个1四周0的个数加一块

代码：

```python

n,m=map(int,input().split())

area=[[0]\*(m+2)]

C=0

for i in range(n):

s=[0]+list(map(int,input().split()))+[0]

area.append(s)

area.append([0]\*(m+2))

for i in range(1,n+1):

for j in range(1,m+1):

if area[i][j]==1:

C+=(4-area[i][j-1]-area[i][j+1]-area[i-1][j]-area[i+1][j])

print(C)

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

### 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：向四个方向铺的时候加上限制条件，保证不会铺错，铺不下去了就break出来

代码：

```python

def f(matrix,n):

matrix[1][1]=1

i=1

j=1

while matrix[i][j]!=0:

if matrix[i][j+1]==0 and matrix[i-1][j]!=0:

matrix[i][j+1]=matrix[i][j]+1

j+=1

elif matrix[i+1][j]==0 and matrix[i][j+1]!=0:

matrix[i+1][j]=matrix[i][j]+1

i+=1

elif matrix[i][j-1]==0 and matrix[i+1][j]!=0:

matrix[i][j-1]=matrix[i][j]+1

j-=1

elif matrix[i-1][j]==0 and matrix[i][j-1]!=0:

matrix[i-1][j]=matrix[i][j]+1

i-=1

else:

break

return matrix

n=int(input())

matrix=[[1]\*(n+2)]

for i in range(n):

matrix.append([1]+[0]\*n+[1])

matrix.append([1]\*(n+2))

matrix=f(matrix,n)

for i in range(1,n+1):

print(\*matrix[i][1:n+1],sep=' ')

```

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：先在矩阵中标记每个点投放炸弹可以炸掉的垃圾数量，然后找最大值

代码：

```python

d=int(input())

n=int(input())

msk=[[0]\*1025 for i in range(1025)]

for \_ in range(n):

x,y,i=map(int,input().split())

for j in range(max(0,x-d),min(1025,x+d+1)):

for k in range(max(0,y-d),min(1025,y+d+1)):

msk[j][k]+=i

a=[]

b=0

for i in msk:

a.append(max(i))

c=max(a)

for i in msk:

b+=i.count(c)

print(str(b)+' '+str(c))

```

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：看起来用了dp但实际上好像并没有用。。。代码写太复杂了，题解的方法确实很妙

代码：

```python

class Solution:

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

n=len(nums)

if n==1:

return 1

else:

d=[]

for i in range(n-1):

d.append(nums[i+1]-nums[i])

if sum([i\*\*2 for i in d])==0:

return 1

else:

for i in range(n-1):

if d[i]!=0:

m=i

break

dp=[0]\*n

dp[m]=1

dp[m+1]=2

for i in range(m+1,n-1):

if d[i-1]\*d[i]<0:

dp[i+1]=dp[i]+1

elif d[i-1]\*d[i]>0:

dp[i+1]=dp[i]

elif d[i-1]!=0 and d[i]==0:

x=d[i-1]

dp[i+1]=dp[i]

elif d[i-1]==0 and d[i]!=0:

if x\*d[i]<0:

dp[i+1]=dp[i]+1

else:

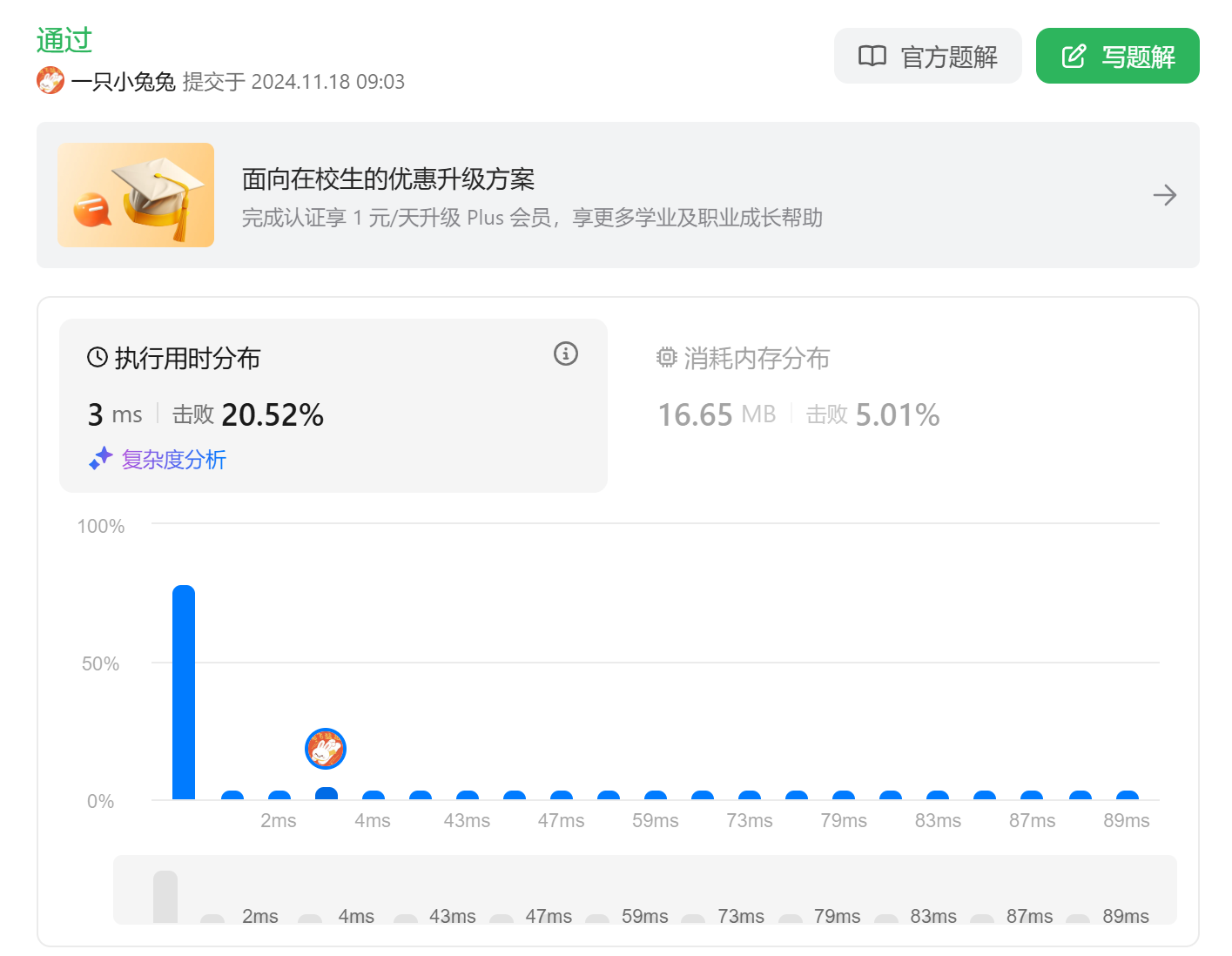
dp[i+1]=dp[i]

else:

dp[i+1]=dp[i]

return dp[n-1]

```

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：把这些数从小到大排，对于某个数i，可以选择获得i分而舍弃i-1的分，也可以要i不要i-1（对于i+1，会在下一个进行讨论，所以此处不用考虑），如果选择后者，就把得分加上i-2时的最高分，一开始加分时使用了i\*a.count(i)导致超时，于是改而使用一个列表储存每个数的数量

代码：

```python

n=int(input())

a=list(map(int,input().split()))

num=[]

a.sort(reverse=True)

dp1=[0]\*(a[0]+1)

for i in a:

dp1[i]+=1

dp=[0]\*(a[0]+1)

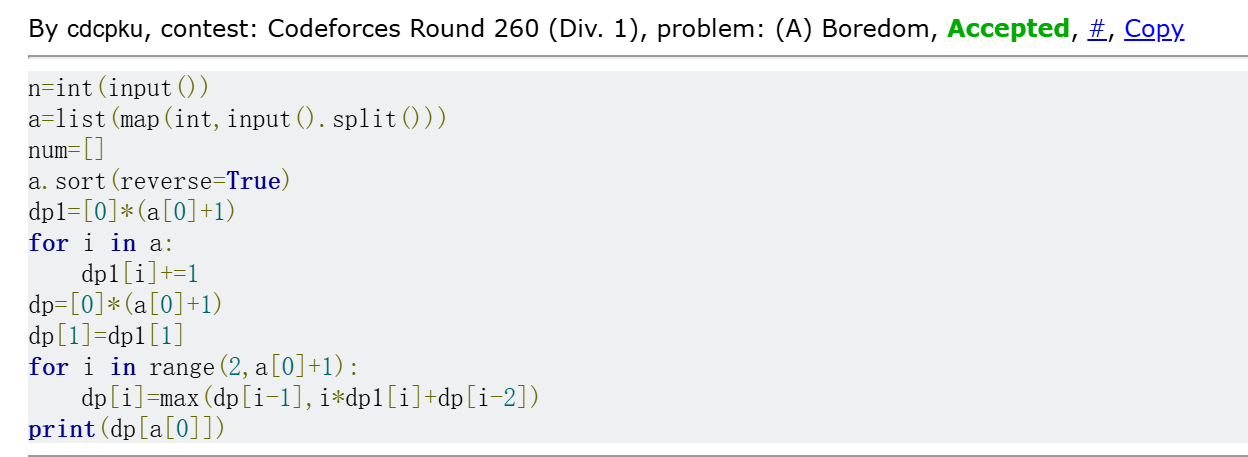
dp[1]=dp1[1]

for i in range(2,a[0]+1):

dp[i]=max(dp[i-1],i\*dp1[i]+dp[i-2])

print(dp[a[0]])

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

### 02287: Tian Ji -- The Horse Racing

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

思路：一直卡在如何处理马的速度相同的情况，因为这时要分类讨论，想不出怎么写成代码，看了题解之后发现完全可以把这里绕过去，真是奇妙！

代码：

```python

while True:

n=int(input())

if n==0:

break

else:

tian=sorted(list(map(int,input().split())))

king=sorted(list(map(int,input().split())))

earning=0

for i in range(n):

if tian[0]>king[0]:

del tian[0]

del king[0]

earning+=200

elif tian[-1]>king[-1]:

del tian[-1]

del king[-1]

earning+=200

elif tian[0]<king[-1]:

del tian[0]

del king[-1]

earning-=200

print(earning)

```

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>

这周补了一丁点每日选做，感觉做作业有些吃力，摆动序列疯狂打补丁写出来个屎山代码，田忌赛马更是被逼到看题解，有些corner case真的想不到，oj上还看不了数据，如果考试时wa了却想不出是什么corner case的话感觉很危啊