**# Assignment #B: Dec Mock Exam大雪前一天**

Updated 1649 GMT+8 Dec 5, 2024

2024 fall, Complied by <mark>同学的姓名、院系</mark>

**\*\*说明：\*\***

1）⽉考： AC3<mark>（请改为同学的通过数）</mark> 。考试题⽬都在“题库（包括计概、数算题目）”⾥⾯，按照数字题号能找到，可以重新提交。作业中提交⾃⼰最满意版本的代码和截图。

2）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。

3）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。

4）如果不能在截止前提交作业，请写明原因。

**## 1. 题目**

**### E22548: 机智的股民老张**

http://cs101.openjudge.cn/practice/22548/

思路：

不断维护最高的收益

代码：

```python

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

num=0

minn=100000000

for i in array:

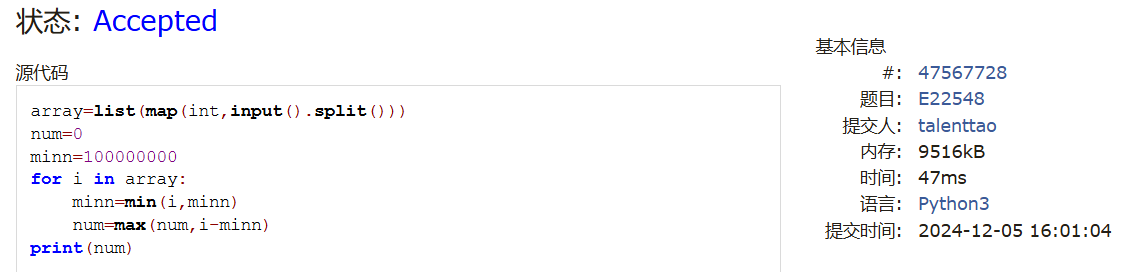
minn=min(i,minn)

num=max(num,i-minn)

print(num)

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



**### M28701: 炸鸡排**

greedy, http://cs101.openjudge.cn/practice/28701/

思路：

贪心即可

代码：

```python

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

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

lst.sort(reverse=True)

tot=sum(lst)

res=k

while True:

if res==1:

print(f"{tot:.3f}")

break

ave=tot/res

if lst[0]<=ave:

print(f"{ave:.3f}")

break

else:

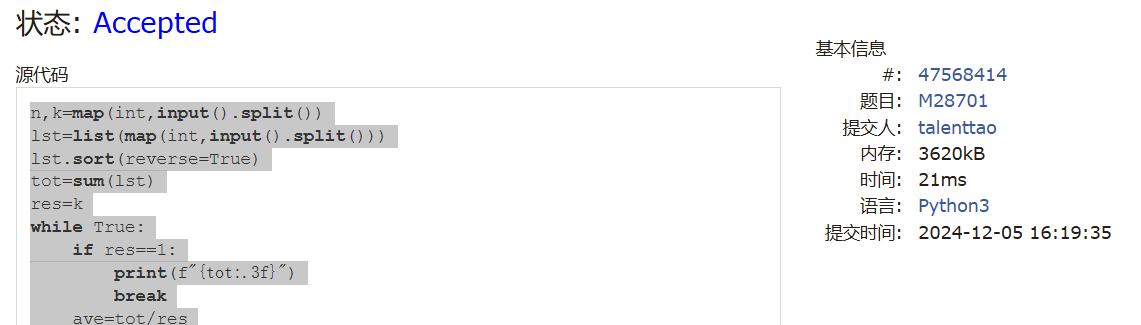
res-=1

tot-=lst[0]

lst=lst[1:]

```

代码运行截图 ==（至少包含有"Accepted"）==



**### M20744: 土豪购物**

dp, http://cs101.openjudge.cn/practice/20744/

思路：

使用两个数组dp

代码：

```python

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

dp1 = [0] \* len(a);

dp2 = [0] \* len(a)

dp1[0] = a[0];

dp2[0] = a[0]

for i in range(1, len(a)):

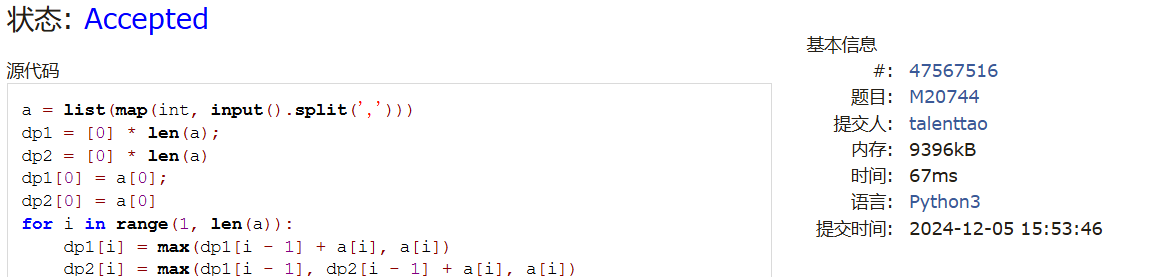
dp1[i] = max(dp1[i - 1] + a[i], a[i])

dp2[i] = max(dp1[i - 1], dp2[i - 1] + a[i], a[i])

print(max(dp2))

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



**### T25561: 2022决战双十一**

brute force, dfs, http://cs101.openjudge.cn/practice/25561/

思路：

最开始没想到如何dfs，参考了答案写的

代码：

```python

result = float("inf")

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

store\_prices = [input().split() for \_ in range(n)]

coupons = [input().split() for \_ in range(m)]

def dfs(store\_prices, coupons, items=0, total\_price=0, each\_store\_price=[0] \* m):

    global result

    if items == n:

        coupon\_price = 0

        for i in range(m):

            store\_p = 0

            for coupon in coupons[i]:

                a, b = map(int, coupon.split('-'))

                if each\_store\_price[i] >= a:

                    store\_p = max(store\_p, b)

            coupon\_price += store\_p

        result = min(result, total\_price - (total\_price // 300) \* 50 - coupon\_price)

        return

    for i in store\_prices[items]:

        idx, p = map(int, i.split(':'))

        each\_store\_price[idx - 1] += p

        dfs(store\_prices, coupons, items + 1, total\_price + p, each\_store\_price)

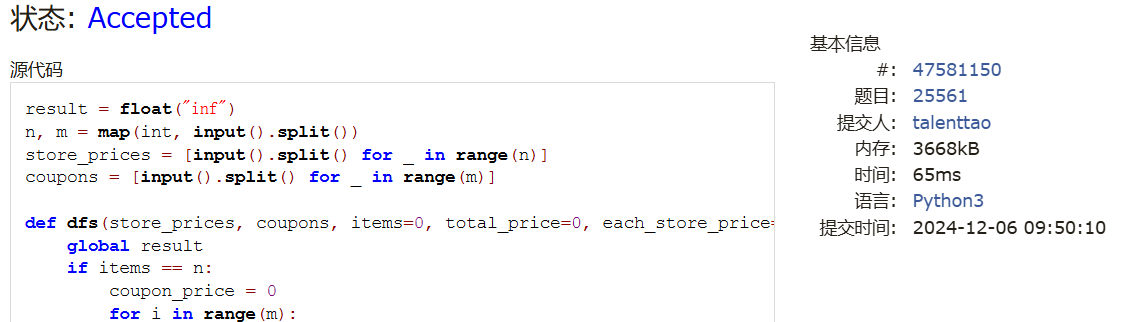
        each\_store\_price[idx - 1] -= p

dfs(store\_prices, coupons)

print(result)

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



**### T20741: 两座孤岛最短距离**

dfs, bfs, http://cs101.openjudge.cn/practice/20741/

思路：

先用dfs得到一个岛，再用bfs求最短路径

代码：

```python

from collections import deque

n=int(input())

land=[list(map(int, input())) for \_ in range(n)]

dx=[0,0,1,-1]

dy=[1,-1,0,0]

queue = deque()

def dfs(x, y, land, n, queue):

    land[x][y]=2

    queue.append((x,y,0))

    for i in range(4):

        nx,ny=x+dx[i],y+dy[i]

        if 0 <= nx < n and 0 <= ny < n and land[nx][ny] == 1:

            dfs(nx,ny,land,n,queue)

def bfs(land,n,queue):

    while queue:

        x,y,length=queue.popleft()

        for i in range(4):

            nx, ny = x + dx[i], y + dy[i]

            if 0 <= nx < n and 0 <= ny < n:

                if land[nx][ny]==1:

                    return length

                elif land[nx][ny]==0:

                    land[nx][ny]=2

                    queue.append((nx,ny,length+1))

def output(land,n):

    for i in range(n):

        for j in range(n):

            if land[i][j]==1:

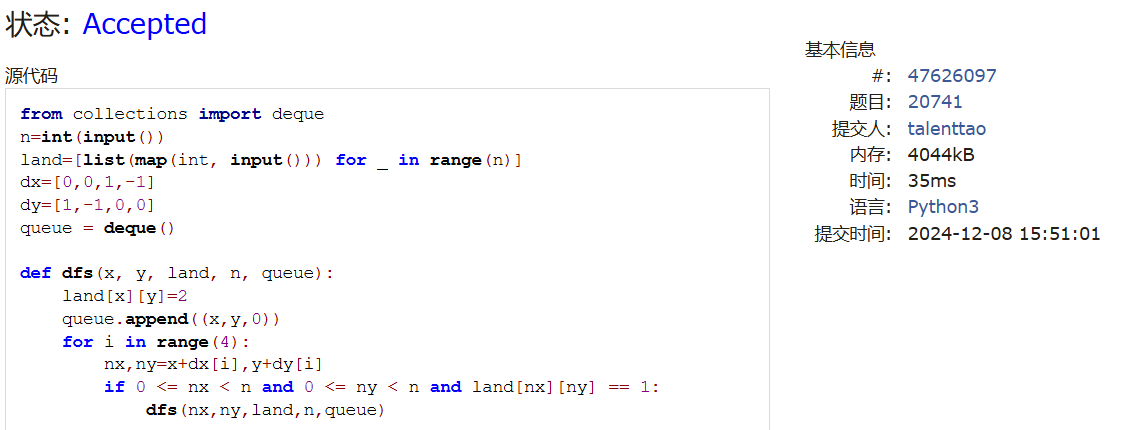
                dfs(i, j, land, n, queue)

                return bfs(land,n,queue)

print(output(land,n))

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



**### T28776: 国王游戏**

greedy, http://cs101.openjudge.cn/practice/28776

思路：

只需推导出最优条件即可

代码：

```python

n=int(input())

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

array=[]

ans=0

for i in range(n):

    array.append(list(map(int,input().split())))

array.sort(key=lambda x:(x[0]\*x[1]))

for i in range(n):

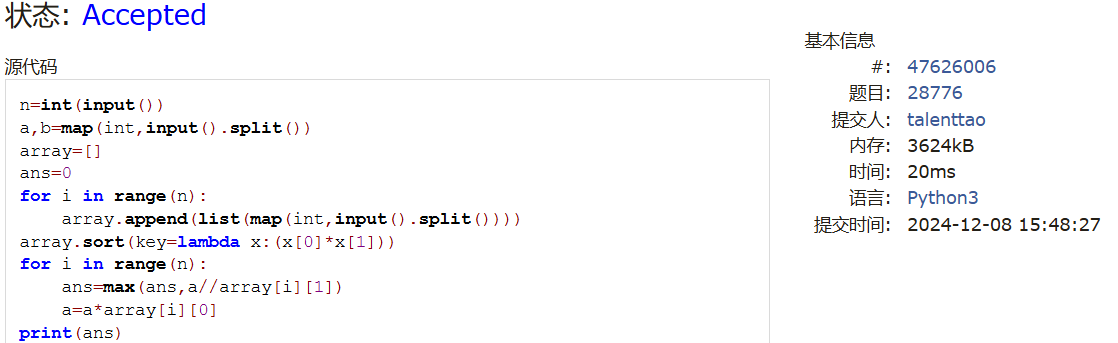
    ans=max(ans,a//array[i][1])

    a=a\*array[i][0]

print(ans)

```

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>

本周只做了每日练习，感觉dfs与bfs掌握欠佳，感觉从“知道怎么写”到“写对”还是要花很多的功夫的。