

## 1. 题目

E22548: 机智的股民老张

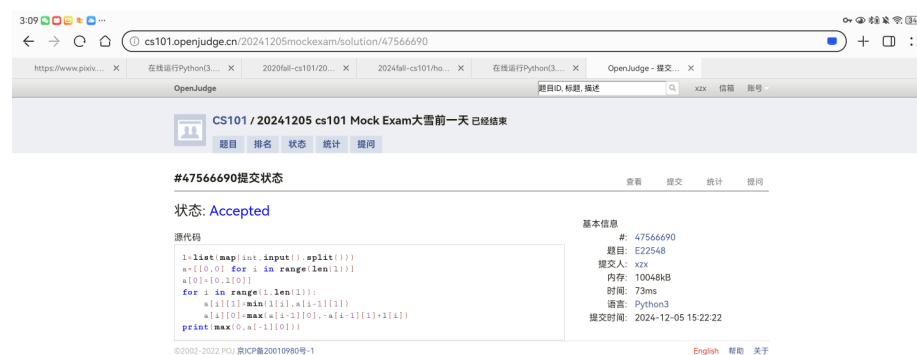
<http://cs101.openjudge.cn/practice/22548/>

思路:

代码:

```
l=list(map(int,input().split()))
a=[[0,0] for i in range(len(l))]
a[0]=[0,l[0]]
for i in range(1,len(l)):
    a[i][1]=min(l[i],a[i-1][1])
    a[i][0]=max(a[i-1][0],-a[i-1][1]+l[i])
print(max(0,a[-1][0]))
```

代码运行截图（至少包含有"Accepted"）



M28701: 炸鸡排

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

思路:

代码:

```
n,k=map(int,input().split())
l=list(map(int,input().split()))
l.sort(reverse=True)
t=sum(l)/k
while t<l[0]:
    while l[0]>=t:
        del l[0]
    k-=1
```

The screenshot shows a web browser window with the address bar displaying 'cs101.openjudge.cn/practice/solution/47621523'. The page content includes the OpenJudge logo, the title 'CS101 / 题库 (包括计概、数算题目)', and a table with columns '题目', '排名', '状态', and '提交'. The problem number '#47621523提交状态' is highlighted. The status is 'Accepted'. The code is written in Python3 and implements a sorting algorithm. The submission time is 2024-12-08 12:10:37.

思路：

```
代码:
l=list(map(int,input().split(',')))
a=[[0,0] for i in range(len(l))]
a[0]=[l[0],l[0]]
a[1]=[max(l[0]+l[1],l[1]),max(l[1],l[0]+l[1])]
for i in range(2,len(l)):
    a[i][0]=max(l[i],l[i]+a[i-1][0])
    a[i][1]=max(l[i]+a[i-2][0],l[i]+a[i-1][1],l[i])
s=a[0][-1]
for k in range(len(l)):
    if a[k][-1]>s:
        s=a[k][-1]
print(s)
代码运行截图 （至少包含有"Accepted"）
```



T25561: 2022 决战双十一

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

思路：

代码:

```
n,m=map(int,input().split())
price=[[0 for i in range(m)]for j in range(n)]
for i in range(n):
    a=input().split()
    for k in a:
        b=list(map(int,k.split(':')))
        price[i][b[0]-1]=b[1]
dis=[]
for i in range(m):
    a=input().split()
    for k in a:
        b=list(map(int,k.split('-')))
        dis[i].append(b)
```

```
dic={}
for i in range(1,m+1):
    dic[i]=0
ms=0
def dfs(i,dic):
    global ms,discount
    if i==n+1:
```

```

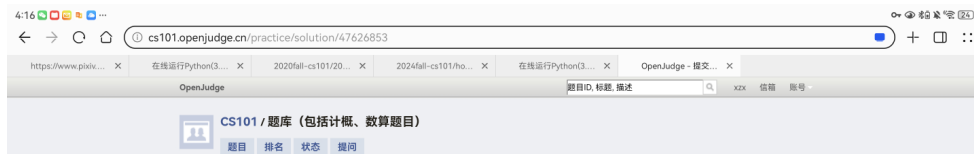
        ns=jiage(dic)
        if ns<ms or ms==0:
            ms=ns
    else:
        for k in range(m):
            if price[i-1][k]!=0:
                dic[k+1]+=price[i-1][k]
                dfs(i+1,dic)
                dic[k+1]-=price[i-1][k]
def jiage(dic):
    global dis
    s=0
    sdis=0
    for j in range(1,1+m):
        s+=dic[j]
        if dic[j]!=0:
            sdis+=find(dic[j],dis[j-1])
    sdis+=50*(s//300)
    return s-sdis
def find(y,l):
    if y<l[0][0]:
        return 0
    else:
        q=0
        i=0
        while i<len(l) and l[i][0]<=y:
            q=max(q,l[i][1])
            i+=1
        return q

```

dfs(1,dic)

print(ms)

代码运行截图 （至少包含有"Accepted"）



#47626853提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
n,m=map(int,input().split())
price=[[0 for i in range(m)]for j in range(n)]
for i in range(n):
    a=input().split()
    for k in a:
        b=list(map(int,k.split(':')))
        price[i][b[0]-1]=b[1]
dis=[[1]for i in range(m)]
for i in range(m):
    a=input().split()
    for k in a:
        b=list(map(int,k.split('-')))
        dis[i].append(b)

dic={}
for i in range(1,m+1):
    dic[i]=0
ms=0
def dfs(i,dic):
    global ms,discount
    if i==n+1:
        ms=jiahe(dic)
        if ms<ms or ms==0:
            ms=ms
    else:
        for k in range(m):
            if price[i-1][k]!=0:
                dic[k+1]=price[i-1][k]
                dfs(i+1,dic)
                dic[k+1]-=price[i-1][k]
def jiahe(dic):
    global dis
    res=0
    for i in range(1,m+1):
        res+=dis[i]*dic[i]
    return res
```

基本信息

# 47626853  
题目: 25561  
提交人: xzx  
内存: 3740kB  
时间: 40ms  
语言: Python3  
提交时间: 2024-12-08 16:16:32

T20741: 两座孤岛最短距离

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

思路:

代码:

```
n=int(input())
```

```
l=[]
```

```
for i in range(n):
```

```
    a=input()
```

```
    l.append([int(k) for k in a])
```

```
step=[[1,0],[-1,0],[0,1],[0,-1]]
```

```
def find(x,y,st):
```

```
    global que,l,step
```

```
    l[x][y]=-1
```

```
    for k in step:
```

```
        nx=x+k[0]
```

```
        ny=y+k[1]
```

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

```
            st.append([nx,ny])
```

```
            find(nx,ny,st)
```

```
island=[[[],[]]]
```

```
p=0
```

```
for i in range(n):
```

```
    for j in range(n):
```

```
        if l[i][j]==1 and p!=2:
```

```
            island[p].append([i,j])
```

```
            find(i,j,island[p])
```

```

        p+=1
mdis=0
for p in island[0]:
    for q in island[1]:
        dis=abs(p[0]-q[0])+abs(p[1]-q[1])-1
        if dis<mdis or mdis==0:
            mdis=dis
print(mdis)

```

代码运行截图（至少包含有"Accepted"）

The screenshot shows a web browser window displaying the OpenJudge submission page for problem 47627372. The page title is "CS101 / 题库 (包括计概、数算题目)". The submission status is "Accepted". The code is a Python solution for finding the minimum distance between two islands in a grid. The code uses a BFS approach to find the distance between two islands.

```

源代码
n=int(input())
l=[]
for i in range(n):
    a=input()
    l.append([int(k) for k in a])
step=[[1,0],[0,1],[-1,0],[0,-1]]
def find(x,y,st):
    global que,l,step
    l[x][y]=-1
    for k in step:
        nx=x+k[0]
        ny=y+k[1]
        if 0<=nx<n and 0<=ny<n and l[nx][ny]==1:
            st.append([nx,ny])
            find(nx,ny,st)
island=[[],[]]
p=0
for i in range(n):
    for j in range(n):
        if l[i][j]==1 and p==2:
            island[p].append([i,j])
            find(i,j,island[p])
            p+=1
mdis=0
for p in island[0]:
    for q in island[1]:
        dis=abs(p[0]-q[0])+abs(p[1]-q[1])-1
        if dis<mdis or mdis==0:
            mdis=dis
print(mdis)

```

基本信息

- #: 47627372
- 题目: 20741
- 提交人: xzx
- 内存: 4244kB
- 时间: 394ms
- 语言: Python3
- 提交时间: 2024-12-08 16:34:38

T28776: 国王游戏

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

思路:

代码:

```

n=int(input())
a,b=map(int,input().split())
l=[]
for i in range(n):
    l.append(list(map(int,input().split())))
l.sort(key=lambda x:x[0]*x[1])
front=a
s=a//l[0][1]
for i in range(1,n):
    front*=l[i-1][0]
    ns=front//l[i][1]

```

```
if ns>s:
    s=ns
print(s)
```

代码运行截图（至少包含有"Accepted"）



## 2. 学习总结和收获

这次考试只 ac 了 2 题，是土豪和股民小张（说明前段时间做每日选做的 dp 题目有效果了）。

国王游戏很可惜，想到了类排序方法，但在判断条件上出了小错没能 ac，后来好好想了下直接用乘积判断即可，考试时没有用数学化证明导致想当然出错（这题其实也提示了一种技巧，即所有的元素优劣性只与各元素自身性质有关的最优问题都可用类排序思路）。

而小岛联通问题用 bfs 超时了，又没 ac，考后发现不用 bfs 找，直接用 bfs 分别找出两岛位置，再遍历求距离取最小即可（可证明此法正确性，想出此法时简直惊为天人），思维还是有点固化。

炸鸡排真的在考试里很难想清楚所有鸡排都小于平均时为什么一定可构造（现在想到一简单证明，把所有鸡排时间视作条状，首尾相连，把这条时间平分为 k 段，每段头部对齐，这样就构造出一种方案，且由于所有鸡排都小于平均，一定不会有同一鸡排同时出现在不同锅里），当时直接跳过。

双十一来不及做，甚至题都没看，考完做感觉十分繁琐，对代码细节要求很高，花了不少时间才 ac。

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。

日常跟进 OJ 每日选做