

Assignment #B: Dec Mock Exam大雪前一天

Updated 1649 GMT+8 Dec 5, 2024

2024 fall, Compiled by 胡杨 元培学院

说明:

- 1) 月考: AC? (未参加月考)。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: 本来想写用目前的价格减去前面最小的价格, 发现会超时。决定改为写储存到第i天为止的最低价格和最高价格, 注意最高价格必须在最低价格之后出现, 所以如果最低价格更新了, 最高价格需要同时更新为和他一样的值, 不管前面的最高价有多高。同时储存一个max_profit, 将目前的max_profit和max_price-min_price相比较, 不断更新目前获得的最高利益

耗时: 30min (寄)

代码:

```
a=list(map(int,input().split()))
n=len(a)
min_price=float('-inf')
max_price=float('-inf')
max_profit=0
for i in range(n):
    if a[i]<min_price:
        min_price=a[i]
        max_price=a[i]
        max_profit=max(max_profit,max_price-min_price)
    if a[i]>max_price:
        max_price=a[i]
        max_profit=max(max_profit,max_price-min_price)
print(max_profit)
```

代码运行截图 (至少包含有"Accepted")

OpenJudge

题目ID, 标题, 描述

24n2300017728

信箱

账号

CS101 / 题库 (包括计概、数算题目)

题目 排名 状态 提问

#47602918提交状态

查看 提交 统计 提问

状态: Accepted

源代码

pre>a=list(map(int,input().split()))
n=len(a)
min_price=float('-inf')
max_price=float('-inf')
max_profit=0
for i in range(n):
 if a[i]<min_price:
 min_price=a[i]
 max_price=a[i]
 max_profit=max(max_profit,max_price-min_price)
 if a[i]>max_price:
 max_price=a[i]
 max_profit=max(max_profit,max_price-min_price)
print(max_profit)</pre>

基本信息

pre>#: 47602918
题目: 22548
提交人: 24n2300017728
内存: 9552kB
时间: 37ms
语言: Python3
提交时间: 2024-12-07 11:31:28</pre>

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

M28701: 炸鸡排

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

思路：卧槽，这啥啊，给我干哪来了，这还是M吗

去群里找了点大佬的思路：最理想的情况是每个锅里的位置都能炸sum(t)/k的时间，记为avg，但如果有的鸡排炸的时间超过了avg，说明计算avg的时候存在“把这个大鸡排拆成几部分来补其他空位置”的情况，需要单独考虑。这种时候就把这种大鸡排占掉一个位置一直炸，剩下几个位置同理。

耗时：看了思路，40min

代码：

```
n,k=map(int,input().split())
t=list(map(int,input().split()))
t.sort(reverse=True)
s=sum(t)
for i in range(n):
    if t[i]>s/k:
        s-=t[i]
        k-=1

print(f'{s/k:.3f}')
```

代码运行截图 (至少包含有"Accepted")

OpenJudge

题目ID, 标题, 描述

24n2300017728

信箱

账号

CS101 / 题库 (包括计概、数算题目)

题目

排名

状态

提问

#47605306提交状态

查看

提交

统计

提问

状态: Accepted

源代码

基本信息

```
n, k = map(int, input().split())
t = list(map(int, input().split()))
t.sort(reverse=True)
s = sum(t)
for i in range(n):
    if t[i] > s/k:
        s -= t[i]
        k -= 1

print(f' {s/k:.3f}')
```

©2002-2022 POJ 京ICP备20010980号-1

#: 47605306

题目: 28701

提交人: 24n2300017728

内存: 3628KB

时间: 21ms

语言: Python3

提交时间: 2024-12-07 14:05:26

English

帮助

关于

M20744: 土豪购物

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

思路：类似于最大上升子序列和，只是可以放回元素。建立两个dp数组，其中一个表示最大上升子序列和，另一个表示可以放回当前元素的最大上升子序列和。但是放回元素的状态转移方程我是真的想不通

耗时：20min（查看群中的放回元素的状态转移方程）

代码：

```
a=list(map(int,input().split(',')))
n=len(a)
dp1=[0]*n #以第i个元素结尾的最大子序列和，不放回
dp2=[0]*n #以第i个元素结尾的最大子序列和，且可以放回一个元素
dp1[0]=a[0]
dp2[0]=a[0]
for i in range(1,n):
    dp1[i]=max(dp1[i-1]+a[i],a[i])
    dp2[i]=max(dp1[i-1],dp2[i-1]+a[i],a[i]) #dp1[i-1]放回当前元素

print(max(dp2))
```

代码运行截图（至少包含有"Accepted"）

OpenJudge

题目ID, 标题, 描述

24n2300017728

信箱

账号

CS101 / 题库 (包括计概、数算题目)

题目

排名

状态

提问

#47605890提交状态

查看

提交

统计

提问

状态: Accepted

源代码

基本信息

```
a=list(map(int,input().split(',')))
n=len(a)
dp1=[0]*n #以第i个元素结尾的最大子序列和, 不放回
dp2=[0]*n #以第i个元素结尾的最大子序列和, 且可以放回一个元素
dp1[0]=a[0]
dp2[0]=a[0]
for i in range(1,n):
    dp1[i]=max(dp1[i-1]+a[i],a[i])
    dp2[i]=max(dp1[i-1],dp2[i-1]+a[i],a[i]) #dp1[i-1]放回当前元素
print(max(dp2))
```

#: 47605890

题目: 20744

提交人: 24n2300017728

内存: 9428kB

时间: 66ms

语言: Python3

提交时间: 2024-12-07 14:25:52

©2002-2022 POJ 京ICP备20010980号-1

English

帮助

关于

T25561: 2022决战双十一

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

思路：唯一想到的办法就是把所有情况算出来取最小，好崩溃。计概很强的同学让我学习用dfs写枚举，好的。每一件商品有不同的店售卖相当于不同的岔路口，先逮着排在第一位的店买，看最后买了多少钱，由此遍历所有买东西的情况。取最小值

耗时：1.5h

代码：

```
n,m=map(int,input().split()) #n件商品, m家店
goods=[]
for _ in range(n):
    goods.append(list(input().split())) #第i件商品的出售店铺和价格
yohui=[]
for _ in range(m):
    yohui.append(list(input().split())) #第i家店的优惠券
result=float('inf')
sold=[0]*m #第i家店的已买的价格

def dfs(i,sum_price): #i表示当前商品编号(0-(n-1)), sum_price表示当前购物车的总价
    global result
    if i==n: #所有商品都已购买完
        result_i=0
        #店铺计算满减
        for i1 in range(m): #第i1家店
            price=float('inf')
            for quan in yohui[i1]: #第i1家店的每种优惠券
                man,jian=map(int,quan.split('-'))
                if sold[i1]>=man: #满减条件成立
                    price=min(price,sold[i1]-jian)

            if price!=float('inf'): #有满减优惠
```

```

        result_i+=price
    else: #无满减优惠
        result_i+=sold[i1]
    #计算跨店满减并更新全局最优
    result=min(result,result_i-(sum_price//300)*50)
    return

for shoujia in goods[i]:
    k,price_i=map(int,shoujia.split(':')) #第i件商品的第k家店的价格
    sold[k-1]+=price_i #第k家店的已买的价格增加
    sum_price+=price_i #购物车总价增加
    dfs(i+1,sum_price) #购买第i+1件商品
    #回溯
    sold[k-1]-=price_i #第k家店的已买的价格减少
    sum_price-=price_i #购物车总价减少

dfs(0,0)
print(result)

```

代码运行截图 (至少包含有"Accepted")

OpenJudge 题目ID, 标题, 描述 24n2300017728 信箱 账号

CS101 / 题库 (包括计概、数算题目)

题目 排名 状态 提问

#47642727提交状态 查看 提交 统计 提问

状态: Accepted

源代码

```

n,m=map(int,input().split()) #n件商品, m家店
goods=[]
for _ in range(n):
    goods.append(list(input().split())) #第i件商品的出售店铺和价格
youhui=[]
for _ in range(m):
    youhui.append(list(input().split())) #第i家店的优惠券
result=float('inf')
sold=[0]*m #第i家店的已买的价格

def dfs(i,sum_price): #i表示当前商品编号 (0~ (n-1) ), sum_price表示当前购物车
    global result
    if i==n: #所有商品都已购买完
        result_i=0
        #店铺计算满减
        for i1 in range(m): #第i1家店
            price=float('inf')
            for quan in youhui[i1]: #第i1家店的每种优惠券
                man,jian=map(int,quan.split('-'))
                if sold[i1]>=man: #满减条件成立
                    price=min(price,sold[i1]-jian)

```

基本信息

#: 47642727
 题目: 25561
 提交人: 24n2300017728
 内存: 3676kB
 时间: 73ms
 语言: Python3
 提交时间: 2024-12-09 16:40:25

T20741: 两座孤岛最短距离

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

思路: 先通过dfs找到其中的一座岛屿, 仅把它标记为2, 然后从2出发bfs寻找1.也想到了用dijkstra, 1-1权重0, 1-0权重1, 0-0权重1, 0-1就是找到另一座岛了, 但是不知道怎么写

耗时: 很久

代码:

```

from collections import deque

```

```

dx=[0,0,-1,1]
dy=[1,-1,0,0]
def find(x,y):
    a[x][y]=2
    q.append((0,(x,y))) #(step,(x,y))
    for i in range(4):
        nx=x+dx[i]
        ny=y+dy[i]
        if 0<=nx<n and 0<=ny<n and a[nx][ny]==1:
            find(nx,ny)

def bfs(q,a):
    while q:
        step,(x,y)=q.popleft()
        for i in range(4):
            nx=x+dx[i]
            ny=y+dy[i]
            if 0<=nx<n and 0<=ny<n:
                if a[nx][ny] == 1:
                    return step
                elif a[nx][ny]==0:
                    a[nx][ny]=2
                    q.append((step+1,(nx,ny)))

n=int(input())
a=[list(map(int,input())) for _ in range(n)]
q=deque()
ans=0

for i in range(n):
    for j in range(n):
        if a[i][j]==1:
            find(i,j)
            ans=bfs(q,a)
            break
    if ans:
        break

print(ans)

```

代码运行截图 (至少包含有"Accepted")

OpenJudge 题目ID, 标题, 描述 24n2300017728 信箱 账号

CS101 / 题库 (包括计概、数算题目)

题目 排名 状态 提问

#47639891 提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
from collections import deque

dx=[0,0,-1,1]
dy=[1,-1,0,0]
def find(x,y):
    a[x][y]=2
    q.append((0,(x,y))) # (step,(x,y))
    for i in range(4):
        nx=x+dx[i]
        ny=y+dy[i]
        if 0<=nx<n and 0<=ny<n and a[nx][ny]==1:
            find(nx,ny)

def bfs(q,a):
    while q:
        step,(x,y)=q.popleft()
        for i in range(4):
            nx=x+dx[i]
            ny=y+dy[i]
            if 0<=nx<n and 0<=ny<n:
                if a[nx][ny]==1:
                    return step
```

基本信息

#: 47639891
 题目: 20741
 提交人: 24n2300017728
 内存: 4264kB
 时间: 32ms
 语言: Python3
 提交时间: 2024-12-09 14:17:44

T28776: 国王游戏

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

思路: 看的题解, 完全不会, 整理如下图:

但是我并没有理解为什么最大值在最后一个人处取到, 整个都不太懂

换顺序:

person	left	right	person	left	right
国王 p_0	a_0	b_0	p_0	a_0	b_0
p_1	a_1	b_1	p_2	a_2	b_2
p_2	a_2	b_2	p_1	a_1	b_1

$$ans_1 = \max \left\{ \frac{a_0}{b_1}, \frac{a_0 a_1}{b_2} \right\}$$

$$ans_2 = \max \left\{ \frac{a_0}{b_2}, \frac{a_0 a_2}{b_1} \right\}$$

设 $ans_1 > ans_2$

$$\frac{a_0}{b_1} < \frac{a_0 a_2}{b_1} \Rightarrow \frac{a_0}{b_1} < ans_2$$

$$\Rightarrow \frac{a_0 a_1}{b_2} > ans_2$$

$$\Rightarrow \frac{a_0 a_1}{b_2} > \frac{a_0}{b_2} \checkmark \quad \text{且} \quad \frac{a_0 a_1}{b_2} > \frac{a_0 a_2}{b_1}$$

$$\Rightarrow a_1 b_1 > a_2 b_2$$

\Rightarrow 调换顺序的目标是让排在后面的人左右手乘积 max

代码：

```
n=int(input())
a0,b0=map(int,input().split())
total=a0
m=1
for _ in range(n):
    a,b=map(int,input().split())
    total*=a
    m=max(m,a*b)

print(total//m)
```

代码运行截图 (至少包含有"Accepted")

OpenJudge

题目ID, 标题, 描述

24n2300017728

信箱

账号

CS101 / 题库 (包括计概、数学题目)

题目 排名 状态 提问

#47609527提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
n=int(input())
a0,b0=map(int,input().split())
total=a0
m=1
for _ in range(n):
    a,b=map(int,input().split())
    total*=a
    m=max(m,a*b)

print(total//m)
```

基本信息

#: 47609527

题目: 28776

提交人: 24n2300017728

内存: 3612kB

时间: 22ms

语言: Python3

提交时间: 2024-12-07 16:45:20

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

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。

太恶心了，感觉考试我顶多AC1，及不了格了XD

感受到了贪心的思路可以如此难想，想出来了就想出来了，想不出来是真的也不会做，那这种题目有什么办法啊（悲）

用dfs写枚举是一项强大的技能，本周想要在每日选座里找找练习一些枚举和dp