

Assignment #B: Dec Mock Exam大雪前一天

Updated 1649 GMT+8 Dec 5, 2024

2024 fall, Compiled by 万馨雅 城环

说明：

- 1) 月考：AC6 (0)。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：在考场上写的几个都TLE了，应该可以做出来的

代码：

```
a=list(map(int,input().split()))
min_num=a[0]
max_num=0
for i in a:
    if i < min_num:
        min_num=i
    max_num=max(max_num,i-min_num)
print(max_num)
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
a=list(map(int,input().split()))
min_num=a[0]
max_num=0
for i in a:
    if i < min_num:
        min_num=i
    max_num=max(max_num,i-min_num)
print(max_num)
```

基本信息

#: 47604930
题目: 22548
提交人: [erxiduodianlazi](#)
内存: 9584kB
时间: 41ms
语言: Python3
提交时间: 2024-12-07 13:48

M28701: 炸鸡排

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

思路:

代码:

```
n,k=map(int,input().split())
c=list(map(int,input().split()))
c.sort()
s=sum(c)
while True:
    if s/k < c[-1]:
        s-=c.pop()
        k-=1
    else:
        print('{:.3f}'.format(s/k))
        break
```

代码运行截图 == (至少包含有"Accepted") ==

#47607393提交状态

状态: Accepted

基本信息

源代码

```
n,k=map(int,input().split())
c=list(map(int,input().split()))
c.sort()
s=sum(c)
while True:
    if s/k < c[-1]:
        s-=c.pop()
        k-=1
    else:
        print('{:.3f}'.format(s/k))
        break
```

题
提交
内
时
语
提交时

©2002-2022 POJ 京ICP备20010980号-1

M20744: 土豪购物

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

思路:

代码:

```
c=list(map(int,input().split(',')))
dp1=[0]*len(c)
dp2=[0]*len(c)
dp1[0]=dp2[0]=c[0]
for i in range(1,len(c)):
    dp1[i]=max(c[i],dp1[i-1]+c[i])
    dp2[i]=max(dp1[i-1],dp2[i-1]+c[i],c[i])
print(max(dp2))
```

代码运行截图 (至少包含有"Accepted")

```
c=list(map(int,input().split(',')))
dp1=[0]*len(c)
dp2=[0]*len(c)
dp1[0]=dp2[0]=c[0]
for i in range(1,len(c)):
    dp1[i]=max(c[i],dp1[i-1]+c[i])
    dp2[i]=max(dp1[i-1],dp2[i-1]+c[i],c[i])
print(max(dp2))
```

T25561: 2022决战双十一

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

思路：这个真看不懂

代码：

```
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 (至少包含有"Accepted")

状态: Accepted

源代码

```
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):
    global result
    if items == n:
```

T20741: 两座孤岛最短距离

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

思路:

代码:

```
from collections import deque
def dfs(x,y,c,queue):
    c[x][y]=2
    queue.append((x,y))
    for dx,dy in directions:
        nx,ny=x+dx,y+dy
        if 0<=nx<n and 0<=ny<n and c[nx][ny]==1:
            dfs(nx,ny,c,queue)

distance=0
def bfs(c,queue):
    global distance
    while queue:
        for _ in range(len(queue)):
            x,y=queue.popleft()
            for dx,dy in directions:
                nx,ny=x+dx,y+dy
                if 0<=nx <n and 0<=ny<n:
                    if c[nx][ny]==1:
                        return distance
                    elif c[nx][ny]==0:
                        c[nx][ny]=2
                        queue.append((nx,ny))
            distance+=1
        return distance
n = int(input())
c=[list(map(int,input())) for i in range(n)]
directions=[(-1,0),(0,1),(1,0),(0,-1)]
queue=deque()
def jisuan(n,c):
```

```

for i in range(n):
    for j in range(n):
        if c[i][j]==1:
            dfs(i,j,c,queue)
    return bfs(c,queue)
print(jisuan(n,c))

```

代码运行截图 (至少包含有"Accepted")

POJ 3984 迷宫问题

状态: Accepted

源代码

```

from collections import deque
def dfs(x,y,c,queue):
    c[x][y]=2
    queue.append((x,y))
    for dx,dy in directions:
        nx,ny=x+dx,y+dy
        if 0<=nx<n and 0<=ny<n and c[nx][ny]==1:
            dfs(nx,ny,c,queue)
distance=0
def bfs(c,queue):
    global distance
    while queue:
        for _ in range(len(queue)):
            x,y=queue.popleft()
            for dx,dy in directions:
                nx,ny=x+dx,y+dy
                if 0<=nx <n and 0<=ny<n:
                    if c[nx][ny]==1:
                        return distance
                    elif c[nx][ny]==0:
                        c[nx][ny]=2
                        queue.append((nx,ny))
            distance+=1
        return distance
n = int(input())
c=[list(map(int,input())) for i in range(n)]
directions=[(-1,0),(0,1),(1,0),(0,-1)]
queue=deque()
def jisuan(n,c):
    for i in range(n):
        for j in range(n):
            if c[i][j]==1:
                dfs(i,j,c,queue)
    return bfs(c,queue)
print(jisuan(n,c))

```

T28776: 国王游戏

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

思路:

代码:

```
n=int(input())
a0,b0=map(int,input().split())
c=[]
for _ in range(n):
    c.append(list(map(int,input().split())))
c.sort(key=lambda x:(x[0]*x[1]))
result=0
for i in range(n):
    result=max(result,a0//c[i][1])
    a0*=c[i][0]
print(result)
```

代码运行截图 (至少包含有"Accepted")

#47640399提交状态

状态: Accepted

源代码

```
n=int(input())
a0,b0=map(int,input().split())
c=[]
for _ in range(n):
    c.append(list(map(int,input().split())))
c.sort(key=lambda x:(x[0]*x[1]))
result=0
for i in range(n):
    result=max(result,a0//c[i][1])
    a0*=c[i][0]
print(result)
```

©2002-2022 POJ 京ICP备20010980号-1

2. 学习总结和收获

完蛋了，现在要努力去做简单和中等的题目。争取把基础的掌握。

