

Assignment #B: Dec Mock Exam大雪前一天

Updated 1649 GMT+8 Dec 5, 2024

2024 fall, Compiled by 徐贤天, 工学院

说明:

- 1) 月考: AC1 ()。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:

$dp[i]$ 表示以第 i 天的股价为结尾的区间所能得到的最大利润(虽然没有用 dp), $minimum_price$ 维护前 i 天的最小值

代码:

```
prices = [int(x) for x in input().split()]
n = len(prices)
dp = [0]*n
minimum_price = prices[0]
for i in range(1, n):
    if prices[i] < minimum_price:
        minimum_price = prices[i]
    else:
        dp[i] = prices[i] - minimum_price
print(max(dp))
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
prices = [int(x) for x in input().split()]
n = len(prices)
dp = [0]*n
minimun_price = prices[0]
for i in range(1, n):
    if prices[i] < minimun_price:
        minimun_price = prices[i]
    else:
        dp[i] = prices[i] - minimun_price
print(max(dp))
```

基本信息

#: 47566806
题目: E22548
提交人: 24n2400011033
内存: 9648kB
时间: 40ms
语言: Python3
提交时间: 2024-12-05 15:25:47

M28701: 炸鸡排

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

思路:

greedy的思路好难想,但是想到了代码就好简洁,然而证明又好复杂

代码:

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

代码运行截图 == (至少包含有"Accepted") ==

状态: Accepted

源代码

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

基本信息

#: 47627737
题目: 28701
提交人: 24n2400011033
内存: 3616kB
时间: 21ms
语言: Python3
提交时间: 2024-12-08 16:46:34

M20744: 土豪购物

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

思路:

有两种情况: 取出和不取出, 而取出的情况又与不取出的情况有关, 于是设置两个dp。dp1[i]和dp2[i]均表示最后一个物品为i时可以取出的最大价值。dp2的状态转移方程有两个分类条件: 如果取出prices[i], 则相当于最大价值为以i-1物品为结尾时的连续物品最大价值, 即dp1[i-1]; 如果不取出prices[i], 则又可以分为两种情况: 其一为加入前面dp2[i-1], 一种是单独成一段。

代码:

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

代码运行截图 (至少包含有"Accepted")

#47630061提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

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

基本信息

#: 47630061
题目: 20744
提交人: 24n2400011033
内存: 9476kB
时间: 69ms
语言: Python3
提交时间: 2024-12-08 18:36:25

©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:

用dfs探索每一条可能的物品选择路径, 主要是细节上需要小心

代码:

```
n, m = map(int, input().split())
prices = [input().split() for _ in range(n)]
coupons = [input().split() for _ in range(m)]
```

```

ans = float('inf')
store_prices = [0] * m

def dfs(item):
    global ans
    if item == n:
        total_coupon = 0
        for i in range(m):
            store_coupon = 0
            store_price = store_prices[i]
            for coupon in coupons[i]:
                a, b = map(int, coupon.split('-'))
                if store_price >= a:
                    store_coupon = max(store_coupon, b)
            total_coupon += store_coupon
        total_price = sum(store_prices) - total_coupon - sum(store_prices) // 300
        * 50
        ans = min(ans, total_price)
        return
    for price in prices[item]:
        idx, p = map(int, price.split(':'))
        store_prices[idx-1] += p
        dfs(item+1)
        store_prices[idx-1] -= p

dfs(0)
print(ans)

```

代码运行截图 (至少包含有"Accepted")

#47633962提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

n, m = map(int, input().split())
prices = [input().split() for _ in range(n)]
coupons = [input().split() for _ in range(m)]
ans = float('inf')
store_prices = [0] * m

def dfs(item):
    global ans
    if item == n:
        total_coupon = 0
        for i in range(m):
            store_coupon = 0
            store_price = store_prices[i]
            for coupon in coupons[i]:
                a, b = map(int, coupon.split('-'))
                if store_price >= a:
                    store_coupon = max(store_coupon, b)
            total_coupon += store_coupon
        total_price = sum(store_prices) - total_coupon - sum(store_prices) // 300
        ans = min(ans, total_price)
        return
    for price in prices[item]:
        idx, p = map(int, price.split(':'))
        store_prices[idx-1] += p
        dfs(item+1)
        store_prices[idx-1] -= p

dfs(0)
print(ans)

```

基本信息

#: 47633962
 题目: 25561
 提交人: 24n2400011033
 内存: 3664kB
 时间: 65ms
 语言: Python3
 提交时间: 2024-12-08 21:58:05

T20741: 两座孤岛最短距离

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

思路:

考场上想到了用dfs找其中一个岛再从这个岛上的每个点开始bfs, 但尝试了很久都不能把自己的想法变成代码流畅地写出来。

这次学到了很多思路, 比如直接把找到的点加到队列q里面, 直接从这些点开始一起bfs可以节省很多时间; 把一段代码写到函数里面去直接return就不会有双重循环外层无法结束的烦恼了。还有就是不能死套模板, 一些灵活的变化需要注意。

代码:

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

def bfs():
    inq = set(q)
    step = 0
    while q:
        for _ in range(len(q)):
            x, y = q.popleft()
            for dx, dy in directions:
                nx, ny = x + dx, y + dy
                if 0 <= nx < n and 0 <= ny < n and (nx, ny) not in inq:
                    if matrix[nx][ny] == 0:
                        q.append((nx, ny))
                        inq.add((nx, ny))
                    elif matrix[nx][ny] == 1:
                        return step
            step += 1

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

def main():
    for i in range(n):
        for j in range(n):
            if matrix[i][j] == 1:
                dfs(i, j)
    return bfs()

print(main())
```

代码运行截图 (至少包含有"Accepted")

#47626512提交状态

查看 提交 统计 提问

状态: Accepted

源代码

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

def bfs():
    inq = set(q)
    step = 0
    while q:
        for _ in range(len(q)):
            x, y = q.popleft()
            for dx, dy in directions:
                nx, ny = x + dx, y + dy
                if 0 <= nx < n and 0 <= ny < n and (nx, ny) not in inq:
                    if matrix[nx][ny] == 0:
                        q.append((nx, ny))
                        inq.add((nx, ny))
                    elif matrix[nx][ny] == 1:
                        return step
            step += 1

n = int(input())
matrix = [list(map(int, input().split())) for _ in range(n)]
q = deque()

def main():
    for i in range(n):
        for j in range(n):
            if matrix[i][j] == 1:
                dfs(i, j)
    return bfs()
```

基本信息

#: 47626512
题目: 20741
提交人: 24n2400011033
内存: 4896kB
时间: 32ms
语言: Python3
提交时间: 2024-12-08 16:04:22

T28776: 国王游戏

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

思路:

直接看很难想到思路, 数学的计算很有必要

代码:

```
n = int(input())
a, b = map(int, input().split())
nums = [[int(x) for x in input().split()] for _ in range(n)]
nums.sort(key = lambda x: (x[0] * x[1]))
ans = 0
for i in range(n):
    ans = max(ans, a // nums[i][1])
    a *= nums[i][0]
print(ans)
```

#47630857提交状态

查看提交统计提问

状态: Accepted

源代码

```
n = int(input())
a, b = map(int, input().split())
nums = [[int(x) for x in input().split()] for _ in range(n)]
nums.sort(key = lambda x: (x[0] * x[1]))
ans = 0
for i in range(n):
    ans = max(ans, a // nums[i][1])
    a *= nums[i][0]
print(ans)
```

基本信息

#: 47630857

题目: 28776

提交人: 24n2400011033

内存: 3620kB

时间: 21ms

语言: Python3

提交时间: 2024-12-08 19:24:05

©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。

这次月考虽然已经做好了被薄纱的准备，但在考场上还是深深感到无力（

“土豪购物”考试的时候第一遍先是tle，想着用空间换时间结果mle，结果是不得不放弃这道题；看到孤岛问题通过率挺高就想着做一下，结果一直在犹豫两次搜索会不会超时，不敢轻易写代码，又在“把想法变成现实”上磕磕绊绊。

这次考试中我也学到了很多，比如土豪购物让我加深了dp的理解，搜索的题目要求更加灵活地运用dfs和bfs，而贪心题还需要再多想想。

最近要准备cheatsheet，希望有所收获。