

Assignment #B: Dec Mock Exam大雪前一天

Updated 1649 GMT+8 Dec 5, 2024

2024 fall, Compiled by 刘家亦, 物院

说明:

- 1) 月考: AC1 (请改为同学的通过数)。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: 最小堆

代码:

```
import heapq
a = list(map(int, input().split()))
q = []
ans = 0
for i in range(len(a)):
    heapq.heappush(q, a[i])
    Min = heapq.heappop(q)
    ans = max(ans, a[i] - Min)
    heapq.heappush(q, Min)
print(ans)
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
import heapq
a = list(map(int, input().split()))
q = []
ans = 0
for i in range(len(a)):
    heapq.heappush(q, a[i])
    Min = heapq.heappop(q)
    ans = max(ans, a[i] - Min)
    heapq.heappush(q, Min)
print(ans)
```

基本信息

#: 47566411
题目: E22548
提交人: 24n2400011431|沧海月明
内存: 9576kB
时间: 78ms
语言: Python3
提交时间: 2024-12-05 15:13:31

M28701: 炸鸡排

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

思路：一个简单的贪心，但是考试的时候太慌了，没有静下来去想算法的实现细节，导致WA

代码：

```
n, k = map(int, input().split())
a = sorted(list(map(int, input().split()))), reverse=True)
t = sum(a[k:])
idx = k - 1
h = a[k - 1]
while t > 0 and idx > 0:
    if t >= (k - idx) * (a[idx - 1] - h):
        t -= (k - idx) * (a[idx - 1] - h)
        idx -= 1
        h = a[idx]
    else:
        h += t / (k - idx)
        t = 0
if idx == 0 and t != 0:
    h += t / k
print(f'{h:.3f}')
```

代码运行截图 == (至少包含有"Accepted") ==

状态: Accepted

源代码

```
n, k = map(int, input().split())
a = sorted(list(map(int, input().split())), reverse=True)
t = sum(a[k:])
idx = k - 1
h = a[k - 1]
while t > 0 and idx > 0:
    if t >= (k - idx) * (a[idx - 1] - h):
        t -= (k - idx) * (a[idx - 1] - h)
        idx -= 1
        h = a[idx]
    else:
        h += t / (k - idx)
        t = 0
if idx == 0 and t != 0:
    h += t / k
print(f'{h:.3f}')
```

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

基本信息

#: 47577620

题目: 28701

提交人: 24n2400011431|沧海月明

内存: 3656kB

时间: 20ms

语言: Python3

提交时间: 2024-12-05 21:08:17

M20744: 土豪购物

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

思路: dp, 和flowers差不多, 在考试的时候已经想到应该先去研究不能放回的情况, 然后根据那个确定答案。但是同样是因为没有认真去想dp, 而是用野路子去做, 导致WA。

代码:

```
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 (至少包含有"Accepted")

状态: Accepted

源代码

```
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))
```

基本信息

#: 47607137
题目: 20744
提交人: 24n2400011431|沧海月明
内存: 9436kB
时间: 67ms
语言: Python3
提交时间: 2024-12-07 15:11:21

©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：考试的时候没有来得及看。原本想用dp，但是由于状态方程不好列，所以改用dfs暴力。

最开始这道题用了lru_cache，但是其实不需要，这道题的核心也不是dp，没有lru_cache优化的空间

代码：

```
def find_the_discount(shop, price):
    res = 0
    for i in range(len(discount[shop]) - 1, -1, -1):
        if price >= discount[shop][i][0]:
            res = max(res, discount[shop][i][1])
    return res

def dfs(costs, idx):
    ans = float('inf')
    if idx == n:
        ans = sum(costs) - (sum(costs) // 300) * 50
        for shop, cost in enumerate(costs):
            ans -= find_the_discount(shop, cost)
        return ans
    cost_list = [costi for costi in costs]
    for shop, price in goods[idx]:
        cost_list[shop] += price
        ans = min(ans, dfs(tuple(cost_list), idx + 1))
        cost_list[shop] -= price
    return ans

n, m = map(int, input().split())
goods = []
for _ in range(n):
    goods.append(list(map(lambda x: [int(x.split(':')[0]) - 1, int(x.split(':')[1])], input().split(' '))))
discount = []
for _ in range(m):
    discount.append(list(map(lambda x: [int(x.split('-')[0]), int(x.split('-')[1])], input().split(' '))))
cost = tuple([0] * m)
```

```
print(dfs(cost, 0))
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
from functools import lru_cache
def find_the_discount(shop, price):
    res = 0
    for i in range(len(discount[shop]) - 1, -1, -1):
        if price >= discount[shop][i][0]:
            res = max(res, discount[shop][i][1])
    return res

def dfs(costs, idx):
    ans = float('inf')
    if idx == n:
        ans = sum(costs) - (sum(costs) // 300) * 50
        for shop, cost in enumerate(costs):
            ans -= find_the_discount(shop, cost)
        return ans
    cost_list = [costi for costi in costs]
    for shop, price in goods[idx]:
        cost_list[shop] += price
        ans = min(ans, dfs(tuple(cost_list), idx + 1))
        cost_list[shop] -= price
    return ans

n, m = map(int, input().split())
goods = []
for _ in range(n):
    goods.append(list(map(lambda x: [int(x.split(':')[0]) - 1, int(x.split(':')[1])], input().split(':'))))
discount = []
for _ in range(m):
    discount.append(list(map(lambda x: [int(x.split('-')[0]), int(x.split('-')[1])], input().split('-'))))
cost = tuple([0] * m)
print(dfs(cost, 0))
```

基本信息

#: 47604265
题目: 25561
提交人: 24n2400011431|沧海月明
内存: 3716kB
时间: 43ms
语言: Python3
提交时间: 2024-12-07 13:11:56

T20741: 两座孤岛最短距离

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

思路: 难绷的题目, 思路很简单, 有一个优化没有想到, 就是在bfs的时候要先标记, 这也是本题所需要的唯一一个优化, 只要有这个优化, 怎么都可以过。纯纯的基本功不过关

代码:

```
from collections import deque
def bfs(x, y, matrix):
    begin = [[x, y, 0]]
    queue = deque([begin])
    while queue:
        x, y = queue.popleft()
        for dx, dy in [(1, 0), (-1, 0), (0, 1), (0, -1)]:
            nx, ny = x + dx, y + dy
            if 0 <= nx < len(matrix) and 0 <= ny < len(matrix[0]) \
                and matrix[nx][ny] == '1':
                queue.appendleft([nx, ny])
                begin.append([nx, ny, 0])
                matrix[nx][ny] = '2'
```

```

return begin

def search(begin):
    queue = deque(begin)
    while queue:
        x, y, depth = queue.pop()
        for dx, dy in [(1, 0), (-1, 0), (0, 1), (0, -1)]:
            nx, ny = x + dx, y + dy
            if 0 <= nx < len(matrix) and 0 <= ny < len(matrix[0]) \
                and matrix[nx][ny] == '1':
                return depth
            if 0 <= nx < len(matrix) and 0 <= ny < len(matrix[0]) \
                and matrix[nx][ny] == '0':
                matrix[nx][ny] = '2'
                queue.appendleft([nx, ny, depth + 1])

n = int(input())
matrix = [list(input()) for _ in range(n)]
for i in range(n):
    for j in range(n):
        if matrix[i][j] == '1':
            matrix[i][j] = '2'
            begin = bfs(i, j, matrix)
            print(search(begin))
            exit(0)

```

代码运行截图 (至少包含有"Accepted")

T28776: 国王游戏

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

思路：很容易想到的贪心，但是证明起来还是有点麻烦，可惜考试的时候没有时间做了

代码：

```

n = int(input())
p = []
total, useless = map(int, input().split())
for i in range(n):
    a, b = map(int, input().split())
    p.append(a * b)
    total *= a
p.sort()
print(total // p[-1])

```

代码运行截图 (至少包含有"Accepted")

状态: **Accepted**

源代码

```
n = int(input())
p = []
total, useless = map(int, input().split())
for i in range(n):
    a, b = map(int, input().split())
    p.append(a * b)
    total *= a
p.sort()
print(total // p[-1])
```

基本信息

#: 47596572
题目: 28776
提交人: 24n2400011431|沧海月明
内存: 3608kB
时间: 21ms
语言: Python3
提交时间: 2024-12-06 21:50:23

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。

自己的这次考试大挫特挫。究其原因还是最近疏忽了对计概的练习和思考。做题的时候都是直接套模版，没有去想代码是怎么运行的。再加上考试时的时候很慌，很多题目都没有想实现的细节，导致错误，然后还一直debug，浪费时间。