Assignment #B: Dec Mock Exam大雪前一天

Updated 1649 GMT+8 Dec 5, 2024

2024 fall, Complied by 佟永鑫 元培学院

说明：

1）月考： AC3 。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

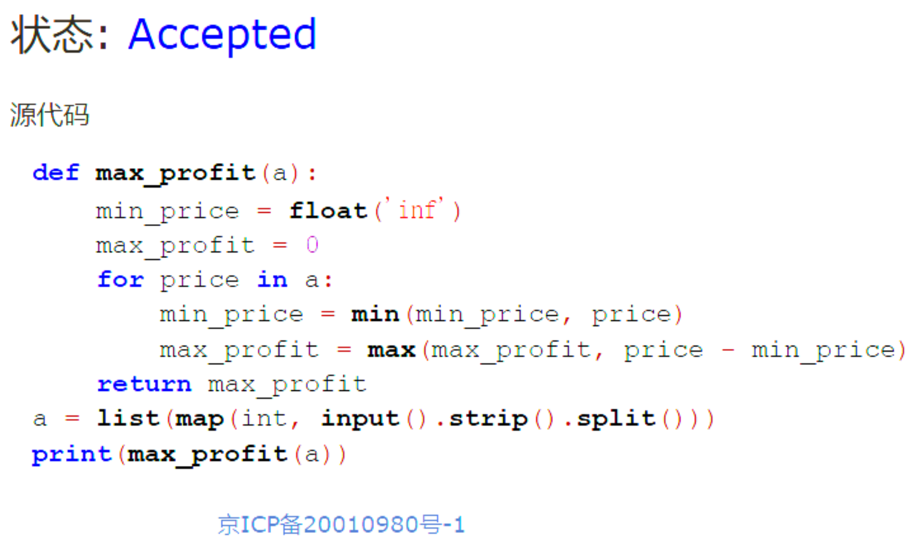
http://cs101.openjudge.cn/practice/22548/

思路：

代码：

```python
def max_profit(a):
    min_price = float('inf')
    max_profit = 0
    for price in a:
        min_price = min(min_price, price)
        max_profit = max(max_profit, price - min_price)
    return max_profit
a = list(map(int, input().strip().split()))
print(max_profit(a))
```

代码运行截图 （至少包含有"Accepted"）



状态: Accepted

源代码

```
def max_profit(a):
    min_price = float('inf')
    max_profit = 0
    for price in a:
        min_price = min(min_price, price)
        max_profit = max(max_profit, price - min_price)
    return max_profit
a = list(map(int, input().strip().split()))
print(max_profit(a))
```

京ICP备20010980号-1

M28701: 炸鸡排

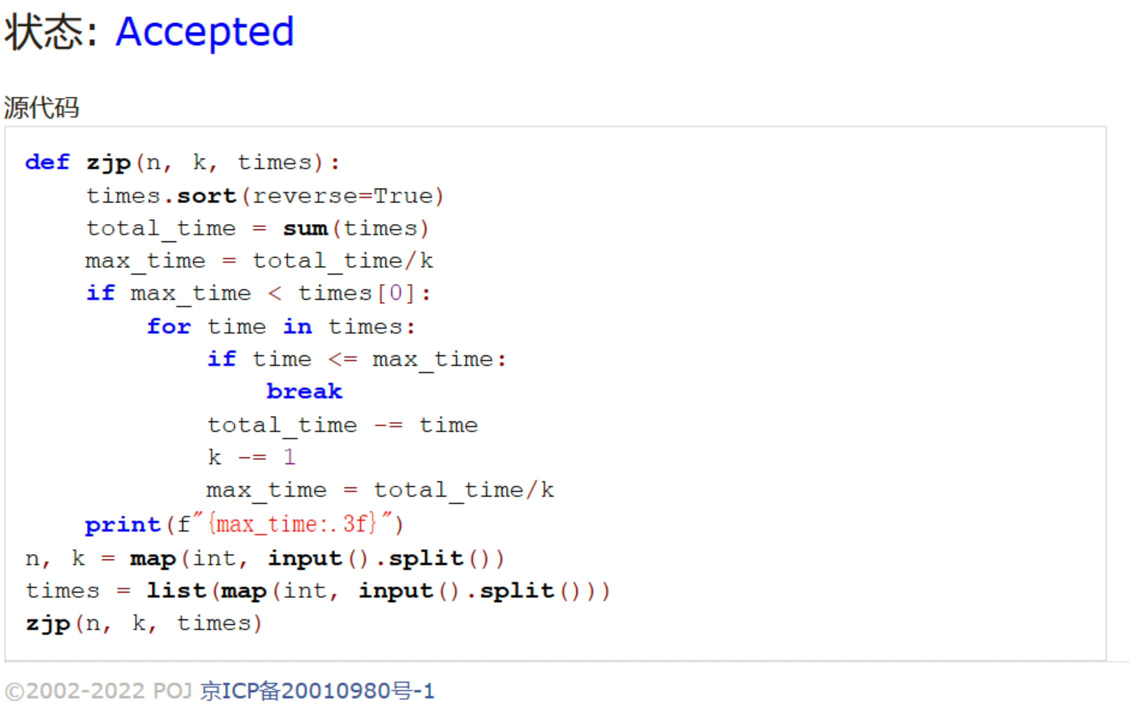
greedy, http://cs101.openjudge.cn/practice/28701/

思路:

代码:

```
def zjp(n, k, times):
    times.sort(reverse=True)
    total_time = sum(times)
    max_time = total_time/k
    if max_time < times[0]:
        for time in times:
            if time <= max_time:
                break
            total_time -= time
            k -= 1
            max_time = total_time/k
    print(f"{max_time:.3f}")
n, k = map(int, input().split())
times = list(map(int, input().split()))
zjp(n, k, times)
```

代码运行截图 ==（至少包含有"Accepted"） ==



状态: Accepted

源代码

```
def zjp(n, k, times):
    times.sort(reverse=True)
    total_time = sum(times)
    max_time = total_time/k
    if max_time < times[0]:
        for time in times:
            if time <= max_time:
                break
            total_time -= time
            k -= 1
            max_time = total_time/k
    print(f"{max_time:.3f}")
n, k = map(int, input().split())
times = list(map(int, input().split()))
zjp(n, k, times)
```

©2002-2022 POJ 京ICP备20010980号-1

M20744: 土豪购物

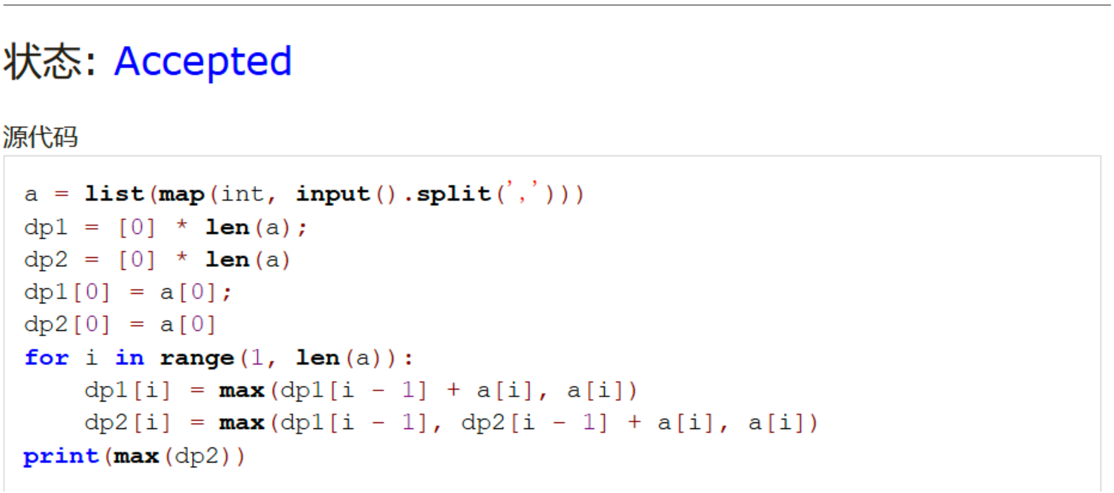
dp, http://cs101.openjudge.cn/practice/20744/

思路:

代码:

```
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))
```

T25561: 2022决战双十一

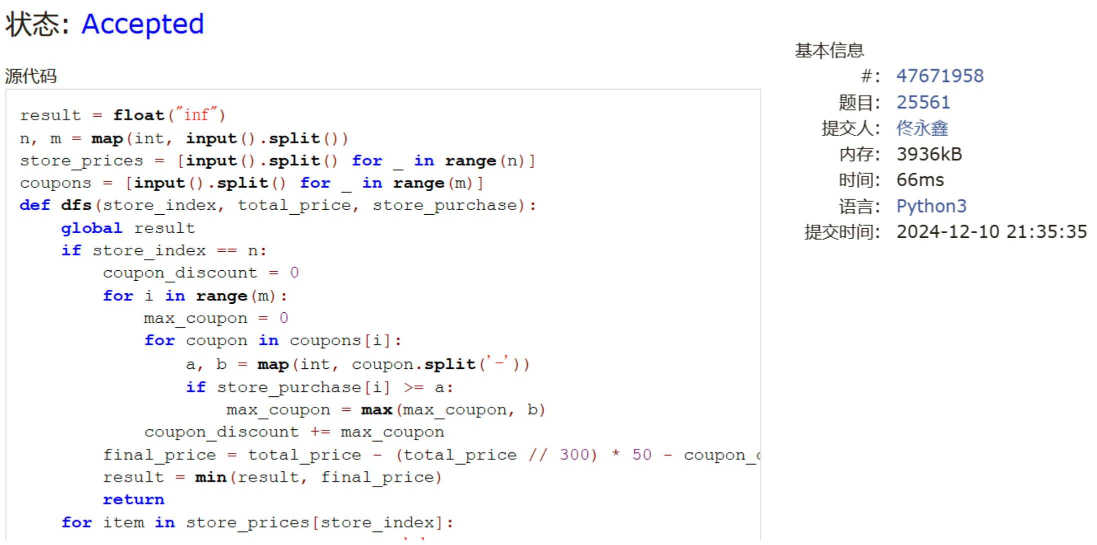
brute force, dfs, http://cs101.openjudge.cn/practice/25561/

思路:

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_index, total_price, store_purchase):
    global result
    if store_index == n:
        coupon_discount = 0
        for i in range(m):
            max_coupon = 0
            for coupon in coupons[i]:
                a, b = map(int, coupon.split('-'))
                if store_purchase[i] >= a:
                    max_coupon = max(max_coupon, b)
            coupon_discount += max_coupon
        final_price = total_price - (total_price // 300) * 50 - coupon_discount
        result = min(result, final_price)
        return
    for item in store_prices[store_index]:
        idx, p = map(int, item.split(':'))
        store_purchase[idx - 1] += p
        dfs(store_index + 1, total_price + p, store_purchase)
        store_purchase[idx - 1] -= p
dfs(0, 0, [0] * m)
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_index, total_price, store_purchase):
    global result
    if store_index == n:
        coupon_discount = 0
        for i in range(m):
            max_coupon = 0
            for coupon in coupons[i]:
                a, b = map(int, coupon.split('-'))
                if store_purchase[i] >= a:
                    max_coupon = max(max_coupon, b)
            coupon_discount += max_coupon
        final_price = total_price - (total_price // 300) * 50 - coupon_c
        result = min(result, final_price)
        return
    for item in store_prices[store_index]:
```

基本信息

#:	47671958
题目:	25561
提交人:	佟永鑫
内存:	3936kB
时间:	66ms
语言:	Python3
提交时间:	2024-12-10 21:35:35

T20741: 两座孤岛最短距离

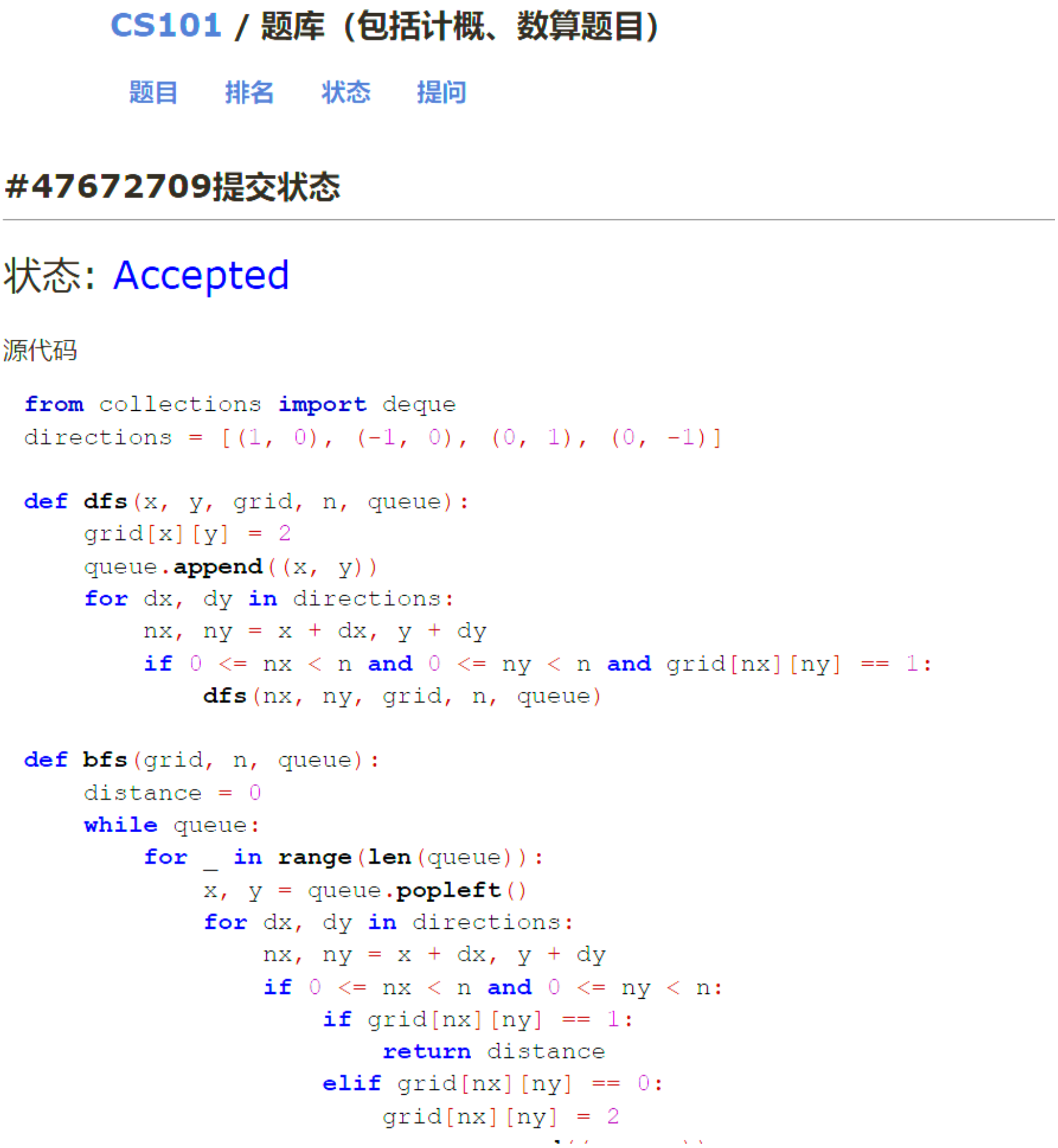
dfs, bfs, http://cs101.openjudge.cn/practice/20741/

思路:

代码:

```python
from collections import deque
directions = [(1, 0), (-1, 0), (0, 1), (0, -1)]

def dfs(x, y, grid, n, queue):
    grid[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 grid[nx][ny] == 1:
            dfs(nx, ny, grid, n, queue)

def bfs(grid, n, queue):
    distance = 0
    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 grid[nx][ny] == 1:
                        return distance
                    elif grid[nx][ny] == 0:
                        grid[nx][ny] = 2
                        queue.append((nx, ny))
        distance += 1
    return distance

def main():
    n = int(input())
    grid = [list(map(int, input().strip())) for _ in range(n)]
    queue = deque()
    for i in range(n):
        for j in range(n):
            if grid[i][j] == 1:
                dfs(i, j, grid, n, queue)
                return bfs(grid, n, queue)

if __name__ == "__main__":
    print(main())
```

代码运行截图 （至少包含有"Accepted"）



CS101 / 题库（包括计概、数算题目）

#47672709提交状态

状态: Accepted

源代码

```
from collections import deque
directions = [(1, 0), (-1, 0), (0, 1), (0, -1)]

def dfs(x, y, grid, n, queue):
    grid[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 grid[nx][ny] == 1:
            dfs(nx, ny, grid, n, queue)

def bfs(grid, n, queue):
    distance = 0
    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 grid[nx][ny] == 1:
                        return distance
                    elif grid[nx][ny] == 0:
                        grid[nx][ny] = 2
```

T28776: 国王游戏

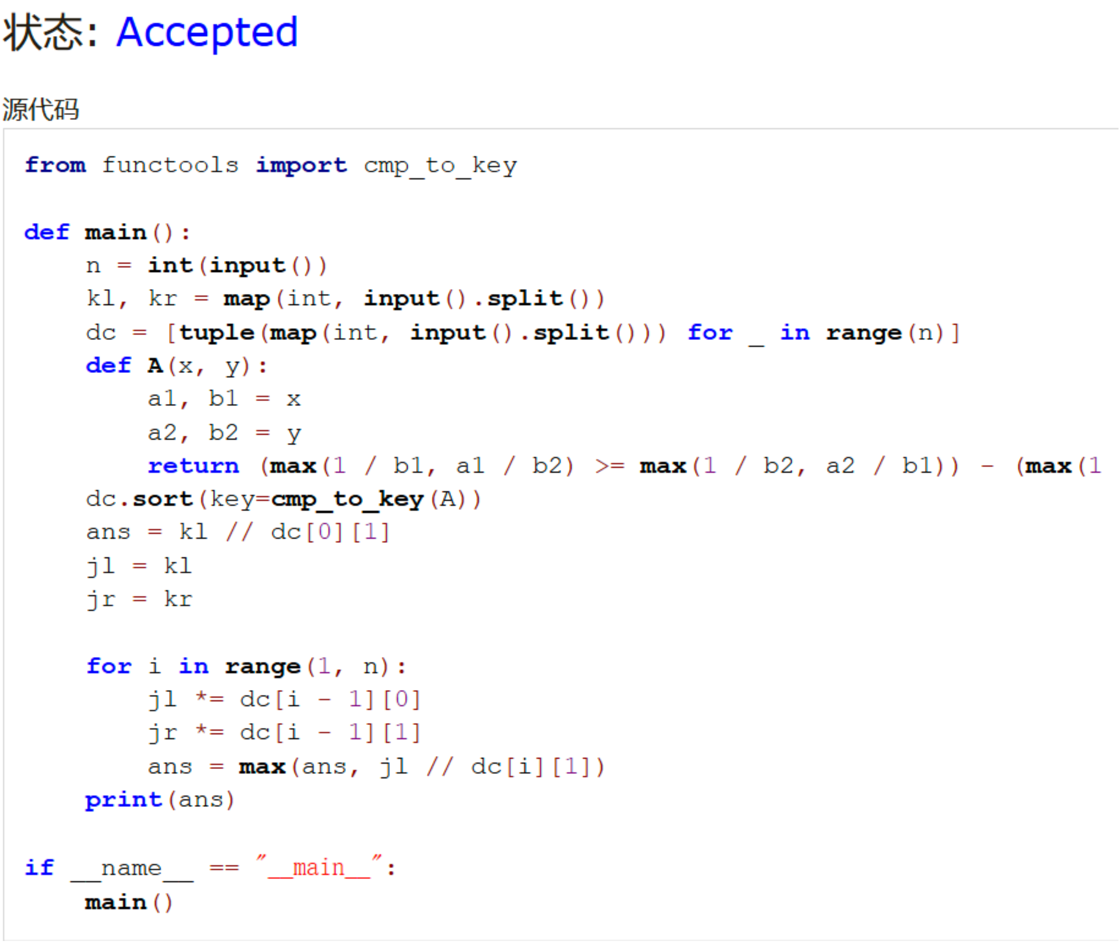
greedy, http://cs101.openjudge.cn/practice/28776

思路：

代码：

```python
from functools import cmp_to_key

def main():
    n = int(input())
    kl, kr = map(int, input().split())
    dc = [tuple(map(int, input().split())) for _ in range(n)]
    def A(x, y):
        a1, b1 = x
        a2, b2 = y
        return (max(1 / b1, a1 / b2) >= max(1 / b2, a2 / b1)) - (max(1 / b1, a1 / b2) < max(1 / b2, a2 / b1))
    dc.sort(key=cmp_to_key(A))
    ans = kl // dc[0][1]
    jl = kl
    jr = kr

    for i in range(1, n):
        jl *= dc[i - 1][0]
        jr *= dc[i - 1][1]
        ans = max(ans, jl // dc[i][1])
    print(ans)

if __name__ == "__main__":
    main()
```

代码运行截图 （至少包含有"Accepted"）



状态: Accepted

源代码

```
from functools import cmp_to_key

def main():
    n = int(input())
    kl, kr = map(int, input().split())
    dc = [tuple(map(int, input().split())) for _ in range(n)]
    def A(x, y):
        a1, b1 = x
        a2, b2 = y
        return (max(1 / b1, a1 / b2) >= max(1 / b2, a2 / b1)) - (max(1
    dc.sort(key=cmp_to_key(A))
    ans = kl // dc[0][1]
    jl = kl
    jr = kr

    for i in range(1, n):
        jl *= dc[i - 1][0]
        jr *= dc[i - 1][1]
        ans = max(ans, jl // dc[i][1])
    print(ans)

if __name__ == "__main__":
    main()
```

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。

股民老张、土豪购物、孤岛距离能AC，炸鸡排感觉出了思路但觉得不太能证明就没写，双十一没什么思路，国王游戏没时间看了。