

B3928 [GESP202312 四级] 田忌赛马

一句话思路

双方都从最慢的马开始排队。

能赢就上,赢不了就把这匹"废马"扔掉,继续用下一匹更快的马挑战田忌的慢马。

题目大意

你和田忌各有 N 匹马,速度已知。你可以任意安排出场顺序,问最多能赢几场。

思路分析

1. 为什么要排序?

让两队的马都从慢到快站好队,之后我们就能"按顺序对战",一眼看出谁打得过谁。

2. 贪心策略

我们盯着田忌最慢的那匹马,然后从自己的马里挑出最弱但能赢它的那一匹去打。

- 这样赢了一场,而且把"刚刚够用"的马用掉了,好马留着对付田忌后面的强马。
- 如果我最慢的马连田忌最慢的马都打不过,那它对谁都赢不了——直接放弃这匹马,换更快的一匹再来挑战田忌这匹慢马。

3. 双指针怎么动?

用 i 指自己的马, j 指田忌的马,都从 0 开始(最慢的)。

情况	含义	怎么做
$u[i] > v[j]$	我能赢这场	赢一场, $i++$, $j++$, $ans++$
$u[i] \leq v[j]$	我这匹太弱了	弃用,只 $i++$

当任一队的指针走到尽头,比赛结束。

C++ 参考代码

```

#include <iostream>
#include <algorithm>
using namespace std;

int u[50005], v[50005];

int main() {
    int n;
    cin >> n;
    for (int i = 0; i < n; i++) cin >> u[i];
    for (int i = 0; i < n; i++) cin >> v[i];

    sort(u, u + n); // 我的马从慢到快
    sort(v, v + n); // 田忌的马从慢到快

    int i = 0, j = 0, ans = 0;
    while (i < n && j < n) {
        if (u[i] > v[j]) { // 能赢:派出去,两边各换下一匹
            ans++;
            i++;
            j++;
        } else { // 赢不了:这匹太废,扔掉
            i++;
        }
    }

    cout << ans << endl;
    return 0;
}

```

小贴士

- 别忘了排序,否则贪心完全失效。
- "赢"是严格大于,题目里平局不算赢。