**Semifinals**

**Link submit:** <http://codeforces.com/problemset/problem/378/B>

**Solution:**

|  |  |
| --- | --- |
| C++ | <https://ideone.com/21xcLU> |
| Java | <https://ideone.com/Dkq77j> |
| Python | <https://ideone.com/THnfiO> |

**Tóm tắt đề:**

Cho 2n vận động viên nằm ở hai bảng, mỗi bảng gồm n vận động viên.

Mỗi vận động viên được đại diện bởi một con số là thành tích của họ sau khi chạy ở vòng bán kết (số càng nhỏ xếp hạng càng cao).

Biết rằng ban tổ chức sẽ chọn n vận động viên vào chung kết theo quy tắc như sau:

* Chọn một con số k bất kỳ (0 ≤ 2k ≤ n).
* k thành viên có thành tích tốt nhất ở bảng 1 và k thành viên có thành tích tốt nhất ở bảng 2 sẽ ưu tiên lọt vào chung kết. Nếu chưa đủ n vận động viên thì chọn tiếp n – 2k vận động viên có thành tích tốt nhất tính chung của hai bảng để vào vòng trong (không tính những người đã được chọn trước đó).

Nhiệm vụ của bạn là tìm ra tất cả các vận động viên **có khả năng** lọt vòng chung kết.

**Input:**

Dòng đầu tiên chứa một số nguyên *n* (1 ≤ n ≤ 105) -- số lượng vận động viên ở mỗi bảng.

n dòng tiếp theo, mỗi dòng gồm hai số nguyên *ai và* *bi* (1 ≤ ai, bi ≤ 109) lần lượt là kết quả thi đấu của vận động viên thứ i ở bảng 1 và bảng 2. Dữ liệu đảm bảo hai mảng a, b được sắp xếp tăng dần và các kết quả đều phân biệt.

**Output:**

In ra trên hai dòng lần lượt kết quả của các vận động viên ở bảng 1 và bảng 2.

Trong đó, mỗi dòng gồm n kí tự “0” nếu vận động viên đó không có cơ hội vào chung kết hoặc “1” nếu ngược lại.

**Ví dụ:**

|  |  |
| --- | --- |
| 4  9840 9920  9860 9980  9930 10020  10040 10090 | 1110  1100 |

|  |  |
| --- | --- |
| 4  9900 9850  9940 9930  10000 10020  10060 10110 | 1100  1100 |

**Giải thích ví dụ:**

***Ví dụ 1:*** Ở mỗi bảng gồm 4 vận động viên với kết quả là {9840, 9860, 9930, 10040} và {9920, 9980, 10020, 10090}.

Ban tổ chức có thể chọn bất cứ con số k nào trong khoảng [0, 2]:

* Nếu k = 0, không ai được ưu tiên, 4 người có kết quả tốt nhất được chọn là 9840, 9860, 9920 và 9930.
* Nếu k = 1, người nhất của mỗi bảng (9840 và 9920) được ưu tiên, trong 6 người còn lại chọn ra thêm 4 - 2 \* 1 = 2 người tốt nhất thì được 9860 và 9930.
* Nếu k = 2, hai người tốt nhất của mỗi bảng (bảng 1 là 9840 và 9860, bảng 2 là 9920 và 9980) được chọn. Vì đã đủ số lượng vào chung kết nên không ai được chọn thêm.

Như vậy, ta thấy trong tất cả các trường hợp thì 9840, 9860 và 9930 ở bảng 1 có khả năng được chọn vào chung kết. Ở bảng 2 là 9920 và 9980 có khả năng được chọn vào chung kết. Do đó kết quả bảng 1 sẽ là "1110" và bảng 2 sẽ là "1100".

***Ví dụ 2:*** Làm tương tự như ví dụ 1.

**Hướng dẫn giải:**

Nhận xét: Ta thấy rằng các vận động viên (VĐV) có khả năng được chọn vào vòng trong đều được bao quát bởi một trong hai trường hợp sau:

* Trường hợp 1: Chỉ chọn các VĐV được ưu tiên, tức k = n / 2 VĐV tốt nhất ở mỗi bảng.
* Trường hợp 2: Không ưu tiên chọn bất cứ VĐV nào, tức k = 0. Lúc này n VĐV vào vòng trong sẽ được chọn từ n VĐV có thành tích tốt nhất tính chung của cả hai bảng.

Như vậy, ta có các bước để giải bài toán trên như sau:

* Bước 1: Đưa kết quả chạy của các VĐV ở từng bảng vào mảng động tương ứng.
* Bước 2: Tạo hai mảng đánh dấu ở hai bảng với nhiệm vụ đánh dấu lại những VĐV có khả năng lọt vào chung kết nếu không có bất cứ sự ưu tiên nào (tức k = 0).
* Bước 3: Xử lý trường hợp 2 đã nêu trên bằng cách sử dụng một biến đếm số lượng VĐV đã chọn được và hai biến chạy i, j trên mỗi bảng. Trong trường hợp ta vẫn chưa đánh dấu đủ n VĐV:
  + So sánh kết quả VĐV thứ i của bảng 1 và VĐV thứ j của bảng 2 và chọn người có kết quả tốt hơn.
  + Tăng biến chạy i (nếu VĐV đó thuộc bảng 1) hoặc j (nếu VĐV đó thuộc bảng 2).
  + Tăng số lượng VĐV đã chọn được.
* Bước 4: Gộp chung cả hai trường hợp đã nêu bằng cách duyệt lại lần nữa VĐV trên từng bảng:
  + Nếu VĐV đó nằm trong số n / 2 VĐV đầu tiên của bảng đó hoặc đã được đánh dấu trong bước 2 thì in "1".
  + Ngược lại, in "0".

**Độ phức tạp:** **O(n)** với n là số lượng vận động viên ở mỗi bảng.