Running routes

locked

Problem Submissions Leaderboard Discussions

Your friend Rob is a member of the Spilamberto Runners Club, a top-notch team of runners and race walkers, Italy's pride and joy.

Runners work out by travelling routes from a start marker to an end marker. For simplicity, we assume that such markers are just numbers liying on a secret map. For example, Rob walks from 1 to 5, Giuly runs from 3 to 6, and Mark walks from 10 to 32.

Every day, runners indicate their daily route by filling a registry. However, at the end of the day, the result is a mess! As you can imagine, some routes *might* overlap. Back to the example above, Rob and July *together* travel from 1 to 6.

Rob needs your help with tidying up the registry. He wants you to merge all such overlapping routes and print them ordered by smaller starting markers. Your program will run every month so expect a non-trivial input size. The example above should be printed as:

```
1 6
10 32
```

Input Format

The first line contains the number of routes N.

Then, N lines follow, each containing a route as $start_i$ end_i

Constraints

- $1 \leq N \leq 10^5$
- $1 \leq start_i \leq end_i \leq 10^4$

Output Format

The result of merging overlapping routes, that is the (non-overlapping) routes that cover all the routes in the input, in ascending order. The routes must be printed as lines of pairs $start_i$ end_i (as the input).

Sample Input 0

```
4
1 3
8 10
2 6
15 18
```

Sample Output 0

```
1 6
8 10
15 18
```

Explanation 0

Only the first and the third routes can be merged.

Sample Input 1

```
2
```

```
1 4
4 5
```

Sample Output 1

```
1 5
```

Explanation 1

Both the routes overlap and can be merged.

Submissions: 11
Max Score: 30
Difficulty: Medium

Rate This Challenge:

Download problem statement

Download all test cases

Suggest Edits

Collapse

C++

```cpp
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;


int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code