Hello, **2024101067**.

[DSA] CPU Job Scheduling

Submit solution

All submissions
Best submissions

Points: 100 (partial)
Time limit: 0.5s
Memory limit: 256M

Allowed languages
C

CPU Job Scheduling

Problem Statement

Process scheduling algorithms are used by a CPU to schedule the running processes optimally. A core can execute one process at a time, but a CPU may have multiple cores.

There are N jobs where the i^{th} job starts its execution at $\texttt{start}[i]$ and ends at $\texttt{end}[i]$, both inclusive.

When there is only one core, the strategy to schedule the maximum number of jobs is the following:

- when the core gets free, pick a job from the unfinished set with the least end time (and start time greater than the current time), and schedule it on that core.

Note that given only one core, it may not be possible to schedule all the given N jobs. We may have to use more cores.

Find the minimum number of crores required to execute all the N jobs.

Input

The first line contains the number of jobs $1 \leq N \leq 2 \times 10^5$

The second line contains N-spaced integers, i^{th} integer denoting the start time of the job i.

The third line contains N-spaced integers, i^{th} integer denoting the end time of job i.

$1 \leq \texttt{start}[i] \leq \texttt{end}[i] \leq 10^9$

Output

Sample Testcases

Sample 1

Input:

```
3
1 3 4
3 5 6
```

Copy

Output

```
2
```

Copy

Explanation

Since Job 1 and Job 2 can't run on single core (both are running during time = 3), we need two cores, Job 3 and Job 1 can run on first core, and Job 2 on second core.

Clarifications

Request clarification

No clarifications have been made at this time.