Day 1 Notice

For all tasks:

- There is an attachment package that you can download from the contest system.
- The attachment package contains sample graders, sample implementations, example test cases, and compile scripts.
- You may make up to 50 submissions for each task, and you have to submit exactly one file in each submission.
- The name of the file that you should submit is given in the task statement header.
- It should implement the procedures described in the task statement using the signatures provided in the sample implementations.
- You are free to implement other procedures.
- Your submissions must not read from the standard input, write to the standard output, or interact with any other file. However, they may output to the standard error stream.
- Your submissions **must not call exit() or System.exit().** The verdict of your submission is undefined if you call these functions to end execution prematurely.
- When testing your programs with the sample grader, your input should match the format and constraints from the task statement, otherwise, unspecified behaviors may occur.
- In sample grader inputs, every two consecutive tokens on a line are separated by a single space, unless another format is explicitly specified.
- When you test your code on your local machine, we recommend you to use scripts in the attachment package. Otherwise, especially in C++, make sure to add `-std=gnu++17` option to compile.
- If you are unable to submit to CMS, you can use the `ioisubmit` tool to store your code for evaluation after the end of the contest.
    - Run `ioisubmit <task_shortname> <source_file>` in directory with `<source_file>`.
    - Ask your proctor to take a picture of the output of `ioisubmit` and send it to the organizers. Your submission will not be considered unless this step was done.

Convention

The task statements specify signatures using generic type names `void`, `int`, `int64`, `int[]` (array), and `int[][]` (2D array).

In each of the supported programming languages, the graders use appropriate data types or implementations, as listed below

Language	`void`	`int`	`int64`	`int[]`	length of array `a`
C++	`void`	`int`	`long long`	`std::vector<int>`	`a.size()`
Java	`void`	`int`	`long`	`int[]`	`a.length`

A 2D array is a non-empty array of arrays of the same length.

Language	`int[][]`	#rows in 2D array `a`	#columns in 2D array `a`
C++	`std::vector<std::vector<int>>`	`a.size()`	`a[0].size()`
Java	`int[][]`	`a.length`	`a[0].length`

Limits

Task	Name	Time limit	Memory Limit
plants	Comparing Plants	4.000 seconds	2.00 GiB
supertrees	Connecting Supertrees	1.000 second	2.00 GiB
tickets	Carnival Tickets	2.000 seconds	2.00 GiB