



Notice

For all tasks:

- Time and memory limits are available in the "Overview" page in the contest system.
- There is an attachment package that you can download from the contest system, containing sample graders, sample implementations, example test cases, and compile and run scripts.
- Each task has a subtask with index 0 which is worth 0 points. The test cases for this subtask are the same as the sample test cases in the downloadable attachment.
- You may make up to 50 submissions for each task, and you have to submit exactly one file in each submission.
- When testing a program with the sample grader, your input should match the format and constraints from the task statement, otherwise, unspecified behaviors may occur. 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 the scripts from the attachment packages. Please note that we use the `-std=gnu++17` compiler option.
- 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 the following command from the directory containing `<source_file>` - the file that you want to submit: `ioisubmit <task_shortname> <source_file>`
 - Ask a committee member to take a picture of the output of `ioisubmit`. Your submission will be rejected unless this step was done.
 - Your submission will only be considered if it is accompanied by an appeal from your Team Leader.

Convention

The task statements specify signatures using generic type names `bool`, `int`, `int64`, `int[]` (array), and `int[][]` (array of arrays).

In C++, the graders use appropriate data types or implementations, as listed below

| <code>bool</code> | <code>int</code> | <code>int64</code> | <code>int[]</code> |
|-------------------|------------------|------------------------|-------------------------------------|
| <code>bool</code> | <code>int</code> | <code>long long</code> | <code>std::vector<int></code> |

| <code>int[][]</code> | length of array a |
|--|-----------------------|
| <code>std::vector<std::vector<int>></code> | <code>a.size()</code> |