



注意事项

对于所有题目：

- 比赛系统的“概述”页中可以查到相应的限制。
- 比赛系统中可下载附件包，其中包含评测程序示例、实现示例、测试用例示例、编译和运行脚本。
- 每道题目最多可以提交 50 次，每次只能提交一个文件。
- 在使用评测程序示例来评测你的程序时，输入数据应当符合题面所给定的格式和约束条件，否则可能会导致不确定的结果。
- 除非明确指定了其他格式，否则在评测程序示例的输入中，每一行里连续两项之间用一个空格分隔。
- 在本地机器测试你的代码时，建议使用附件包中的脚本。注意评测系统使用了 `-std=gnu++17` 编译选项。
- 如果出现无法提交到 CMS 的情况，可以使用 `ioisubmit` 工具来保存你的代码，以便在比赛结束后进行评测。
 - 在 `<源代码文件>` 所在的目录下运行 `ioisubmit <题目简称> <源代码文件>`。
 - 请委员会成员对 `ioisubmit` 的输出拍照。如果不这样做，你的提交将不会被处理。
 - 如果以在线方式参赛，请你的监考人对 `ioisubmit` 的输出拍照，并发送给竞赛组织方。

约定

题面在给出函数接口时，会使用一般性的类型名称 `void`、`bool`、`int`、`int[]`（数组）和 `union(bool, int[])`。

在 C++ 中，评测程序会采用适当的数据类型或实现，如下表所示：

<code>void</code>	<code>bool</code>	<code>int</code>	<code>int[]</code>
<code>void</code>	<code>bool</code>	<code>int</code>	<code>std::vector<int></code>

<code>union(bool, int[])</code>	数组 <code>a</code> 的长度
<code>std::variant<bool, std::vector<int>></code>	<code>a.size()</code>

C++ 语言里，`std::variant` 定义在 `<variant>` 头文件中。一个返回类型为 `std::variant<bool, std::vector<int>>` 的函数可以返回一个 `bool` 或一个 `std::vector<int>`。以下示例代码给出了三个返回 `std::variant` 的函数，它们都能正常工作：

```
std::variant<bool, std::vector<int>> foo(int N) {  
    return N % 2 == 0;  
}  
std::variant<bool, std::vector<int>> goo(int N) {  
    return std::vector<int>(N, 0);  
}  
std::variant<bool, std::vector<int>> hoo(int N) {  
    if (N % 2 == 0) {  
        return false;  
    }  
    return std::vector<int>(N, 0);  
}
```