



注意

全ての問題について:

- 制限事項はコンテストシステムの "Overview" ページから見る事ができる。
- 採点プログラムのサンプル, 実行プログラムのサンプル, 入出力例, コンパイルと実行のためのスクリプトを含む添付のパッケージをコンテストシステムからダウンロードすることができる。
- あなたは各課題について, 50 回まで提出することができる。また, それぞれの提出において, ちょうど1つのファイルを提出しなければならない。
- 採点プログラムのサンプルを試すとき, 入力の問題文で示されたフォーマットと制約を満たしていなければならない。そうでない場合, 予期しない挙動をするかもしれない。
- 採点プログラムのサンプルの入力について, 特別な指示がない限り, 各列の連続する2つのトークンは, 1つの半角スペースによって区切る事。
- 手元のコンピュータであなたの書いたプログラムを試すとき, 添付のパッケージに含まれるスクリプトを用いることを推奨する。コンパイルオプション `-std=gnu++17` を使用していることに注意すること。
- もし CMS に提出できなくなった場合, `ioisubmit` コマンドを用いてあなたのコードを記録することができる。記録されたコードはコンテスト終了後に評価される。
 - `<source_file>` があるディレクトリで `ioisubmit <task_shortname> <source_file>` というコマンドを実行すること。
 - 委員会のメンバーに, `ioisubmit` の出力の写真を撮るよう頼むこと。これが行われない限り, あなたの提出は考慮されない。
 - オンラインで競技に参加している場合, 監督者に `ioisubmit` の出力の写真を撮り, 主催者に送るよう頼むこと。

約束事

問題文では, `void`, `int`, `int[]` (配列), `union(bool, int[])` といった一般的な形式で型を表す。

C++ では, 採点プログラムは以下のように適切な型や実装が用いられる。

<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>	配列 <code>a</code> の長さ
<code>std::variant<bool, std::vector<int>></code>	<code>a.size()</code>

C++ では、`std::variant` はヘッダー `<variant>` において定義されている。 `std::variant<bool, std::vector<int>>` を戻り値の型とする関数は、`bool` と `std::vector<int>` のどちらかを返すことができる。 以下に示すサンプルコードは、`std::variant` の使用例である。

```
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);
}
```