



鉱物 (Minerals)

JOI 教授の研究室では、 N 種類の鉱物を研究している。研究室には、それぞれの種類の鉱物の薄い切片が 2 枚ずつ保管されている。切片は合計 $2N$ 枚あり、1 から $2N$ までの番号が付けられている。

しかしある日、助手のビ太郎は $2N$ 枚の切片が入っている箱を落としてしまい、どの切片とどの切片が同じ種類の鉱物のものであるかが分からなくなってしまった。

研究室には、0 枚以上 $2N$ 枚以下の切片を入れると、鉱物が吸収する光の波長を測定することで、何種類の鉱物が入っているかを調べる装置がある。ビ太郎はこの装置を使うことにより、 $2N$ 枚の切片から、同じ種類の鉱物である N 組のペアを特定しようとしている。ビ太郎は、最初は装置に切片が入っていない状態から始めて、以下の操作のいずれかを繰り返す：

- 装置に切片を 1 枚新たに入れて、装置に何種類の鉱物が入っているかを調べる。
- 装置に入っている切片を 1 枚取り出して、装置に何種類の鉱物が入っているかを調べる。

時間をかけて作業を行っていると JOI 教授に見つかってしまうので、ビ太郎は、装置に切片を出し入れする回数を合計 1 000 000 回以内に抑えたい。

鉱物の種類数が与えられたとき、装置に切片を出し入れすることで、同じ種類の鉱物であるペアを特定するプログラムを作成せよ。

実装の詳細

あなたは 1 つのファイルを提出しなければならない。

あなたの提出するファイルは `minerals.cpp` という名前である。このファイルは以下の関数を実装していなければならない。そのプログラムは `minerals.h` をインクルードすること。

- `void Solve(int N)`

この関数は各テストケースにおいて 1 回だけ呼び出される。

- 引数 N は鉱物の種類数 N を表す。

あなたのプログラムは以下の関数を呼び出すことができる。

- ★ `int Query(int x)`

この関数は、切片の番号を指定すると、その切片がすでに装置の中に入っているのであれば装置から取り出し、切片が装置の中に入っていないのであれば装置に切片を入れる。その後、装置に入っている鉱物の種類数を調べ、その種類数を返す。

- 切片の番号を引数 x で指定する。このとき、 $1 \leq x \leq 2N$ が成り立たなければならない。これが満たされない場合、不正解 [1] と判定される。関数 `Query` を 1 000 000 回を超えて呼び



出してはならない。1 000 000 回を超えて呼び出した場合、不正解 [2] と判定される。

★ void Answer(int a, int b)

この関数を用いて、同じ種類の鉱物の切片のペアを回答する。

- ◇ 引数 a, b は、 a 番目の切片と b 番目の切片が同じ種類の鉱物であることを表す。このとき、 $1 \leq a \leq 2N$, $1 \leq b \leq 2N$ が成り立たなければならない。これが満たされない場合、不正解 [3] と判定される。 a または b に同じ値が累計で 2 回以上引数として登場した場合、不正解 [4] と判定される。異なる種類の鉱物が指定された場合、不正解 [5] と判定される。

関数 Answer はちょうど N 回呼び出される必要がある。関数 Solve の実行の終了時に関数 Answer の呼び出し回数が N 回でなかった場合、不正解 [6] と判定される。

重要な注意

- 内部での使用のために他の関数を実装したり、グローバル変数を宣言するのは自由である。
- あなたの提出したプログラムは、標準入力・標準出力、あるいは他のファイルといかなる方法でもやりとりしてはならない。ただし、標準エラー出力にデバッグ情報等を出力することは許される。

コンパイル・実行の方法

作成したプログラムをテストするための、採点プログラムのサンプルが、コンテストサイトからダウンロードできるアーカイブの中に含まれている。このアーカイブには、提出しなければならないファイルのサンプルも含まれている。

採点プログラムのサンプルは 1 つのファイルからなる。そのファイルは `grader.cpp` である。作成したプログラムをテストするには、これらのファイル `grader.cpp`, `minerals.cpp`, `minerals.h` を同じディレクトリに置き、次のようにコマンドを実行する。

```
g++ -std=gnu++14 -O2 -o grader grader.cpp minerals.cpp
```

コンパイルが成功すれば、`grader` という実行ファイルが生成される。

実際の採点プログラムは、採点プログラムのサンプルとは異なることに注意すること。採点プログラムのサンプルは単一のプロセスとして起動する。このプログラムは、標準入力から入力を読み込み、標準出力に結果を出力する。



採点プログラムのサンプルの入力

採点プログラムのサンプルは標準入力から以下の入力を読み込む。

```
N
X1 Y1
⋮
XN YN
```

X_i, Y_i ($1 \leq i \leq N$) は、 X_i 番目の切片と Y_i 番目の切片が同じ種類の鉱物であることを表す。

採点プログラムのサンプルの出力

採点プログラムのサンプルは標準出力へ以下の情報を出力する (引用符は実際には出力されない)。

- 正解の場合、関数 `Query` の呼び出し回数が “Accepted: 100” のように出力される。
- 不正解の場合、不正解の種類が “Wrong Answer [1]” のように出力される。

実行するプログラムが複数の不正解の条件を満たした場合、表示される不正解の種類はそれらのうち 1 つのみである。

制約

すべての入力データは以下の条件を満たす。 X_i, Y_i の意味については「採点プログラムのサンプルの入力」の項目を参照せよ。

- $1 \leq N \leq 43\,000$.
- $1 \leq X_i \leq 2N$ ($1 \leq i \leq N$).
- $1 \leq Y_i \leq 2N$ ($1 \leq i \leq N$).
- $X_i \neq X_j$ ($1 \leq i < j \leq N$).
- $Y_i \neq Y_j$ ($1 \leq i < j \leq N$).
- $X_i \neq Y_j$ ($1 \leq i \leq N, 1 \leq j \leq N$).



小課題

1. (6 点) $N \leq 100$.
2. (25 点) $N \leq 15\,000$, $1 \leq X_i \leq N$, $N + 1 \leq Y_i \leq 2N$ ($1 \leq i \leq N$).
3. (9 点) $N \leq 15\,000$.
4. (30 点) $N \leq 38\,000$.
5. (5 点) $N \leq 39\,000$.
6. (5 点) $N \leq 40\,000$.
7. (5 点) $N \leq 41\,000$.
8. (5 点) $N \leq 42\,000$.
9. (10 点) 追加の制約はない.

やり取りの例

採点プログラムのサンプルが読み込む入力の例と、それに対応する関数の呼び出しの例を以下に示す.

入力例 1	関数の呼び出しの例		
	呼び出し	呼び出し	戻り値
4	Solve(4)		
1 5		Query(1)	1
2 6		Query(2)	2
3 4		Query(5)	2
7 8		Query(2)	1
		Answer(3, 4)	(なし)
		Answer(5, 1)	(なし)
		Answer(8, 7)	(なし)
		Answer(2, 6)	(なし)