



図書館 (Library)

数百年の時を経て、JOI 市は廃墟となってしまった。探検家の IOI ちゃんは、図書館があった地域を調査している。調査により、以下のことが分かっている。

- JOI 市の図書館の書庫には、 N 冊の本が横一列に並べられていた。
- N 冊の本には、1 から N までの番号が付けられていた。ただし、書庫には、本が番号順に並べられていたとは限らない。
- 1 回の作業で、書庫から、連続して並べられている何冊かの本を同時に取り出すことができた。

IOI ちゃんは、残念ながら、かつての蔵書を見つけることはできなかったが、図書館の書庫管理を行っていた機械を発見することはできた。1 冊以上の何冊かの本を番号で指定してこの機械に 1 回質問すると、それらの本のみを書庫から取り出すのに必要な作業の回数の最小値を求めることができる。

IOI ちゃんは、この機械に何回か質問することで、書庫に本が並んでいた順番を特定したい。ただし、 N 冊の本が逆順に並んでいたとしても機械の回答は変わらないため、左右方向の順番を特定する必要はない。

古い機械なので、質問の回数は 20 000 回以内に抑えたい。

課題

機械に 20 000 回以内の質問を送ることで、書庫に本が並んでいた順番を特定するプログラムを作成せよ。ただし、左右方向の順番を特定する必要はない。

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 `library.cpp` という名前である。このファイルは以下のルーチンを実装していなければならない。そのプログラムは `library.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 は蔵書の冊数 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(const std::vector<int>& M)`

この関数は、何冊かの本を番号で指定すると、それらの本のみを書庫から取り出すのに必要な作業の回数の最小値を返す。



- ◇ 取り出す本を引数 M で指定する。 M はサイズ N の vector である。それぞれの $i (1 \leq i \leq N)$ について、 $M[i-1] = 0$ のときは番号 i の本を取り出さないことを、 $M[i-1] = 1$ のときは番号 i の本を取り出すことを表す。 M のサイズが N でなかった場合、不正解 [1] と判定される。それぞれの i について、 $M[i-1]$ の値は 0 か 1 でなければならない。また、 $M[i-1] = 1$ となる $i (1 \leq i \leq N)$ が少なくとも一つ存在しなければならない。この 2 つの条件のうち、少なくとも一方が満たされていない場合、不正解 [2] と判定される。Query を 20000 回を超えて呼び出した場合、不正解 [3] と判定される。

★ void Answer(const std::vector<int>& res)

この関数を用いて、書庫に本が並んでいた順番を回答する。ただし、左右方向の順番を特定する必要はない。

- ◇ 引数 res は、書庫に本が並んでいた順番を表すサイズ N の vector である。それぞれの $i (1 \leq i \leq N)$ について、 $res[i-1]$ は、左から i 番目の本の番号である。 res のサイズが N でなかった場合、不正解 [4] と判定される。 $res[i-1]$ は 1 以上 N 以下の整数でなければならない。そうでない場合、不正解 [5] と判定される。また、 $res[0], res[1], \dots, res[N-1]$ は互いに重複してはならない。そうでない場合、不正解 [6] と判定される。

関数 Solve の実行の終了時に関数 Answer の呼び出し回数が 1 回でなかった場合、不正解 [7] と判定される。

関数 Solve で回答された順番が、書庫に本が並んでいた順番と異なる場合、不正解 [8] と判定される。ただし、左右方向の順番を特定する必要はない。

重要な注意

- 内部での使用のために他のルーチンを実装したり、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 `library.cpp` とする。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`, `library.cpp`) と、`library.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=c++14 -O2 -o grader grader.cpp library.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。

- 1 行目には、整数 N が書かれている。これは蔵書の冊数が N であることを表す。
- 続く N 行のうちの i 行目 ($1 \leq i \leq N$) には、整数 A_i が書かれている。これは、書庫に並んでいた本のうち、左から i 番目の本の番号が A_i であったことを表す。

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。

制限

すべての入力データは以下の条件を満たす。 N, A_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。

- $1 \leq N \leq 1000$.
- $1 \leq A_i \leq N$ ($1 \leq i \leq N$).
- $A_i \neq A_j$ ($1 \leq i < j \leq N$).



小課題

この課題では小課題は全部で2個ある。各小課題の配点および追加の制限は以下の通りである。

小課題 1 [19 点]

- $N \leq 200$.

小課題 2 [81 点]

追加の制限はない。

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	ルーチンの呼び出しの例			
	呼び出し	戻り値	呼び出し	戻り値
5	Solve(5)			
4			Query({ 1, 1, 1, 0, 0 })	
2				2
5			Answer({ 4, 2, 5, 3, 1 })	
3				(なし)
1				

この課題では、書庫に本が並んでいた左右方向の順番を特定する必要はないことに注意せよ。左右方向の順番を逆にした `Answer({ 1, 3, 5, 2, 4 })` を呼び出しても正解となる。