



古代の機械 (Ancient Machine)

考古学者の Anna と Bruno は IOI 国の遺跡で発掘調査をすることになった。Anna は遺跡 A で古代の機械の設計書を、Bruno は遺跡 B で古代の機械を発見した。

この機械は 1 本の電気線上に、 N 個の装置が横一列に取り付けられた構造をしている。装置には 3 つの種類があり、それぞれ X, Y, Z と名前が付いている。装置には左から順に 0 から $N-1$ までの番号が付いており、装置 i ($0 \leq i \leq N-1$) の種類は S_i である (つまり、 S_i は X, Y, Z のいずれかである)。

この機械はとても大きいため、Bruno は機械に付いている N 個の装置すべてを 1 つずつ電気線上から取り外して持ち帰ることにした。ただし、装置は電気線上で相互に作用しているため、装置を取り外す順番には気を付ける必要がある。

電気線上の装置の取り外し方について、次のように定める。

- 装置 x, y, z ($0 \leq x < y < z \leq N-1$) はまだ取り外されておらず、 $S_x = X, S_y = Y, S_z = Z$ を満たすとす。さらに、 $x < j < y$ を満たすすべての装置 j と、 $y < k < z$ を満たすすべての装置 k は取り外されているとする。このとき、装置 y を電気線から取り外す取り外し方を良い取り外し方という。
- これ以外の取り外し方は、良い取り外し方ではない。

Bruno は、良い取り外し方をした回数が最大になるように、 N 個の装置すべてを取り外す必要がある。しかし、3 つの種類の装置は見た目がとても似ているため、Bruno にはどの装置がどの種類かを判別することができない。

Anna は設計書を持っているので、どの装置がどの種類であるかが分かっている。そこで、送信機を用いて Bruno に協力することにした。Anna は Bruno に 0 または 1 を何回か送ることができる。

良い取り外し方をした回数が最大になるように Bruno が装置を取り外すための、Anna と Bruno の戦略を実装したプログラムを作成せよ。この課題では、Anna が Bruno に 0 または 1 を送る回数が少ないほど、あなたは高得点を得る。

実装の詳細

あなたは 2 つのファイルを提出しなければならない。

1 つ目のファイルは `Anna.cpp` という名前である。このファイルは Anna の戦略を実装したファイルであり、以下の関数を実装していなければならない。また、`#include` プリプロセッサ指令によって `Anna.h` を読み込むこと。

- `void Anna(int N, std::vector<char> S)`
この関数は、最初に 1 回だけ呼び出される。



- 引数 N は、装置の個数 N である。
- 引数 S は長さ N の配列であり、 $S[i]$ は装置 i の種類 S_i を表す ($0 \leq i \leq N-1$)。 $S[i]$ は文字 'X', 'Y', 'Z' のいずれかである。

このファイルでは以下の関数を呼び出すことができる。

★ void Send(int a)

この関数を用いて、Bruno に 0 または 1 を送る。

- ◇ 引数 a は、Bruno に送る情報を表す。 a は 0 または 1 でなければならない。これが満たされない場合、不正解 [1] と判定される。
- ◇ 関数 Send を 200 000 回を超えて呼び出してはならない。200 000 回を超えて呼び出した場合、不正解 [2] と判定される。

2 つ目のファイルは Bruno.cpp という名前である。このファイルは Bruno の戦略を実装したファイルであり、以下の関数を実装していなければならない。また、#include プリプロセッサ指令によって Bruno.h を読み込むこと。

● void Bruno(int N, int L, std::vector<int> A)

この関数は、Anna が呼び出された後に 1 回だけ呼び出される。

- 引数 N は、装置の個数 N である。
- 引数 L は、Anna から送られてきた 0 または 1 の個数 L である。
- 引数 A は長さ L の配列であり、Anna から 0 または 1 が $A[0], A[1], \dots, A[L-1]$ の順に送られてきたことを表す。

このファイルでは以下の関数を呼び出すことができる。

★ void Remove(int d)

この関数を用いて、装置の取り外す順番を回答する。

- ◇ 引数 d は、次に装置 d を取り外すことを表す。
- ◇ $0 \leq d \leq N-1$ でなければならない。これが満たされない場合、不正解 [3] と判定される。
- ◇ d に同じ値を 2 回以上指定してはならない。これが満たされない場合、不正解 [4] と判定される。
- ◇ 関数 Remove はちょうど N 回呼び出される必要がある。関数 Bruno の実行の終了時に関数 Remove の呼び出し回数が N 回でなかった場合、不正解 [5] と判定される。
- ◇ N 個の装置を取り外し終えたとき、良い取り外し方をした回数が最大になっていなければならない。これが満たされない場合、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。ただし、提出された2つのプログラムは、採点プログラムとまとめてリンクされて1つの実行ファイルになるので、各ファイル内のすべてのグローバル変数と内部関数を無名名前空間内で宣言して、他のファイルとの干渉を避ける必要がある。採点時には、このプログラムは Anna 側、Bruno 側として2個のプロセスとして実行されるので、Anna 側と Bruno 側でプログラム中のグローバル変数を共有することはできない。
- 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、`grader.cpp`, `Anna.cpp`, `Bruno.cpp`, `Anna.h`, `Bruno.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++17 -O2 -fsigned-char -o grader grader.cpp Anna.cpp Bruno.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形式で入力を読み込む。

$$N$$
$$S_0 S_1 \cdots S_{N-1}$$

S_i と S_{i+1} ($0 \leq i \leq N-2$) は、半角空白で区切られている。



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).

- 不正解 [1], [2], [3], [4], [5] のいずれかの場合, 不正解の種類が “Wrong Answer [1]” のように出力される.
- そうでない場合, `Send` が呼び出された回数 L と, 良い取り外し方をした回数 D が “Accepted: L D” のように出力される. 採点プログラムのサンプルは, 実際の採点プログラムと違って不正解 [6] であるかを判定しないことに注意せよ.

実行するプログラムが不正解 [1], [2], [3], [4], [5] のうち, 複数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

- $3 \leq N \leq 100\,000$.
- S_i は X, Y, Z のいずれかである ($0 \leq i \leq N - 1$).

小課題

1. (5 点) $N \leq 18$.

2. (95 点) 追加の制約はない. この小課題では, 以下に従い得点が決定される.

- この小課題のすべてのテストケースにおける, `Send` が呼び出された回数の最大値を L とする.
- このとき, 小課題の得点は以下のように決定される.

◦ $160\,000 < L \leq 200\,000$ のとき, $25 + \left\lfloor 10 \times \frac{200\,000 - L}{40\,000} \right\rfloor$ 点.

◦ $100\,000 < L \leq 160\,000$ のとき, $35 + \left\lfloor 30 \times \frac{160\,000 - L}{60\,000} \right\rfloor$ 点.

◦ $70\,000 < L \leq 100\,000$ のとき, $65 + \left\lfloor 30 \times \left(\frac{100\,000 - L}{30\,000} \right)^2 \right\rfloor$ 点.

◦ $L \leq 70\,000$ のとき, 95 点.

ここで, $\lfloor x \rfloor$ は x を超えない最大の整数を表す.



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
4 X Y X Z	Anna(4, {X, Y, X, Z})	
		Send(0)
		Send(1)
	Bruno(4, 2, {0, 1})	
		Remove(2)
		Remove(1)
		Remove(0)
		Remove(3)

この呼び出し例においては、4 個の装置は次のように取り外される。

1. まず最初は 4 個の装置は X Y X Z となっている。
2. 装置 2 を取り外すと、装置は X Y - Z となる (取り外した装置を - で表す)。
3. 次に装置 1 を取り外すと、装置は X - - Z となる。 $(x, y, z) = (0, 1, 3)$ とすると条件を満たすため、これは良い取り外し方である。
4. 次に装置 0 を取り外すと、装置は - - - Z となる。
5. 最後に装置 3 を取り外すと、装置は - - - - となる。

良い取り外し方をした回数は 1 回である。

この入力例では、2 回以上の良い取り外し方をすることはできない。