



## 伝達 (Transmission)

Anna は整数  $N$  が好きである。しかし、Bruno は  $N$  を知らないので、Anna は電子メールを送ることで  $N$  を Bruno に伝えたい。送信する電子メールは `0` と `1` からなる文字列でなければならない。また、料金の都合上、電子メールは 100 000 文字以下にしたい。

Anna と Bruno の戦略を実装したプログラムを作成せよ。

### 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。そのプログラムは `#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)`

この関数は各テストケースでちょうど 1 回呼び出される。

- 引数  $N$  は Anna が好きな整数  $N$  である。

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。

- ★ `void Send(std::string S)`

この関数を用いて、Bruno に電子メールを送信する。

- 引数  $S$  は Bruno に送信する電子メールを表す。
- 引数  $S$  の長さは 1 以上 100 000 以下でなければならない。これが満たされていない場合、不正解 [1] と判定される。
- 引数  $S$  の各文字は `'0'` または `'1'` でなければならない。これが満たされていない場合、不正解 [2] と判定される。
- 関数 `Send` はちょうど 1 回呼び出さなければならない。関数 `Send` を 2 回以上呼び出した場合、不正解 [3] と判定される。関数 `Anna` の実行の終了時に関数 `Send` が 1 回も呼び出されていない場合、不正解 [4] 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。そのプログラムは `#include` プリプロセッサ指令によって `Bruno.h` を読み込むこと。

- `int Bruno(std::string S)`

この関数は各テストケースで関数 `Anna` が呼び出された後にちょうど 1 回呼び出される。この関数は `Anna` が好きな整数 `N` と等しい整数を返さなければならない。

- 引数 `S` は `Anna` が送信した電子メールを表す。
- 戻り値は 1 以上 100 000 以下の整数でなければならない。これが満たされていない場合、不正解 [5] と判定される。
- 戻り値が `Anna` の好きな整数 `N` と一致していない場合、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。なお、アーカイブの中に含まれている `compile.sh` というファイルを代わりに実行してもよい。

```
g++ -std=gnu++20 -O2 -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。



## 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。

$N$

## 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，Anna が送信した電子メールの長さが“Accepted: 22”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

- $1 \leq N \leq 100\,000$ .
- $N$  は整数である。



## 採点基準

この課題のテストケースの中で、1つでも不正解 [1] ~ [6] (実装の詳細を参照) と判定されたものや、実行時間制限超過、メモリ制限超過、実行時エラーと判定されたものがあつた場合、0点となる。

そうでない場合、この課題のすべてのテストケースに対する Anna が送信した電子メールの長さの最大値を  $L$  として、この課題の得点は以下のように与えられる。

- $101 \leq L \leq 100\,000$  のとき、10 点.
- $18 \leq L \leq 100$  のとき、30 点.
- $L = 17$  のとき、60 点.
- $L \leq 16$  のとき、100 点.

## やりとりの例

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		
	Anna の呼び出し	Bruno の呼び出し	Bruno の戻り値
3	Anna(3)		
	Send("00110101")		
		Bruno("00110101")	
			3

この入力例では、Anna が好きな整数は 3 であり、それを伝えるために Anna は電子メール “00110101” を Bruno に送っている。Bruno は電子メール “00110101” を受信し、Anna の好きな整数が 3 であると推測している。

コンテストサイトからダウンロードできるファイルのうち、sample-01-in.txt は入力例 1 に対応する。

コンテストサイトからダウンロードできるファイルに含まれる sample-01-in.txt は、採点プログラムのサンプルの入力として用いることができる。