

Task: KOL

Bytemon Collector

Farias Brito High School Programming Camp, Day 4. Available memory: 8 MB.

28.07.2016

Byteasar loves collecting Bytémon cards. Each card in his deck has a picture of one Bytémon together with its catalog number, which is an integer from the range $[1, 2 \cdot 10^9]$. So far, Byteasar has not collected all the Bytémóns. In fact, his collection is a peculiar one: each Bytémón in it is depicted on several cards; moreover, each one is depicted on the same number of cards.

One day, Bytesar has realized that someone must have stolen a few (i.e., one or more) cards from his collection. He knows that all the stolen cards were the same, i.e., depicted the same Bytémón. And fortunately, at least one more such card remains in his collection. Unfortunately though, our hero is of very little brain and has already forgotten which Bytémón it is. Are you able to help him out by reminding him which cards he is missing? Note that your program too has to conform with severe memory limitations. . .

Write a program to communicate with a library that enables sweeping Byteasar's deck of cards that will determine the catalog number of the stolen Bytémón.

Communication

To use the library, you should write the following in the preamble of your program:

```
#include "ckollib.h"
```

The library provides the following functions and procedures:

- `int karta();`

(Polish for *card*)

Returns an integer from the range $[1, 2 \cdot 10^9]$ or 0. The 0 value signifies the end of the deck, whereas a positive integer – a catalog number of the Bytémón on the successive card. After receiving the 0 value, your program may still call the function `karta` (Polish for *card*) – this corresponds to another pass through the deck. In each pass, the catalog numbers of the Bytémóns are given in the same order.

- `void odpowiedz(int wynik);`

(Polish for *answer* and *result*)

Tell the library that `wynik` is the catalog number of the stolen Bytémón. Calling this function **terminates your program**.

Your program **cannot** read any data, neither from the standard input nor from any files. Similarly, it **cannot** write anything to the standard output or to any files. It may write to the standard error (`stderr`) – but keep in mind that this uses precious time.

Grading

For any test, the scoring rules are as follows:

- Your program will receive the full score if it makes at most one pass through the deck, i.e., if it does not call the function `karta` after it has returned 0 for the first time.
- If your program starts a second pass through the deck, it will receive only 40% of the score.
- If your program starts the third pass, it will be terminated by the library and no score will be awarded.

There are no more than 60 000 000 cards in the deck. Moreover, in tests worth 20% of the total score, there are no more than 200 000 cards in the deck.

Sample program execution

C/C++	Result	Explanation
<code>k = karta();</code>	13	Catalog number of the Bytémon on the first card.
<code>k = karta();</code>	13	
<code>k = karta();</code>	39	
<code>k = karta();</code>	26	
<code>k = karta();</code>	26	
<code>k = karta();</code>	0	End of the deck.
<code>k = karta();</code>	13	New pass through the deck; the cards are in the same order as previously.
<code>odpowiedz(39);</code>		Giving the answer. Program terminates.

The above program execution is correct in that it performs no more than two passes through the deck and returns the correct value. However, it starts the second pass through the deck, and so would receive only 40% of the score. To be awarded the full score, a program has to do with only a single pass.

Experiments

In the directory `dlazaw`, there is a sample library that lets you test the formal correctness of your solution. The library reads the deck of cards description from the standard input in the following format:

- in the first line, there is a single positive integer n – the number of cards in the deck,
- in the second line, there are n positive integers from the range $[1, 2 \cdot 10^9]$ – the numbers of Bytémions on successive cards of the deck.

The sample library **does not check** if indeed there is only a single Bytémon depicted on fewer cards in the deck than any other Bytémon.

A sample input for the library can be found in the file `ko10.in`. After a call of the procedure `odpowiedz`, the library prints to the standard output information about the given answer, number of calls of the function `karta`, and the number of passes.

In the same directory, there are sample solutions `ko1.c` and `ko1.cpp`, that utilize the library. These are not correct, as they always answer that the number sought by Byteasar is the one on the last card in deck.

To compile your solution together with the library, use the following commands:

- C: `gcc -O2 -static ckollib.c ko1.c -lm`
- C++: `g++ -O2 -static ckollib.c ko1.cpp -lm -std=gnu++0x`

The file with your solution and the library should be placed in the same directory.