НАЦИОНАЛНА ШКОЛА ПО ИНФОРМАТИКА
СТАРА ЗАГОРА, 24-30 АВГУСТ 2017 Г.

Топ секретно

На Ави ѝ е възложена топ секретна мисия, при изпълнението на която момичето трябва да се представя за служител в дадена фирма (детайлите на операцията са строго поверителни). Във фирмата работят $N+1$ служители, номерирани $0, 1, ..., N$. Ави се представя за служител номер N, който тя се е погрижила да не е на работа. Всичко върви гладко, докато не идва време за обяд...

Служителите във фирмата се нареждат на опашката за храна в точно определен ред. Нека означим с a_i кой подред се нарежда i-тият служител. Разбира се, редицата $a_0, a_1, ..., a_N$ е пермутация, т.е. $a_i \neq a_j$ за всички $0 \leq i < j \leq N$ и $0 \leq a_i \leq N$ за всяко $0 \leq i \leq N$. Проблемът е, че Ави не знае този ред и съответно не знае къде в опашката да застане. Тя трябва да заеме място a_N (мястото на служител N) - ако не го стори, рискува да бъде разкрита.

За радост Ави не е в пълна безизходица, тъй като тя може да извлича информация от другите служители. Момичето може да пита служител i дали е сред първите k души в опашката, т.е. дали $a_i < k$ ($0 \leq i \leq N-1$). Разбира се, тя не може да задава въпроси на служител номер N, защото него го няма. За да не буди подозрения, Ави може да зададе най-много Q такива въпроса.

Вашата задача е да напишете програма, която помага на Ави да открие стойността на a_N, задавайки най-много Q въпроса.

Детайли по имплментацията

Вашата програма трябва да съдържа в началото си `#include "topsecret.h"`.
Вашата програма трябва да реализира следната функция:

- `int find_position(int N, int Q)`
  - `N`: номерът на служителя, за когото Ави се представя. Общият брой на служителите във фирмата е $N + 1$.
  - `Q`: максималният разрешен брой въпроси за съответния тест
  - Функцията трябва да връща стойността на a_N - позицията, на която трябва да застане Ави.
  - Функцията се вика веднъж за всеки тест.

Освен горната функция Вашата програма може да ползва и реализира други вътрешни функции и глобални променливи. Тя трябва *да не съдържа* функция `main`.

Вашата програма може да задава въпроси, използвайки следната функция:

- `bool is_less(int i, int k)`
  - `i`: индексът на служителя, за чиято позиция в опашката се пита. Трябва да изпълнява $0 \leq i \leq N-1$.
  - `k`: стойността, с която се сравнява позицията на i-тия служител. Трябва да изпълнява $0 \leq k \leq N$.
  - Функцията връща `true`, ако и `false` в противен случай.
  - Функцията може да се вика най-много Q пъти за един тест.

В случай че нарушите ограниченията за параметрите на функцията, ще получите съобщение *„Invalid question asked.“*.

Ако извикате `is_less` повече от Q пъти, ще получите съобщение *„Too many questions asked.“*.

Пример

Грейдърът извършва следното викане на функция:

- `find_position(3, 15)`: Служителите са общо 4, Ави е служител номер 3 и програмата може да зададе най-много 15 въпроса.

Програма задава следните въпроси:

- `is_less(0, 1)` връща `true`
- `is_less(1, 3)` връща `true`
- `is_less(1, 2)` връща `false`
- `is_less(2, 2)` връща `false`

Единствената редица, която отговаря на горните отговори е $[0, 2, 3, 1]$. Следователно `find_position` трябва да върне 1.

Подзадачи

Във всички тестове $a_i \neq a_j$ за всички $0 \leq i < j \leq N$ и $0 \leq a_i \leq N$ за всяко $0 \leq a_i \leq N$.

Тестовете са групирани в подзадачи, като точките за дадена подзадача се получават само ако програмата Ви премине всички тестове успешно.

- (10 точки) $1 \leq N \leq 1\,000$, $Q = 2\,000\,000$
- (15 точки) $1 \leq N \leq 100\,000$, $Q = 2\,000\,000$,
- (15 точки) $1 \leq N \leq 100\,000$, $Q = 200\,000$, $a_i < a_{i+1}$ за всяко $0 \leq i \leq N - 2$
- (60 точки) $1 \leq N \leq 100\,000$, $Q = 200\,000$

Локално тестване

За да можете да тествате решението си на компютъра си, Ви се предоставят файловете `Lgrader.cpp` и `topsecret.h`, които да компилирате заедно с Вашето решение `topsecret.cpp`.

Вход

- Ред 1: две цели числа N и Q
- Ред 2: $N + 1$ числа $a_0, a_1, ..., a_N$

Изход

- Ред 1:
  - *„Output is correct“*, ако програмата е преминала успешно теста
  - *„Invalid question“*, ако програмата е задала въпрос, неотговарящ на гореописаните ограничения
  - *„Too many questions asked“*, ако програмата е извършила повече от Q извиквания на `is_less`
  - *„Output isn't correct“*, ако програмата не е надвишила максималния разрешен брой въпроси, но не е намерила правилно търсената стойност.

Изпращане на тестове към системата

Можете да изпращате собствени тестове към системата. Форматът на входните и изходните данни е същият като този на предоставения локален грейдър.