

XXXII НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАТИКА

Национален кръг
Хасково, 22 – 25 април 2016 г.
Група АВ, 9 – 12 клас, Ден 2

Задача АВ4. ПОЗНАЙ ЧИСЛОТО

Автори: Николай Белухов и Йордан Чапъров

В тази задача Вие ще играете срещу журито, като трябва да познаете намислено от него цяло число в интервала $[0, N)$, т.е., ако числото е x , то са изпълнени неравенствата $0 \leq x < N$. На всеки ход трябва да зададете въпрос от вида: „Принадлежи ли намисленото число на цикличния интервал $[a, b)$, където $0 \leq a, b \leq N$?“, на който ще получите отговор „Да“ или „Не“.

Пояснение: Под цикличен интервал се разбира

$[a, b) = \{x \mid a \leq x < b\}$ ако $a < b$;

$[a, b) = \{x \mid a \leq x < N \text{ или } 0 \leq x < b\}$ ако $a > b$

Ако $a = b$, то интервалът $[a, b)$ е празен.

До тук всичко изглежда лесно, но работата е там, че, за да направи играта по-интересна, журито може да реши да дава лъжлив отговор на всеки въпрос, започвайки от даден момент нататък. Т.е. вие знаете, че до някакъв пореден въпрос всички отговори на журито са коректни, а след него всички са лъжи, но не знаете колко коректни отговора да очаквате. Журито е много своенравно и може да започне да лъже още от самото начало, а може да дава верни отговори през цялото време.

Освен, че трябва да познае числото, Вашата програма трябва да свърши тази работа с не повече от определен брой въпроси, както е зададено в подзадачите, описани по-долу.

Важно: Вашата програма трябва да играе така, че с максималния разрешен брой въпроси, зададен за всяка подзадача, да познае всяко число от интервала $[0, N)$, което журито намисли, при всеки начален въпрос, от който започва лъгането.

Задача

Напишете функция `play()`, която ще се компилира заедно с програмата на журито и трябва да провежда диалог с нея, докато отгатне намисленото число.

Детайли по реализацията

Функцията `play()` трябва да има следния формат

```
long long play(long long N)
```

Тя се вика веднъж от програмата на журито и трябва да върне числото, което си мисли, че е познала.

За комуникация с програмата на журито Ви се предоставя функцията

```
bool isInside(long long l, long long r);
```

Чрез извикване на функция `isInside`, Вие питате програмата на журито дали числото, което трябва да познаете, се намира в цикличния интервал $[l, r)$ (вижте по-горе определението за цикличен интервал). Тя връща `true` (да) или `false` (не).

Тази функция се вика многократно до момента, в който Вашата програма реши, че е познала числото на журито. Не забравяйте, че от някое викане нататък, програмата на журито може да започне да лъже.

Вие трябва да предадете към системата файл `findnumber.cpp`, който съдържа функция `play()`. Той може да съдържа и друг код, необходим за работата на функция `play`, но не трябва да съдържа `main()`.

В началото си Вашият файл трябва да съдържа `#include "findnumber.h"`

XXXII НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАТИКА

Национален кръг
Хасково, 22 – 25 април 2016 г.
Група АВ, 9 – 12 клас, Ден 2

Пример

Програмата на журито вика $play(4)$ ($N = 4$, т.е. числото е 0, 1, 2 или 3)

Вика се <code>isInside</code> за интервал	Отговор на журито	Коментар
[2,0)	True	Интервалът е цикличен, т.е. пита се дали числото е 2 или 3. Не се знае отговорът коректен ли е или не.
[2,2)	False	Интервалът е празен и значи отговорът е коректен. Това значи, че до момента журито не е лъгало и предният отговор също е коректен, т.е. числото е 2 или 3.
[0,4)	False	Интервалът включва всички числа и значи журито лъже. Това значи, че от тук нататък винаги ще лъже.
[2,3)	False	Въпросът е „числото 2 ли е?“, на което журито отговаря „не“. Но тъй като журито вече лъже, то намисленото число е 2. Функция <code>play</code> трябва да върне 2.

Подзадачи

Подзадача	Точки	N	Максимален брой извиквания на <code>isInside</code>
1	9	$1 \leq N \leq 100$	200
2	11	$100 < N \leq 10^9$	65
3	22	$10^9 < N \leq 10^{12}$	48
4	23	$1 \leq N \leq 92$	10
5	35	$10^{12} < N \leq 10^{15}$	56

Локално тестване

За да можете, при желание, да тествате функцията си `play` на локалния компютър, Ви се предоставят файловете `Lgrader.cpp` и `findnumber.h`. Компилирайте ги заедно с Вашия файл `findnumber.cpp` и ще получите програма, която можете да ползвате за тестване на функцията си. Входните данни за тази локална програма са със същия формат както входните данни за програмата на журито – един ред с две цели, положителни числа, разделени с един интервал: N и максималния разрешен брой викания на функция `isInside`. Числото, което трябва да се познае, и номерът на въпроса, от който започва лъгането в локалния грейдър, се генерират случайно. Това може да не Ви върши особено добра работа за тестването, но нищо не пречи да промените локалния грейдър, като направите така, че числото и стартовия въпрос за лъгането да си ги въвеждате Вие.

Изпращане на собствени тестове към системата

Вие можете да изпращате собствени тестове към системата. Входният файл трябва да съдържа един ред с две цели, положителни числа, разделени с един интервал: N и максималния разрешен брой викания на функция `isInside`. На изход ще получите:

- ако Вашата програма познае числото с допустимия брой въпроси: едно служебно положително число, чийто смисъл за Вас е, че всичко е наред.
- ако Вашата програма не познае числото: -1 и още две числа – първото е отговорът, който е дала Вашата функция `play`, а второто – верният отговор.
- ако Вашата програма познае числото, но зададе повече от допустимия брой въпроси: -2 и още две числа – първото е броят въпроси, които е задала Вашата функция `play`, а второто – максималният разрешен брой въпроси, който сте задали във входа.