

## D. Igra pogađanja

Zadatak	Guessing Game
Vremensko ograničenje	4 sekunde
Memorijsko ograničenje	1 gigabajt

Iva je nedavno započela svoje avanture tramvajima. U ulici gdje je njezina kuća ima ukupno  $N$  kuća (ubrajajući i njezinu), a označene su od 0 do  $N - 1$ .

Njezine prijateljica Maša i Lana ostale su toliko očarane njezinim pustolovnim pričama da su odlučile i same okušati se i tramvajima doći do Ivine kuće. Nakon putovanja punog iznenađenja, Maša i Lana stigle su do Ivine ulice. No, Iva je odlučila ne reći im u kojoj kući živi, već igrati igru pogađanja s njima.

Prije početka igre, Maša i Lana su prebrojale kuće, dogovorile strategiju igranja i odabrale prirodan broj  $K$ . Kada igra započne, Maša i Lana više ne mogu razgovarati jedna s drugom.

Igra ima dvije faze. U prvoj fazi, Iva odabire neki poredak obilaženja kuća takav da je njezina kuća zadnja u nizu. Zatim, ona i Maša kreu u obilazak kuća u tom poretku, bez da je Maša unaprijed upoznata s poretkom. Ispred svake kuće, koja nije Ivina, Maša smije napisati jedan prirodan broj između 1 i  $K$ . Ispred zadnje kuće koju posjete, Ivine kuće, Iva će sama odabrati prirodan broj između 1 i  $K$  koji će napisati.

Prije druge faze igre, pored Ivine kuće započela je svirati melodija *Pink Panthera* i Iva i Maša su se brzo sakrile. Lani nije preostalo ništa drugo nego prošetati pored svih kuća, pogledati brojeve zapisane ispred njih, i probati sama odgonetnuti koja je Ivina kuća. Budući da je ostala sama, ima pravo dva puta pogađati.

Možete li odrediti strategiju tako da Lana može pogoditi Ivinu kuću u dva pokušaja? Vaša strategija bodovati će se u ovisnosti o broju  $K$  (čim manje, tim bolje).

### Implementacija

Ovo je *multi-run* zadatak, što znači da će se vaš program pokrenuti više puta. Prvi put kad se pokrene, implementirati će se Mašina strategija. Zatim će se implementirati Lanina strategija.

U prvom retku ulaza su dva prirodna broja  $P$  i  $N$ , gdje je  $P$  jednak ili 1 ili 2 (prva ili druga faza), a  $N$  broj kuća.

**Osim u probnim primjerima (koji se neće koristiti za bodovanje),  $N$  će uvijek biti 100 000.**

Nastavak ulaznih podataka ovisi o fazi:

### Faza 1

Vaš program treba započeti ispisivanjem broja  $K$  ( $1 \leq K \leq 1\,000\,000$ ). Zatim,  $N - 1$  puta mora učitati redak sa cijelim brojem  $i$  ( $0 \leq i < N$ ), i ispisati redak s brojem  $A_i$  ( $1 \leq A_i \leq K$ ), gdje  $A_i$  je broj koji je Maša napisal ispred kuće  $i$ . Svaki broj  $i$ , osim broja Ivine kuće, pojavit će se točno jednom, u poretku koji je odredio ocjenjivač.

### Faza 2

Vaš program treba učitati redak s  $N$  prirodnih brojeva  $A_0, A_1, \dots, A_{N-1}$ , gdje je  $A_i$  broj napisan ispred kuće  $i$ .

Zatim, vaš program treba ispisati dva prirodna broja  $s_1$  i  $s_2$  ( $0 \leq s_i < N$ ), pokušaji pogađanja.  $s_1$  i  $s_2$  mogu biti jednaki.

### Implementacijski detalji

Napomena: vaš program se restarta za fazu 2. To znači da ne možete spremati informacije u varijablama između pokretanja.

Nakon svakog retka izlaza, morate isprazniti standardni output, inače možete dobiti poruku Time Limit Exceeded. U Pythonu, koristite `print()`. U C++, `cout << endl; ;` ili ako koristite `printf`, onda `fflush(stdout)`.

Ocjenjivač za ovaj zadatak je **prilagodljiv**. To znači da se njegovo ponašanje može promijeniti ovisno o izlazu vašeg programa kako bi izbjegnuo prolazak heuristika. Ocjenjivač bi mogao napraviti probno pokretanje faze 1, pogledati vaš izlaz, i onda ponoviti pokretanje faze 1 koristeći informacije prethodnog pokretanja.

**Vaš program mora biti deterministički**, tj. ponašati se jednako kod višestrukog pokretanja. Ako želite koristiti slučajnost u vašem programu, koristite fiksiran *seed*. To možete učiniti tako što ćete funkciji `srand` (u C++ poslati *hard-kodiranu* vrijednost, tj. ne smijete koristiti `srand(time(NULL))`). Ako ocjenjivač odredi da vaš program nije deterministički, vratiti će poruku Wrong Answer.

Ako *zbroj* vremena izvođenja (do 3) različitih pokretanja vašeg programa prelazi vremensko ograničenje, tada ćete dobiti poruku Time Limit Exceeded.

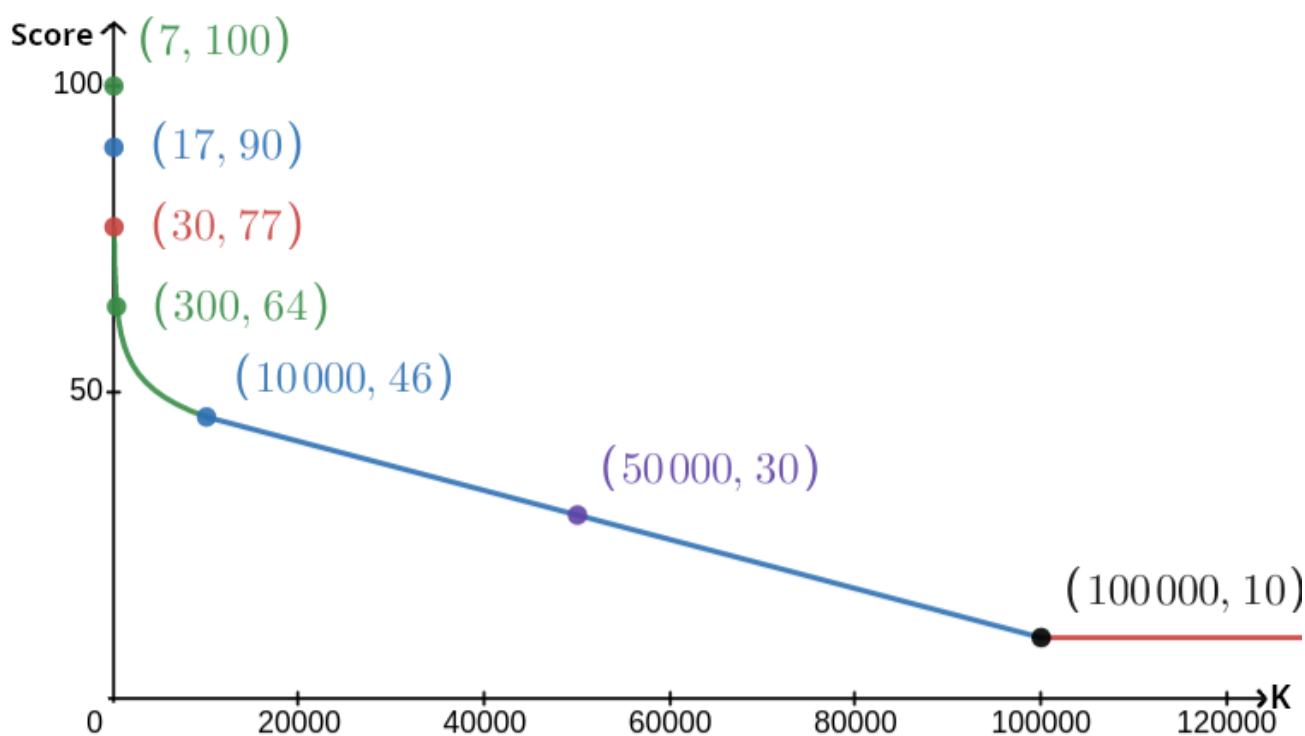
## Bodovanje

Vaše rješenje testirati će se na više testnih primjera. Ako je vaše rješenje netočno na *bilo kojem* testnom primjeru, dobit ćete 0 bodova i odgovarajuću poruku.

Neka je  $K$  maksimalan  $K$  kroz sve test primjere, tada su vaši bodovi sljedeći:

	Bodovi
$K > 99\,998$	10 bodova
$10\,000 < K \leq 99\,998$	$10 + \lfloor 40(1 - K/10^5) \rfloor$ bodova
$30 < K \leq 10\,000$	$46 + \lfloor 31(4 - \log_{10}(K))/(4 - \log_{10}(30)) \rfloor$ bodova
$7 < K \leq 30$	$107 - K$ bodova
$K \leq 7$	100 bodova

Funkcija bodova je prikazna na grafu ispod.



Probni primjer se zanemaruje za bodovanje, i vaše rješenje ne mora biti točno na njemu.

## Testing Tool

Na dnu Kattis stranice, pod "attachments" možete preuzeti *Testing Tool*, koji se razlikuje od službenog ocjenjivača.

Primjer korištenja ( $N = 4$ ,  $s = 2$ , gdje je  $s$  broj koji je Iva napisala ispred svoje kuće):

Za C++ programe, prvo prevedite (npr. `g++ -g -O2 -std=gnu++17 -static solution.cpp -o solution.out`) i zatim pokrenite:

```
python3 testing_tool.py ./solution.out <<<"4 2"
```

*Testing tool* će posjetiti kuće u nasumičnom poretku. Možete prilagoditi *testing tool* na mjestima gdje piše "MODIFY HERE".

## Probni primjeri

Probni primjer se zanemaruje za bodovanje, i vaše rješenje ne mora biti točno na njemu.

Neka je  $N = 4$ , i neka Iva živi u kući broj 1. Neka je  $A$  niz brojeva zapisani ispred kuća. Na početku je  $A = [0, 0, 0, 0]$ , što označava da još nema brojeva ispred nijedne kuće.

U prvom pokretanju vašeg koda:

$N = 4$  se učitava. Vaše rješenje odgovara  $K = 3$ .

$A_2$  se traži. Vaše rješenje odgovara 3.  $A$  je sad  $[0, 0, 3, 0]$ .

$A_0$  se traži. Vaše rješenje odgovara 1.  $A$  je sad  $[1, 0, 3, 0]$ .

$A_3$  se traži. Vaše rješenje odgovara 2.  $A$  je sad  $[1, 0, 3, 2]$ .

Konačno, ocjenjivač postavlja  $A_1 = 2$ , pa je na kraju t  $A = [1, 2, 3, 2]$ . Ovo je kraj prve faze.

U fazi 2 vašeg koda, vaše rješenje učitava niz 1 2 3 2.

Odgovara s 1 3.

Lana je uspješno odredila Ivinu kuću (jedan od pokušaja pogađanja je točan).

izlaz ocjenjivača	izlaz vašeg programa
1 4	
	3
2	
	3
0	
	1
3	
	2

izlaz ocjenjivača	izlaz vašeg programa
2 4	
1 2 3 2	
	1 3