

- A. Testrun
- B. encoded_sum
- C. air_show
- D. toothpick_sculpture

E. gold

Contest Analysis

Questions asked 1

Submissions

Testrun	
0pt	Not attempted 0/4 users correct (0%)
encoded_sum	
6pt	Not attempted 13/13 users correct (100%)
11pt	Not attempted 12/12 users correct (100%)
air_show	
5pt	Not attempted 14/14 users correct (100%)
20pt	Not attempted 1/4 users correct (25%)
toothpick_sculpture	
10pt	Not attempted 9/10 users correct (90%)
15pt	Not attempted 0/3 users correct (0%)
gold	
15pt	Not attempted 6/10 users correct (60%)
18pt	Not attempted 4/6 users correct (67%)

Top Scores

bmerry	65
sevenkplus	65
fhlasek	65
mnbvmar	65
eatmore	52
Merkurev	47
ikatanic	37
tozangezan	32
tmt514	32
wafrelka	22

Problem E. gold

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the Quick-Start Guide to get started.

small 15 points 2 minute timeout	The contest is finished.
large 18 points 10 minute timeout	The contest is finished.

Problem

Gold

A long, long time ago, on an east-west road in southeastern Asia, an ancient emperor was fleeing from the ruins of his fallen city, carrying a sack full of his gold. At times, he glanced back over his shoulder and saw pursuers chasing him, so he threw out a nugget of gold from his sack to the roadside, hoping to lighten his load and provide a distraction. The story continues, but what happened later is less important - the important part is that there is gold lying by the roadside to be picked up!

So, you took your trusty metal detector, and went to search for gold on the road. You have a really fast car, but the detector itself is somewhat slow to set up and operate, and also a bit inaccurate - it can only tell you in which direction the nearest nugget of gold is, but not how far. Also, your car cannot handle off-road driving, so you cannot triangulate; you will just need to search a bit longer.

We will represent the road as a straight line of length **L**. There will be **N** nuggets of gold on the road, at integer positions, no more than one nugget at each position. You will be able to set up the detector at integer positions on the road. After setting up, the detector will provide one of the four possible answers:

- The nearest nugget is to the east (towards decreasing position numbers),
- The nearest nugget is to the west (towards increasing position numbers),
- The nearest nugget to the west and to the east are equally distant, or
- There is a nugget at this position.

Input

The input library will be called "gold", see the sample inputs below for examples in your language. It will define three methods:

- **NumberOfNuggets():**
 - Takes no argument.
 - Returns a 64-bit integer: the number of nuggets on the road.
 - Expect each call to take 0.2 microseconds.
- **RoadLength():**
 - Takes no argument.
 - Returns a 64-bit integer: the number of positions on the road.
 - Expect each call to take 0.2 microseconds.
- **Search(i):**
 - Takes one 64-bit integer argument in the range $0 \leq i < \text{RoadLength}()$.
 - Returns a character describing the output of the metal detector: < if the nearest nugget is to the east, > if it is to the west, = if the nearest nugget to the east and west are equally distant, or X if there is a nugget as position *i*.
 - Expect each call to take 0.2 microseconds.

Output

As printing all the nugget positions would require a lot of printing, you should output one number - the bitwise XOR of the positions of all the nugget positions - as a proof you found them all.

Limits

Number of nodes: 100. (Notice that the number of nodes is the same for both the Small and Large datasets.)
Time limit: 15 seconds. (There is a 10 second overhead of initializing the test data that is not counted against this limit, so each reported time is 10 seconds more than the time your solution executed, up to a maximum of 25.)
Memory limit per node: 512 MB.
Maximum number of messages a single node can send: 5000.
Maximum total size of messages a single node can send: 8 MB.

$1 \leq \text{NumberOfNuggets}() \leq 10^7$.
 $1 \leq \text{RoadLength}() \leq 10^{11}$.

Small input

The positions of the nuggets will be generated using a pseudorandom number generator such that the probability of any subset of positions being chosen is the same.

Large input

No additional limits.

Sample

Input	Output
See input files below.	For sample input 1: 0 For sample input 2: 2 For sample input 3: 2

Sample input libraries:

Sample input for test 1: [gold.h](#) [CPP] [gold.java](#) [Java]

Sample input for test 2: [gold.h](#) [CPP] [gold.java](#) [Java]

Sample input for test 3: [gold.h](#) [CPP] [gold.java](#) [Java]

All problem statements, input data and contest analyses are licensed under the [Creative Commons Attribution License](#).

© 2008-2017 Google [Google Home](#) - [Terms and Conditions](#) - [Privacy Policies and Principles](#)

Powered by



Google Cloud Platform