Distributed Round 1 2016

A. Testrun

B. oops

C. rps

D. crates

E. winning_move

Contest Analysis

Questions asked 8

Submissions

Testrun

0pt	Not attempted **0/422 users** correct (0%)

oops

2pt	Not attempted **893/925 users** correct (97%)
12pt	Not attempted **756/882 users** correct (86%)

rps

1pt	Not attempted **789/857 users** correct (92%)
15pt	Not attempted **585/783 users** correct (75%)

crates

8pt	Not attempted **557/673 users** correct (83%)
25pt	Not attempted **258/433 users** correct (60%)

winning_move

3pt	Not attempted **635/700 users** correct (91%)
34pt	Not attempted **49/309 users** correct (16%)

Top Scores

simonlindholm	100
tomconerly	100
eatmore	100
cgy4ever	100
bmerry	100
Simon.M	100
Klockan	100
tczajka	100
tkociumaka	100
Zlobober	100

Problem D. crates

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 8 points *2 minute timeout*	The contest is finished.
large 25 points *10 minute timeout*	The contest is finished.

Problem

Rearranging Crates

You are the manager of the warehouse of the largest wharf in the area. The warehouse is tall and long, but narrow, so you have created a single row of stacks of crates. Unfortunately, the ship unloaders are usually sloppy and in a rush, so the number of crates can vary wildly between stacks, making them look like an uneven skyline of buildings.

The District Crate Judge (DCJ) is visiting the facilities tomorrow, and you want to get a perfect score in her assessment. You decided to use an old crane to rearrange the crates to make them into a neatly organized crate wall that will impress the DCJ, but this task may take a long time! Your crane can only handle one crate at a time. Moreover, the crane's wheels are not strong enough to move if the crane is handling a crate, so the crane can only drop off a crate on a stack adjacent to the one the crate came from.

The crane starts off at the leftmost stack, without any crates. The crane can carry out only the following sequence of actions, which we will call a *move*:

1. Position the crane at any stack with at least one crate. This may be the stack the crane is already at.
2. Pick up the top crate from the stack the crane is at. (Even if this causes the stack to have zero crates, it still counts as a stack.)
3. Put that crate on top of a stack directly adjacent to the stack the crane is at.

Notice that if you want to move a crate two stacks down to the right, for example, this will take two moves.

You must use some number of moves to transfer crates around so that all stacks are as even as possible, with all the excess distributed among the leftmost possible stacks. If there are N stacks and a total of C crates, you want the $C \% N$ leftmost stacks to have $\lceil C/N \rceil$ crates each and the rest of the stacks to have $\lfloor C/N \rfloor$ crates each. For instance, if there are 3 stacks and a total of 8 crates, you want the stack heights to be 3, 3, 2, from left to right.

What is the minimum number of moves you need to use to balance the stacks as specified above? Since the result can be really big, output it modulo 10^9+7 (1000000007).

Input

The input library is called "crates"; see the sample inputs below for examples in your language. It defines two methods:

- **GetNumStacks()**:
  - Takes no argument.
  - Returns a 64-bit number: the number of stacks.
  - Expect each call to take 0.12 microseconds.
- **GetStackHeight(i)**:
  - Takes a 64-bit number in the range 1 ≤ i ≤ GetNumStacks().
  - Returns a 64-bit number: the starting number of crates in stack i, counting from left to right and starting from 1.
  - Expect each call to take 0.12 microseconds.

Output

Output a single line with a single integer, the remainder of dividing the minimum number of moves needed to balance the stacks by 10^9+7 (1000000007).

Limits

Time limit: 6 seconds.
Memory limit per node: 128 MB.
Maximum number of messages a single node can send: 1000.
Maximum total size of messages a single node can send: 8 MB.

1 ≤ GetStackHeight(i) ≤ 10^9, for all i.

Small dataset

Number of nodes: 10.
1 ≤ GetNumStacks() ≤ 10^6.

Large dataset

Number of nodes: 100.
1 ≤ GetNumStacks() ≤ 10^9.

Sample

Input	Output

`See input files below.`	`For sample input 1:`
	`3`
	`For sample input 2:`
	`5`
	`For sample input 3:`
	`0`

Sample input libraries:
Sample input for test 1: crates.h [CPP] crates.java [Java]
Sample input for test 2: crates.h [CPP] crates.java [Java]
Sample input for test 3: crates.h [CPP] crates.java [Java]