Problem A. Minimal Prime Divisor

Input file:	`a.in`
Output file:	`a.out`
Time limit:	0.5 s
Memory limit:	256 Mb

Let $\operatorname{minp}(n)$ denotes the minimal prime divisor of integer $n > 1$. Denote further $f(n) = \min\{\operatorname{minp}(1 + n + n^2 + n^3 + n^4), K\}$, where K is the given positive integer. For given L and R you need to find the value of the following expression

$$\left(\sum_{n=L}^{R} n \cdot f(n)\right) \pmod{10^9 + 7}.$$

Input

The only line of input file contains three space separated positive integers K, L and R. Here $1 \le K \le 10^7$, $1 \le L \le R \le 10^{18}$ and $R - L \le 2000000$.

Output

The only line of output file should contain the answer to the problem.

Examples

a.in	a.out
`20 1 6`	`207`
`6 13 42`	`4779`

Problem B. Isosceles Triangulation

Input file:	`b.in`
Output file:	`b.out`
Time limit:	0.5 s
Memory limit:	256 Mb

Consider regular polygon with $N \geq 3$ sides. Let's cut it into $N - 2$ triangles by $N - 3$ non-intersecting diagonals. Is it possible that all these triangles are isosceles? If yes then we say that regular N-gon admits an isosceles triangulation. Write the program that answers this question.

Input

The first line contains a positive integer $T \leq 1000$, the number of test cases. Each of the next T lines contains a positive integer N, the number of sides of regular polygon. It is guaranteed that $3 \leq N \leq 10^9$.

Output

For each value of N in the input file output "`YES`" (without quotes) if regular N-gon admits an isosceles triangulation and "`NO`" otherwise.

Examples

b.in	**b.out**
5 3 4 5 6 7	YES YES YES YES NO

Problem C. XOR pairs

Input file:	`c.in`
Output file:	`c.out`
Time limit:	0.5 s
Memory limit:	256 Mb

For given non-negative integers X, A, B consider all pairs of non-negative integers (a, b) such that a xor $b = X$, $a \geq A$ and $b \geq B$. Here a xor b is the xor-operation ("xor" in Pascal, "^" in C/C++/Java). Let's sort these pairs by a in increasing order and pairs with equal value of a by b in increasing order. You need to find N-th pair among these pairs. Numeration starts from 1. But be ready to answer for thousands of such question for given X, A, B.

Input

The first line contains four integers X, A, B and T, the number of test cases. Each of the next T lines contains a positive integer N. It is guaranteed that $0 \leq X, A, B \leq 10^{18}$, $1 \leq T \leq 10000$ and $1 \leq N \leq 10^{18}$.

Output

For each value of N in the input file output two space separated integers a and b, such that pair (a, b) is N-th pair among described sequence of pairs.

Examples

c.in	**c.out**
6 1 4 10 1 2 3 4 5 6 7 8 9 10	1 7 2 4 3 5 8 14 9 15 10 12 11 13 12 10 13 11 14 8

Problem D. Win of the looser

Input file:	`d.in`
Output file:	`d.out`
Time limit:	1 s
Memory limit:	256 Mb

Two players staged a darts tournament. Tournament is a series consisting of N games. In each game participants make some number of shots, gaining points.For one game each player can get any number of points from 0 to K. Of course a game is won by one of two players who gains more points in it than the opponent. If players gain equal points in the game, then its outcome is a draw.

But determining the winner of the tournament is not so straightforward. By some rules the winner is the one who's won more games in the series. On the other the one who's gained more points over all games.

You are required to determine the number of different variants of the tournament's scenario, where the loser of one version is the winner of the other. Two variants of the tournament's scenario are considered as different if at least one of the participants in any game has gained a different number of points.

Input

In a single line you are given 3 numbers N, K and p.

Limit

$1 \leq N \leq 50$, $0 \leq K \leq 100$, $1 \leq p \leq 10^9$.

Output

In a single line output the number of possible scenario of the tournament modulo p.

Examples

d.in	d.out
`3 3 100`	`54`
`2 4 200`	`0`

Problem E. Longest Geometric Progression

Input file:	`e.in`
Output file:	`e.out`
Time limit:	1 sec
Memory limit:	256 Mb

The sequence $\{a_0, a_1, \ldots, a_n\}$ is called *geometric progression* if $a_i^2 = a_{i-1}a_{i+1}$ for all $i \in \{1, 2, \ldots, n-1\}$. Note that the sequence composed of one or two numbers is always a geometric progression. You are given positive integers L and R such that $L < R$. Find the longest geometric progression $\{a_0, a_1, \ldots, a_n\}$ composed of positive integers such that

$$L \le a_0 < a_1 < \ldots < a_n \le R.$$

Input

The first line contains a positive integer $T \le 10000$, the number of test cases. Each of the next T lines contains two space separated positive integer L and R. It is guaranteed that $1 \le L < R \le 10^{18}$.

Output

For each pair (L, R) from the input file output the maximal possible n such that there exists a sequence $\{a_0, a_1, \ldots, a_n\}$ composed of positive integers such that $a_i^2 = a_{i-1}a_{i+1}$ for all $i \in \{1, 2, \ldots, n-1\}$ and $L \le a_0 < a_1 < \ldots < a_n \le R$, followed by numbers a_0 and a_1. (Note that $L < R$. Hence the sequence $\{L, R\}$ always satisfies these conditions. So n will be always positive). If there several such sequences for given n you can output any.

Examples

e.in	e.out
7 1 4 4 9 10 20 100 101 100 1000 999 1333 1 1000000	2 1 2 2 4 6 1 10 11 1 100 101 5 128 192 3 1000 1100 19 1 2