Flower

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

In front of Yuki is a flower with n petals.

She will perform several rounds of operations. In each round, she will first pick a petals off the flower, and then pick off another b petals. If the remaining petals are inadequate, she plucks all. She doesn't stop until no petals remain.

Yuki once promised that she would leave if and only if the last petal she picked belonged to the first a petals picked in some round of the operation; otherwise, she would stay. Sympathetic, you want to pick off some number of petals first, **but you cannot pick all**, to ensure that she stays. You must determine the minimum number of petals you need to pick off; in particular, if she can't stay, output "`Sayonara`".

Input

Each test contains multiple test cases. The first line of input contains a single integer t ($1 \le t \le 100$) — the number of test cases. The description of the test cases follows.

The first and only line of input of each test case contains three integers n, a, and b ($1 \le n \le 10^9$, $1 \le a \le 10^9$, $1 \le b \le 10^9$).

Output

For each test case, output a single line:

- If you can make her stay, output an integer as the minimum number of petals you need to pick off;
- If you cannot, output the string "`Sayonara`".

Example

standard input	standard output
3 1 2 3 10 2 3 11 2 3	Sayonara 0 1

Note

In the first test case, $n = 1$, $a = 2$, $b = 3$. You cannot pick any petals, and Yuki will pick all the petals in the first round, thus leaving. Therefore, the output is "`Sayonara`".

In the second test case, it is easy to see that even if no petals are picked off initially, Yuki will end up picking all the petals after two rounds of operations, thus staying, so the answer is 0.