# Save the Prisoner!

A jail has a number of prisoners and a number of treats to pass out to them. Their jailer decides the fairest way to divide the treats is to seat the prisoners around a circular table in sequentially numbered chairs. A chair number will be drawn from a hat. Beginning with the prisoner in that chair, one candy will be handed to each prisoner sequentially around the table until all have been distributed.

The jailer is playing a little joke, though. The last piece of candy looks like all the others, but it tastes *awful*. Determine the chair number occupied by the prisoner who will receive that candy.

**Example**

$n = 4$
$m = 6$
$s = 2$

There are $4$ prisoners, $6$ pieces of candy and distribution starts at chair $2$. The prisoners arrange themselves in seats numbered $1$ to $4$. Prisoners receive candy at positions $2, 3, 4, 1, 2, 3$. The prisoner to be warned sits in chair number $3$.

**Function Description**

Complete the *saveThePrisoner* function in the editor below. It should return an integer representing the chair number of the prisoner to warn.

saveThePrisoner has the following parameter(s):

- *int n*: the number of prisoners
- *int m*: the number of sweets
- *int s*: the chair number to begin passing out sweets from

**Returns**

- *int:* the chair number of the prisoner to warn

**Input Format**

The first line contains an integer, $t$, the number of test cases.
The next $t$ lines each contain $3$ space-separated integers:

- $n$: the number of prisoners
- $m$: the number of sweets
- $s$: the chair number to start passing out treats at

**Constraints**

- $1 \leq t \leq 100$
- $1 \leq n \leq 10^9$

- $1 \leq m \leq 10^9$
- $1 \leq s \leq n$

**Sample Input 0**

```
2
5 2 1
5 2 2
```

**Sample Output 0**

```
2
3
```

**Explanation 0**

In the first query, there are $n = 5$ prisoners and $m = 2$ sweets. Distribution starts at seat number $s = 1$. Prisoners in seats numbered $1$ and $2$ get sweets. Warn prisoner $2$.

In the second query, distribution starts at seat $2$ so prisoners in seats $2$ and $3$ get sweets. Warn prisoner $3$.

**Sample Input 1**

```
2
7 19 2
3 7 3
```

**Sample Output 1**

```
6
3
```

**Explanation 1**

In the first test case, there are $n = 7$ prisoners, $m = 19$ sweets and they are passed out starting at chair $s = 2$. The candies go all around twice and there are $5$ more candies passed to each prisoner from seat $2$ to seat $6$.

In the second test case, there are $n = 3$ prisoners, $m = 7$ candies and they are passed out starting at seat $s = 3$. They go around twice, and there is one more to pass out to the prisoner at seat $3$.