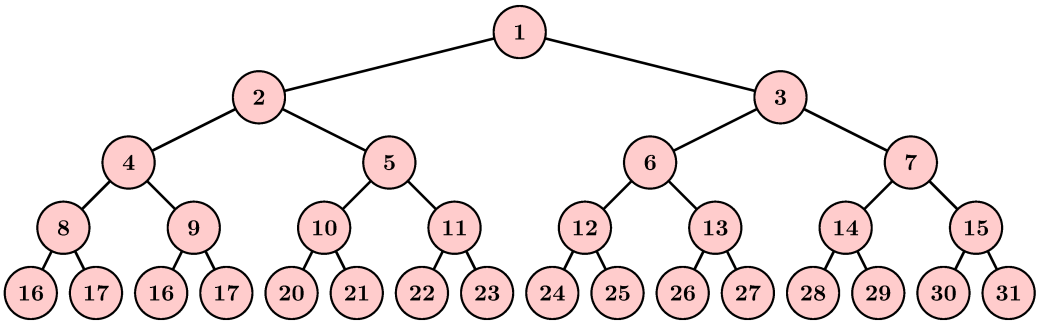
# Problem L
## Left Or Right

Given a complete binary tree of $2^{30} - 1$ vertices, where vertices are numbered from 1 to $2^{30} - 1$. The root is vertex 1. Every vertex $i$ where $1 \le i \le 2^{29} - 1$ has exactly 2 children, the left one is vertex $2 \cdot i$ and the right one is vertex $2 \cdot i + 1$.

The below figure demonstrates 31 first vertices of this binary tree:




For every non-root vertex $n$ in this tree, the path from root to this vertex always consists of several "moving down" steps, in each step we move from the current vertex to either its left or right child. Therefore, the path can be represented by a string consisting of `L` and `R`, which mean moving to the left child and to the right child, respectively.

Given some vertex $n$, your task is to find the string representing the path from root to this vertex.

## Input

The input starts with a positive integer $t$ ($t \le 1000$) — the number of test cases. Then each test case is printed in a single line with a single integer $n$ ($2 \le n \le 2^{30} - 1$).

## Output

For each test case, print a string representing the path from root to vertex $n$.

| Sample Input 1 | Sample Output 1 |
|---|---|
| 2<br>3<br>4 | R<br>LL |