Duplication



Consider a binary string, s, with an initial value of "0". We expand s by performing the following steps:

- 1. Create a string, t, where each character t[i] is equal to 1 s[i]. For example, if s = "01", then t = "10". Note that t and s always have the same length because t is the complement of s.
- 2. Append t to the end of s so that $s_{expanded} = s_{initial} + t_{s_{initial}}$. In the example above, s becomes "0110".
- 3. We keep on expanding s using steps 1 and 2 until the length of s exceeds 1000.

When we repeat the expansion operation, string \boldsymbol{s} grows like this:

Given q queries in the form of a zero-based index, x, solve each query by printing the character at index x in s on a new line.

Input Format

The first line contains an integer denoting q (number of queries).

Each of the q subsequent lines contains an integer describing the value of x for a query.

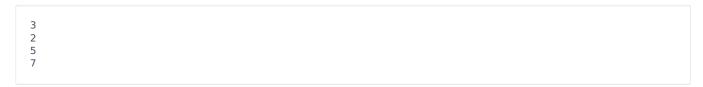
Constraints

•
$$0 < x, q < 10^3$$

Output Format

For each query, print the value of s[x] (i.e., either 0 or 1) on a new line.

Sample Input 0



Sample Output 0

```
1
0
1
```

Explanation 0

First, we build string s= "0110100110010...". Next, we answer the following sequence of q=3 queries:

- 1. For x=2, s[2]=1 so we print 1 on a new line.
- 2. For $\pmb{x}=\pmb{5}$, $\pmb{s}[\pmb{5}]=\pmb{0}$ so we print $\pmb{0}$ on a new line.
- 3. For x = 7, s[7] = 1 so we print 1 on a new line.