F. Meaningless Operations

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Can the greatest common divisor and bitwise operations have anything in common? It is time to answer this question.

Suppose you are given a positive integer a. You want to choose some integer b from 1 to $a - 1$ inclusive in such a way that the greatest common divisor (GCD) of integers $a \oplus b$ and $a \,\&\, b$ is as large as possible. In other words, you'd like to compute the following function:

$$f(a) = \max_{0<b<a} gcd(a \oplus b, a \,\&\, b).$$

Here $\oplus$ denotes the bitwise XOR operation, and $\&$ denotes the bitwise AND operation.

The greatest common divisor of two integers x and y is the largest integer g such that both x and y are divided by g without remainder.

You are given q integers $a_1, a_2, \ldots, a_q$. For each of these integers compute the largest possible value of the greatest common divisor (when b is chosen optimally).

Input

The first line contains an integer q ($1 \le q \le 10^3$) — the number of integers you need to compute the answer for.

After that q integers are given, one per line: $a_1, a_2, \ldots, a_q$ ($2 \le a_i \le 2^{25} - 1$) — the integers you need to compute the answer for.

Output

For each integer, print the answer in the same order as the integers are given in input.

Example

input

```
3
2
3
5
```

output

```
3
1
7
```

Note

For the first integer the optimal choice is $b = 1$, then $a \oplus b = 3$, $a \,\&\, b = 0$, and the greatest common divisor of 3 and 0 is 3.

For the second integer one optimal choice is $b = 2$, then $a \oplus b = 1$, $a \,\&\, b = 2$, and the greatest common divisor of 1 and 2 is 1.

For the third integer the optimal choice is $b = 2$, then $a \oplus b = 7$, $a \,\&\, b = 0$, and the greatest common divisor of 7 and 0 is 7.