**Make Number Odd**

[math](http://www.practice.geeksforgeeks.org/tag-page.php?tag=math&isCmp=0)

Given a number n. Print the minimum positive integer by which it should be divided so that the result is an odd number.

**Input:**

First line of the input will contain the number of test cases.

Each test case will contain a single integer n

**Output:**  
Print the required output.

**Constraints:**

1 <= T <= 100

1 <= N <= 100000  
**Example:**

**Input:**  
3  
2  
5  
8

**Output:**  
2  
1  
8

**Explanation:**

In the first case, 2 must be divided by 2 to make it 1(i.e an odd number).

In the second case, 5 is already odd.

In the third case, 8 must be divided by 8 to make it an odd number.

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=548>

**import** java.util.\*;

**import** java.lang.\*;

**import** java.io.\*;

**class** GFG {

**public** **static** **void** main(String[] args) {

*// TODO code application logic here*

        Scanner sc = **new** Scanner(System.in);

**int** t=Integer.parseInt(sc.nextLine());

**for** (**int** i = 0; i < t; i++)

        {

**int** n = Integer.parseInt(sc.nextLine());

**int** ans = -1;

**for** (**int** j = 1; j <= n; j++)

            {

**float** res = (**float**)n / (**float**)j;

                String rs = String.valueOf(res);

**boolean** decimales = **false**;

**int** ipunto = rs.indexOf('.');

**if** (ipunto != -1)

                {

**for** (**int** k = ipunto + 1; k < rs.length(); k++)

                    {

**if** (rs.charAt(k) != '0')

                        {

                            decimales = **true**;

                        }

                    }

                }

**if** (!decimales)

                {

**if** ( (**int**)res % 2 != 0)

                    {

                        ans = j;

**break**;

                    }

                }

            }

            System.out.println(ans);

        }

    }

}