# Totient in permutation (medium)

In number theory, Euler's totient (or PHI function), is an arithmetic function that counts the number of positive integers less than or equal to a positive integer N that are relatively prime to this number N.

That is, if N is a positive integer, then PHI(N) is the number of integers K for which GCD(N, K) = 1 and 1 ≤ K ≤ N. We denote GCD the Greatest Common Divisor. For example, we have PHI(9)=6.

Interestingly, PHI(87109)=79180, and it can be seen that 87109 is a permutation of 79180.

## Input

The input begins with the number T of test cases in a single line.
In each of the next T lines there are an integer M.

## Output

For each given M, you have to print on a single line the value of N, for which 1 < N < M, PHI(N) is a permutation of N and the ratio N/PHI(N) produces a minimum. If there's several answers output the greatest, or if need, "No solution." without quotes.
Leading zeros are not allowed for integers greater than 0.

## Example

**Input:**
3
22
222
2222

**Output:**
21
63
291

**Explanations :** For the first case, in the range ]1..22[, the lonely number n for witch phi(n) is in permutations(n) is 21, (we have phi(21)=12). So the answer is obviously 21.
For the second case, in the range ]1..222[, there's two numbers n for witch phi(n) is in permutations(n), we have phi(21)=12 and phi(63)=36. But as 63/36 is equal to 21/12, we're taking the greater : 63.
For the third case, in the range ]1..2222[, there's four numbers n for witch phi(n) is in permutations(n), phi(21)=12, phi(63)=36, phi(291)=192 and phi(502)=250. Within those solutions 291/192 is the minimum, we output 291.

## Constraints

1 < T < 10^2
1 < M < 10^12

Code size limit is 10kB ; the upper bound was set at 10^12 to make a (C/pascal/...)-solution

easier to write. Constraints allow Python3 users to get AC under 1.86s (with a sub-optimal solution). (Edit 2017-02-11, after compiler updates)
If if you get TLE, you should try first TIP1.
If it's too easy for you TIP3 is made for you ;-)