I Repeat Myself I Repeat Myself I Repeat

Problem ID: irepeatmyself
CPU Time limit: 1 second
Memory limit: 1024 MB
Difficulty: 2.4

Author: David Sturgill
Source: Baylor Competitive Learning course

The Perl programming language has a lot of convenient little operators. For example, it has an infix operator, x, for creating repeated copies of a string. When used in an expression like p x n, the operator x produces a string containing n repeated copies of the string p.

For this problem, you are going to look for cases where a long input string consists of a repeated pattern. We say string s_1 is a *prefix* of string s if there exists some (possibly empty) string s_2 such that s is the concatenation of s_1 and s_2. We say pattern p *explains* string s if s is a prefix of p x n for some sufficiently large n.

Input

Input starts with an integer, $1 \leq n \leq 200$. This is followed by n test cases, one per line. Each input line consists of a non-empty sequence of up to 70 printable ASCII characters.

Output

For every test case, print a single output line giving the length of the shortest pattern that explains the given input string.

Sample Input 1

```
3
I Repeat Myself I Repeat Myself I Repeat
aaaaaaaaaaaaaaaaaaaa
abbcabbcabbabbcabb
```

Sample Output 1

```
16
1
11
```