

Prof. Dr. Friedrich, Dr. Lenzner, Boockmeyer, Neumann, Stangl  
Sommersemester 2017

## Woche 02 – (Adv.) Competitive Programming

Abgabe 01.05.2017 17:00 Uhr, über das Judge-Interface

### Aufgabe 1 (palindrome-e,h). (100 Points)

A palindrome is a string of letters that is the same forward as backward. For example, the string “ABCDEDCBA” is a palindrome because it is the same when the string is read from left to right as when the string is read from right to left.

Your task is to find the longest palindrome in a string. The palindrome can be any substring of a given string. For this task, we define that a palindrome must have at least 2 letters. That means, a single letter isn’t a palindrome.

**Input** The input will begin with  $n$  ( $1 \leq n \leq 2500$ ), the number of strings. Each string has a length between 1 and  $l$  ( $1 \leq l \leq 3000$ ).

**Output** Print for each string the length of the longest palindrome in that string. If there is no palindrome in the string, please print  $-1$ .

**Points** There are three groups of test sets:

- *e(asy)*: For the second group worth 50 Points, you can assume, that  $n \leq 600$  and  $l \leq 800$ .
- *h(ard)*: For the third group of test sets worth 50 Points, there are no additional assumptions.
- *b(onus)*: For the bonus test cases worth 20 Points, you can assume, that  $n \leq 4000$  and  $l \leq 4000$ .

### Sample Input

```
4
otto
anas
abcotto
ottoaotto
```

### Sample Output

```
4
5
4
9
```