Longest Palindromic Path

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Given a string s of length n, for every palindromic substring of s we construct a node representing it. For every palindromic substring a and b, if a is a substring of b ($a \neq b$), we add an edge from b to a. It is obvious that the graph is a DAG(Directed Acyclic Graph), just find the longest path in the graph. Note that a palindromic string is a string s which satisfies $s_i = s_{n-i+1}$ for all $1 \leq i \leq n$.

Input

The first line of the input contains an integer T, representing the number of test cases. Each of the next $T(1 \leq T \leq 5)$ lines contains a string $s(1 \leq |s| \leq 10^6)$.

Output

For each test case output a number representing the answer in a separate line.

Example

standard input	standard output
3 baaa aaaabbbb abcba	3 4 3