

Zadanie. Naszyjnik

W oryginalnym zadaniu był tu literacki wstęp o gwiazdzie (sceny), która uwielbiała naszyjniki i zamówiła u swojego „Inspector of Bead Makers” naszyjnik spełniający wyśrubowane kryteria. Pomińmy tę część i przejdźmy do konkretów. Opiszem naszyjnika jest łańcuch małych znaków alfabetu łacińskiego $A = a_1a_2...a_m$, gdzie kolejne znaki odpowiadają wielkości kolejnego koralika, a znak (koralik) a_m poprzedza oczywiście, ostatecznie to zamknięty naszyjnik, znak a_1 . Problem z takim naszyjnikiem polega na tym, że może on ulec przerwaniu, a miejscem najsłabszym jest to, w którym zaczynający się łańcuch znaków jest leksykograficznie najmniejszy. Przypomnijmy, że łańcuch $a_1a_2...a_n$ jest leksykograficznie mniejszy od łańcucha $b_1b_2...b_n$ wtedy i tylko wtedy, gdy $\exists_{1 \leq i \leq n} \forall_{j \leq i} a_j = b_j, a_i < b_i$. Należy oczywiście znaleźć pozycję, na której połączenie jest najsłabsze.

Wejście

Pierwsza linia wejścia zawiera liczbę zestawów danych N . Kolejne N linii zawiera opisy naszyjników, składające się z co najwyżej 10000 znaków – małych liter alfabetu łacińskiego.

Wyjście

Dla każdego zestawu powinna zostać wypisana, w osobnej linii, pozycja najsłabszego połączenia. Przypominam, indeksacja zaczyna się od 1.

Przykład

Dla danych wejściowych:	Poprawną odpowiedzią jest:
4	10
helloworld	11
amandamanda	6
dontcallmebfu	5
aaabaaa	

Plik zawierający rozwiązanie powinien nosić nazwę utworzoną według schematu:

ASD2_[NazwiskoI]_naszyjnik.cpp (lub ASD2z_[NazwiskoI]_naszyjnik.cpp)