

Liczba testów: 3 + 5 + 5

Dostępna pamięć: 64 MB

Dla zadanej gramatyki bezkontekstowej $\mathcal{G}$ i słowa w określ, czy dane słowo w da się wyprowadzić z tej gramatyki. Gramatyka składa się z 8 symboli nieterminalnych oznaczanych dużymi literami A, B, C, D, E, F, G i H ; 26 symboli terminalnych oznaczanych małymi literami alfabetu angielskiego oraz zbioru produkcji. Symbolem startowym gramatyki będzie zawsze symbol nieterminalny A . Przy takich założeniach gramatyka jest jednoznacznie definiowana przez podanie zbioru produkcji. Gramatyka jest podana w postaci normalnej Chomskiego, tzn. każda z produkcji jest w jednej z dwóch postaci:

1. $B \rightarrow CD$ (produkcja typu I),
2. $C \rightarrow g$ (produkcja typu II).

W produkcji typu I z jednego symbolu nieterminalnego powstają dwa symbole nieterminalne, a w produkcji typu II z jednego symbolu nieterminalnego powstaje jeden symbol terminalny. W produkcjach mogą nie występować wszystkie symbole nieterminalne czy terminalne.

Przykładowo dla gramatyki $\{A \rightarrow BC, A \rightarrow AC, B \rightarrow b, C \rightarrow c\}$ słowo $bccc$ da się wyprowadzić na przykład w ten sposób:

$$A \rightarrow AC \rightarrow ACC \rightarrow BCCC \rightarrow bCCC \rightarrow bcCC \rightarrow bccC \rightarrow bccc .$$

Natomiast z gramatyki $\{A \rightarrow BC, A \rightarrow AA, A \rightarrow b, B \rightarrow b, C \rightarrow c\}$ słowa $bccc$ nie da się wyprowadzić.

Wiele instancji problemu w jednym teście

Zauważmy, że program, który zawsze odpowiada „TAK” (lub program który zawsze odpowiada „NIE”) udzieliłby wielu poprawnych odpowiedzi. Dlatego też dane wejściowe są pogrupowane; pojedynczy test zawiera wiele instancji problemu opisanego powyżej. W pierwszym wierszu wejścia podana jest jedna liczba naturalna D ($1 \leq D \leq 10$) oznaczająca liczbę podanych instancji. Każda z instancji jest zgodna ze specyfikacją określoną w części „Specyfikacja pojedynczej instancji”.

Specyfikacja pojedynczej instancji

Opis gramatyki $\mathcal{G}$ jest podany w następujący sposób. W pierwszym wierszu znajdują się dwie całkowite, nieujemne liczby m_1 i m_2 oddzielone pojedynczym odstępem, oznaczające odpowiednio liczbę produkcji typu I i typu II. W każdym z kolejnych m_1 wierszy znajdują się trzy duże litery alfabetu angielskiego oddzielone pojedynczą spacją, będące opisem jednej produkcji typu I. Przykładowo „A D C” oznacza produkcję $A \rightarrow DC$. W każdym z kolejnych m_2 wierszy znajduje się para liter: duża i mała litera alfabetu angielskiego oddzielone pojedynczą spacją, będące opisem jednej produkcji typu II. Przykładowo „B c” oznacza produkcję $B \rightarrow c$. Produkcje nie powtarzają się.

W ostatnim wierszu instancji (wierszu o numerze $m_1 + m_2 + 2$) zapisane jest niepuste słowo w o długości co najwyżej 500, składające się z małych liter alfabetu angielskiego.

Specyfikacja danych wyjściowych

Twój program powinien wypisać D wierszy. W i -tym wierszu powinno znaleźć się słowo „TAK”, jeśli w dla zadanej w i -tej instancji gramatyki $\mathcal{G}$ z symbolu startowego A da się wyprowadzić zadane słowo w , zaś słowo „NIE” w przeciwnym przypadku.

Przykład A

Wejście:

```
1
2 2
A B C
A A C
B b
C c
bccc
```

Wyjście:

```
TAK
```

Przykład B

Wejście:

2
0 1
A b
c
2 3
A B C
A A A
A b
B b
C c
bccc

Wyjście:

NIE
NIE

Przykład C

Wejście:

2
1 1
A B C
H a
aaaa
3 2
A B C
D E F
B B B
B b
C c
bbbbc

Wyjście:

NIE
TAK