

---

## DFA

P46692\_es

Final OIE 2010 (2010)

---

Un DFA (Deterministic Finite Automaton; Autómata Finito Determinista) es una especie de máquina que sirve para decidir qué secuencias de caracteres (palabras) de un cierto alfabeto son válidas o no. Por ejemplo, esto es un DFA que sirve para reconocer ciertas palabras del alfabeto  $\{A, B, C\}$ :

FALTA DIBUJO!!!

¿Cómo hay que interpretar este diagrama?

- Los círculos son los *estados* del autómata.
- El 0 es el *estado inicial*. Los estados con un círculo grueso (sólo uno, en el ejemplo) son los *estados aceptadores*.
- Desde cada estado sale exactamente una flecha para cada letra del alfabeto.
- El autómata cambia de estado siguiendo las flechas según indiquen las letras de la palabra, y empezando por el estado inicial.
- Si después de procesar la palabra el autómata acaba en un estado aceptador, responde Ok. Si acaba en un estado que no es aceptador, responde No.

Por ejemplo: consideremos el autómata de la figura y la palabra ABBBBABBC. Este empieza en el estado 0. A continuación, sigue la flecha de la A que sale del 0, y llega al estado 1. Luego sigue la de la B (desde el estado actual, el 1), y llega al estado 2. Si seguimos, veremos que la secuencia total de estados es 0, 1, 2, 3, 0, 1, 2, 0. Como el estado 0 no es aceptador (no tiene un círculo grueso), el DFA responde No. En cambio, si hubiéramos tenido la palabra ABABBAC, la secuencia de estados hubiera sido 0, 1, 2, 1, 2, 3, 4, 4, y hubiera respondido Ok.

Te pedimos que escribas un programa que simule la ejecución de un DFA para decidir si ciertas palabras dadas son o no son válidas.

### Entrada

Una entrada consiste en la descripción de un DFA, seguido de las palabras que hay que validar. En una primera línea, se dan, separados por espacios, el número  $1 \leq n \leq 1000$  de estados del autómata (se numeran del 0 al  $n - 1$ ) y el número  $2 \leq k \leq 26$  de letras mayúsculas consecutivas usadas del alfabeto (empezando siempre por la A). A continuación,  $n$  líneas, cada una de las cuales contiene la información de un estado: un carácter A o N para indicar si el estado es aceptador o no, y  $k$  números de 0 a  $n - 1$  para indicar los destinos de las  $k$  flechas que salen del estado al recibir las  $k$  letras A, B, C, ... (mira el primer ejemplo para ver cómo se representa el autómata de la figura).

A continuación, y después de una línea en blanco, se da el número  $t \leq 100$  de palabras que hay que validar, y  $t$  líneas con las palabras, formadas únicamente por letras del alfabeto elegido. Un guión (-) indica la palabra vacía (aquella palabra que no tiene ninguna letra). Ninguna palabra tendrá más de 1000 letras.

## Salida

Escribe una línea para cada una de las  $t$  palabras del caso de prueba, con `Ok` si la palabra es aceptada por el autómata, o `No` si no lo es.

## Puntuación

- **Test-1:**

**40 Puntos**

Resolver entradas donde  $n \leq 3$ ,  $k = 2$ , y ninguna palabra es vacía ni tiene más de 10 caracteres.

- **Test-2:**

**60 Puntos**

Resolver entradas de todo tipo.

### Ejemplo de entrada 1

5 3  
N 1 0 0  
N 1 2 0  
N 1 3 0  
N 4 0 0  
A 4 4 4  
12  
ABBBABBC  
ABABBAC  
ABB  
ABBBA  
ABABABABA  
ACCA  
ABAB  
ABBA  
ABACABBAC  
CCCABBABBACCC  
ABCABABBACBA  
ABBACCC

### Ejemplo de salida 1

No  
Ok  
No  
No  
No  
No  
No  
Ok  
Ok  
Ok  
Ok  
Ok  
Ok  
Ok  
Ok  
Ok  
Ok  
Ok  
Ok  
Ok

### Ejemplo de entrada 2

2 2  
A 0 1  
N 1 0  
11  
-  
A  
B  
AA  
AB  
BA  
BB  
AAA  
ABA  
ABABABBBAAABABA  
ABABABBBAAABABAB

### Ejemplo de salida 2

Ok  
Ok  
No  
Ok  
No  
No  
Ok  
Ok  
No  
No  
Ok

## Información del problema

Autor : Omer Giménez

Generación : 2024-05-02 18:06:00

© *Jutge.org*, 2006–2024.  
<https://jutge.org>