

**POLITÉCNICO GRANCOLOMBIANO**  
**Departamento de Ingeniería de Sistemas y Telecomunicaciones**  
**Autómatas, Gramáticas y Lenguajes**  
**Proyecto de Aula**

Como primera parte del proyecto de aula se requiere solucionar el siguiente problema:

## Problema

Un autómata finito determinista AFD es una quintupla  $M = (Q, \Sigma, q_0, F, \delta)$  donde:

1.  $Q = \{q_0, q_1, \dots, q_{n-1}\}$  es un conjunto finito no vacío, denominado *conjunto de estados*.
2.  $\Sigma$  es un conjunto finito de símbolos denominado *alfabeto*.
3.  $q_0 \in Q$  denominado *estado inicial*.
4.  $F : F \subseteq Q$  denominado conjunto de estados finales o de aceptación.
5.  $\delta : Q \times \Sigma \rightarrow Q$  función de transición del autómata, a partir de un estado y un símbolo del alfabeto obtiene un estado.

Escriba un programa en **Java** que dada la descripción completa de un autómata permita simular el procesamiento llevado a cabo por éste a un conjunto de cadenas dadas.

## Entrada

La entrada consiste de la descripción completa de un autómata  $M$ : un alfabeto  $\Sigma$ , un conjunto de estados  $Q$ , el estado inicial  $q_0$ , un conjunto no vacío de estados finales  $F$  y una función de transición  $\delta$ , además una descripción del lenguaje que éste acepta.

La primera línea de la entrada contiene la descripción del lenguaje que acepta el autómata.

La segunda línea de la entrada contiene una cadena formada por los símbolos del alfabeto  $\Sigma$ , cada carácter en la cadena no aparece en ella más de una vez.

La tercera línea de la entrada contiene los elementos del conjunto de estados  $Q$  separados por coma (,). Cada estado está representado por medio una cadena de texto en la cual cada uno de sus caracteres puede ser una de las 26 letras el alfabeto inglés, un dígito o el carácter espacio. Este último carácter no puede ser el carácter inicial o final de la cadena.

La siguiente línea contiene el estado inicial  $q_0$ , seguida de otra línea que contiene los elementos del conjunto de estados finales  $F$  separados por coma (,).

La sexta línea contiene un número entero  $n$  el cual es el número de transiciones, en seguida vienen  $n$  líneas en donde cada línea representa una de las transiciones y cada línea contiene un estado  $q_i$ , un símbolo  $a$  y otro estado  $q_j$  ( $q_i, q_j \in Q$  y  $a \in \Sigma$ ).

La siguiente línea contiene el número de cadenas  $C$  a ser procesadas por el autómata, en seguida  $C$  líneas cada una con una cadena de texto  $u \in \Sigma^*$ .

*La entrada debe ser leída de la entrada estándar (System.in).*

## Salida

En las primeras líneas de la salida se debe imprimir la descripción del lenguaje aceptado por el autómata descrito en la entrada seguido del número de cadenas a procesar en el siguiente formato:

Lenguaje: <descripción-del-lenguaje> Cadenas a procesar: <número-de-cadenas>

En seguida por cada cadena  $u$  a procesar se debe imprimir  $l + 1$  líneas donde  $l$  es la longitud de la cadena ( $l = |u|$ ). Cada  $i$ -ésima línea de las  $l$  líneas debe contener  $q_p$ ,  $a_i$  y  $q_s$  los cuales son estado previo a la lectura del  $i$ -ésimo símbolo  $a_i$  por el autómata,  $i$ -ésimo símbolo de la cadena y el estado al que pasará el autómata luego de leer el  $i$ -ésimo símbolo en el siguiente formato:

(<estado-previo>,<iesimo-simbolo>) => (<estado-siguiente>)

La última línea de salida para cada una de las cadenas a ser procesadas por el autómata debe contener "si" en el caso en que la cadena pertenezca al lenguaje aceptado por el autómata y "no." en el caso contrario.

Se debe imprimir una línea en blanco entre cada uno de los procesamientos de cadenas, ver ejemplos de entrada y salida.

*La salida debe ser escrita a la salida estándar (System.out).*

Ejemplo de Entrada	Ejemplo de Salida
Cadenas binarias de longitud par 01 par,impar par par 4 par,0,impar par,1,impar impar,0,par impar,1,par 5  0 1 01 10	Lenguaje L: Cadenas binarias de longitud par  Procesando cadena 1: "" La cadena "" pertenece al lenguaje L.  Procesando cadena 2: "0" (par, 0) => impar La cadena "0" no pertenece al lenguaje L.  Procesando cadena 3: "1" (par, 1) => par La cadena "1" no pertenece al lenguaje L.  Procesando cadena 4: "01" (par, 0) => impar (impar, 1) => par La cadena "01" pertenece al lenguaje L.  Procesando cadena 5: "10" (par, 1) => impar (impar, 0) => par La cadena "10" pertenece al lenguaje L.