

# ANALISI LESSICALE

Ha scopo di riconoscere SEQUENZE DI CARATTERI che rappresentano elementi atomici del programma

Tali sequenze sono dette **TOKEN** o **LESSMI**

## AUTOMA A STATI FINITI

Macchine che riconosce stringhe con memoria finita

INPUT  $\rightarrow$  stringa

OUTPUT  $\rightarrow$  sì/no

Ha una visione **LOCALE** e **LIMITATA**

Ogni simbolo letto altera lo stato

La risposta "sì" corrisponde ad uno **STATO FINALE**

DEFINIZIONE:

DTA (Deterministic Finite-state Automata) è una QUINTUPLA

$$A = (Q, \Sigma, \delta, q_0, F)$$

- **Q** l'insieme finito di stati
- **$\Sigma$**  l'alfabeto riconosciuto dall'automa
- **$\delta$** :  $Q \times \Sigma \rightarrow Q$  è la funzione di transizione
- **$q_0 \in Q$**  è lo stato iniziale
- **$F \subseteq Q$**  è l'insieme di stati finali

## LINGUAGGIO RICONOSCIUTO DA UN DFA

La funzione di transizione estesa dell'automata  $A$  è la funzione:  
 $\hat{\delta}: Q \times \Sigma^* \rightarrow Q$  definita per induzione sul 2° secondo argomento  
come:

$$\hat{\delta}(q, \varepsilon) = q \quad \hat{\delta}(q, wa) = \delta(\hat{\delta}(q, w), a)$$

Il linguaggio riconosciuto/accettato dall'automata  $A = (Q, \Sigma, \delta, q_0, F)$  è  
denotato da  $L(A)$  e definito come:

$$L(A) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid \hat{\delta}(q_0, w) \in F\}$$

Un linguaggio si dice regolare se esiste un automata  $A$  tale che  $L = L(A)$

## TABELLE DI TRANSIZIONE

STATO	B
$\rightarrow q_0$	$q_1$
$* q_1$	$q_0$

le righe corrispondono agli stati

le colonne corrispondono ai simboli dell'alfabeto