07/04/2025 → Incontro 5

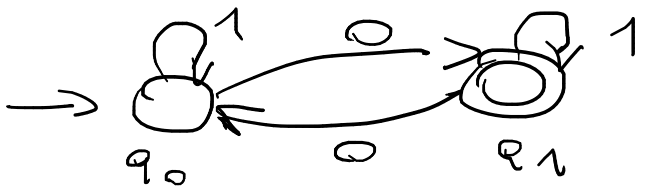
- CFG ——▷ PARSING
- CNF

- VARIABILI
- REGOLE DI SOSTITUZIONE (PRODUZIONI)
- TERMINALI
- VARIABILE INIZIALE

$G = (V, \Sigma, R, S)$

DFA ⇒ CFG (CHIUSURA)

$\{w \in \{0,1\}^* \mid$ w CONTIENE UN N. DISPARI DI $\emptyset\}$



$R_0 \rightarrow 1R_0 \mid 0R_1$

$R_1 \rightarrow 0R_0 \mid 1R_1 \mid \varepsilon$

$L = \{w \in \{0,1\}^* \mid$ w CONTIENE PIÙ 1 RISPETTO AGLI $\emptyset\}$

$S \rightarrow 0S1 \mid 1S0 \mid 1S1 \mid 1S \mid 1$

ALTERNATIVA

$S \rightarrow S_1 1 S_1$

$S_1 \rightarrow 0S_1 1 \mid 1S_1 0 \mid S_1 S_1 \mid 1S_1 \mid \varepsilon$

$S \rightarrow \underline{1S} \mid 0S \mid 1$

$$L_1 = \{ a^i b^j c^k \mid i + j = k,\ i, j, k \geq 1 \}$$

$w \in \{a, b, c\}^*$

$$S \to abcc \mid aSc \mid bSc \mid \underbrace{\ldots}_{\text{PERMUTAZIONI}}$$

——> ALTERNATIVA

$$S \to aSc \mid T$$

$$T \to bTc \mid bc$$

LINGUAGGIO (NON REGOLARE) (L) $\to \underbrace{a^n b^n \mid n \geq 0}_{L}$ $\to$ n UGUALI

$\to$ COMPLEMENTO $\quad \bar{L}$ = COMPLEMENTO DI L

↑ ESPONENTI DI a e b DIVERSI

- DIMOSTRA $\bar{L}$ COME CONTEXT-FREE

- $S_1 \to aS_1b \mid aS_1 \mid a$ $\quad (\#a > \#b)$ NUMERO

$(\#b > \#a)$ - $S_2 \to aS_2b \mid S_2b \mid b$

$S_3 \to XbXaX$

$X \to aX \mid bX \mid \varepsilon$

- $S \to S_1 \mid S_2 \mid S_3$ $\quad$ - $S_3 \to$ UNIONE

GRAMMATICHE AMBIGUE (LEFTMOST DERIVATION) $\to$ LINGUAGGIO

axa

DERIV. 1

axa

DERIV. 2

ESERCIZIO :

LIBRO/ ADEMOODLE

G = (V, Σ, R, <STMT>)

- <STMT> → <ASSIGN> | <IF-THEN> | <IF-THEN-ELSE>
- <IF-THEN> → if condition then <STMT>
- <IF-THEN-ELSE> → if condition then <STMT> else <STMT>
- <ASSIGN> → a := 1

⇒ MOSTRA G COME AMBIGUA

<STMT> → <IF-THEN> | <IF-THEN-ELSE>

<IF-THEN> → <IF-THEN> | <ASSIGN>

<IF-THEN-ELSE> → <IF-THEN> | <ASSIGN>

(PRODUZIONI INTERMEDIE CHE CAMBIANO)

IF CONDITION THEN IF CONDITION ELSE a := 1

→ DIAMO UNA GRAMMATICA H NON AMBIGUA

- <STMT> → <ASSIGN> | <I-T-E>
- <I-T-E> → if condition then <T> else <STMT> | T
- T → <I-T>
- <I-T> → if condition then <STMT>
- <ASSIGN> → a := 1

FORMA NORMALE DI CHOMSKY

$$\left[\begin{array}{l} A \to BC \\ A \to a \end{array}\right] \to$$

1. AGGIUNTA NUOVA VAR. INIZIALE
2. ε-REGOLE : $A \to \varepsilon$
   ELIMINAZIONE
3. REGOLE UNITARIE : $A \to B$
   ELIMINAZIONE
4. TRASFORMAZIONE DI TUTTE LE REGOLE NELLA FORMA CORRETTA

$$G \left[\begin{array}{l} S \to ASA \mid aB \\ A \to B \mid S \\ B \to b \mid \varepsilon \end{array}\right]$$

- TRASFORMA G IN FNC

(1) AGGIUNTA $\Rightarrow S_0$ = NUOVO STATO INIZIALE

$S_0 \to S$

$S \to ASA \mid aB$

$A \to B \mid S$

$B \to b \mid \varepsilon$

(2) ELIMINAZIONE ε-REGOLA : $B \to \varepsilon$

$S_0 \to S$

$S \to ASA \mid aB \mid a$

$A \to B \mid S \mid \varepsilon$

$B \to b \mid \varepsilon$

$\to$

$S_0 \to S$

$S \to ASA \mid aB \mid SA \mid AS \mid S$

$A \to B \mid S$

$B \to b$

③ RIMOZIONE REGOLE UNITARIE $S_0 \to S$, $S \to S$

$S_0 \to S \mid ASA \mid aB \mid SA \mid AS$

$S \to ASA \mid aB \mid a \mid SA \mid AS$

$A \to B \mid S$

$B \to b$

④ RIMOZIONE REGOLE UNITARIE : $A \to B$, $A \to S$

REGOLE COMPOSTE

$S_0 \to A\underbrace{SA} \mid aB \mid a \mid SA \mid AS$ NUOVA REGOLA

$S \to ASA \mid aB \mid a \mid SA \mid AS$

$A \to b \mid ASA \mid aB \mid a \mid SA \mid AS$

$B \to b$

$S_0 \to AA_1 \mid UB \mid a \mid SA \mid AS$

$S \to AA_1 \mid UB \mid a \mid SA \mid AS$

$A \to b \mid AA_1 \mid UB \mid a \mid SA \mid AS$

$U \to a$

$B \to b$

FNC $\to$ DIMOSTRA CHE

PREFIX $(L) = \{ u \mid uv \in L$ per qualche stringa $v \}$

SE $\underbrace{L}_{}$ è CF $\Rightarrow$ PREFIX è CF

G $\to$ G' (CF) $\Rightarrow$ STESSA QUADRUPLA, DIVERSE REGOLE

$G \mid \exists G'$ in FNC

G', $\forall v \in G$, $\exists v'$

- TUTTE LE REGOLE DI G, ANCHE REGOLE DI G'
- $\forall v \in G$, $V' \to V$, $V' \to \varepsilon$
- $\forall$ REGOLA $V \to AB$, $V' \to AB'$, $V \to A'$
- se S è VAR. INIZIALE DI G, S' VAR. INIZIALE DI G'

$\Sigma = \{0, 1, +, =\}$

$$ADD = \left\{ x = y + z \mid \begin{array}{l} x, y, z \text{ NUM. BINARI}, \\ x = \text{SOMMA DI } Y + Z \end{array} \right\}$$

DIMOSTRA CHE ADD NON È REGOLARE

$$\left[\begin{array}{c} w = xyz, \\ y \neq \varepsilon, y^i, i > 0 \\ |xy| \leq k \end{array} \right]$$

$1^k = 1^k + 0$

$$\left[\begin{array}{l} p, q > 0 \\ k = p + q \end{array} \right]$$

$x = 1^p$

$y = 1^q$

$z = 1^{k-p-q} + 0$

$w = xy^iz = xy^pz$

$= 1^p 1^q 1^{k-p-q} \quad 1^k + 0$

$= 1^k + 0$

Inserimento del ragionamento corretto (presente nelle soluzioni dell'homework 5) - riportato formalmente nella pagina successiva.

Per essere più corretti ancora, definiamo il linguaggio come:

ADD = {x#y#z | x, y, z ∈ {0,1}* sono numeri binari e val(x) + val(y) = val(z)}

dove val(b) indica il valore numerico della stringa binaria b.

- Applicazione del pumping lemma:

Supponiamo per assurdo che ADD sia regolare.
Sia k la costante del pumping lemma.
Scegliamo w = 1^k#0#1^k ∈ ADD, con |w| > k.
Per il pumping lemma, w può essere scomposto come w = xyz dove:

|xy| ≤ k
|y| > 0
xy^iz ∈ ADD per ogni i ≥ 0

- Analisi della scomposizione:

Poiché |xy| ≤ k, sia x che y devono essere contenuti interamente nella prima parte 1^k.
Sia y = 1^p dove p > 0.
Sia x = 1^q dove q ≥ 0 e p+q ≤ k.

- Dimostrazione della contraddizione:

Consideriamo xy^2z = 1^q(1^p)^2 1^(k-p-q)#0#1^k = 1^(k+p)#0#1^k
Ora val(1^(k+p)) + val(0) = 2^(k+p) - 1 + 0 = 2^(k+p) - 1
Ma val(1^k) = 2^k - 1
Poiché p > 0, abbiamo 2^(k+p) - 1 > 2^k - 1
Quindi val(x) + val(y) ≠ val(z) per i = 2, il che implica xy^2z ∉ ADD

- Conclusione:

Abbiamo trovato un i (i = 2) per cui xy^iz ∉ ADD, contraddicendo l'ipotesi che ADD sia regolare.
Pertanto, ADD non è un linguaggio regolare.