GRAMMATICA TIPO 0 $\rightarrow$ SENZA RESTRIZIONI (NON LIMITATE)

In generale le produzioni di esse si definiscono

$\alpha \rightarrow \beta \quad \alpha \in V^* \circ V_N \circ V^* \quad \beta \in V^*$

es. P:

$S \rightarrow aSa \mid aAb \mid aAa \mid \varepsilon$

$aAa \rightarrow a \mid \varepsilon$

$aaAb \rightarrow b \mid \varepsilon$

PRODUZIONI CHE APPARTENGONO A GRAMMATICHE DI TIPO 0

es.

$G = <\{a, b\}, \{S, A\}, P, S> \quad P \Rightarrow \quad S \rightarrow aAb$

$aA \rightarrow aaAb$

$A \rightarrow \varepsilon$

TIPO 0

$L = \{a^n b^n \mid n \geq 1\}$

$S \rightarrow aAb \rightarrow aa\cancel{A}\underbrace{bb}$

In generale i linguaggi generabili di tipo 0, generano il linguaggio $L = \{a^n b^n \mid n \geq 1\}$

GRAMMATICHE DI TIPO 1 (CONTESTUALI)

↓

AUTOMI LIMITATI LINEARMENTE

$$\alpha \to \gamma \quad \alpha \in V^* \circ V_N \circ V^* \quad \gamma \in V^+ \quad |\alpha| \leq |\gamma|$$

queste grammatiche ammettono qualunque grammatica che non riduca la lunghezza delle stringhe

es.

$$G = \langle \{a, b\}, \{S, A\}, P, S \rangle$$

$S \to aSa \mid aAb \mid aAa$ TIPO 1

$aA \to aa$

$Ab \to aab$

$S \to aAb \to aab$

} linguaggi generati si chiamano L di tipo 1

ESEMPIO

G_1 $V_T = \{a, b, c\}$ $V_N = \{S, B, C, F, G\}$ G_1 TIPO 0

$S \to aSBC$

$L(G_1) = \{a^n b^n c^n \mid n \geq 1\}$

$CB \to BC$

POSSIAMO GENERARE $L(G_2) = L(G_1)$

$SB \to BF$

G_2 TIPO 1

$S \to aBSc$

$FC \to cG$

$Ba \to aB$

$GC \to cG$

$Bb \to bb$

$G \to \varepsilon$

Per Essendo i linguaggi equivalenti osserviamo che $L(G_1)$ é un L di tipo 0, mentre il linguaggio $L(G_2)$ é di tipo 1

Il termine di linguaggio contestuale, storicamente definito da chomsky come la classe dei linguaggi generati da grammatiche aventi le produzioni contestuali del tipo

$$\beta_1 A \beta_2 \to \beta_1 \gamma \beta_2 \quad A \in V_N \quad \beta_1, \beta_2 \in V^*, \gamma \in V^+$$

La produzione $A \Rightarrow \gamma$ può essere applicata solo se A si trova nel contesto di $<\beta_1, \beta_2>$

Se prendiamo la grammatica la produzione $Bb \to bb$ il carattere B può essere rimpiazzato dal carattere b solo se alla sua destra è presente il contesto b

GRAMMATICHE DI TIPO 2 AUTOMI A PILA

Le grammatiche di tipo 2 sono dette non contestuali e ammettono produzioni del tipo:

$A \to \beta \quad A \in V_N \quad \beta \in V^+$

cioè produzioni in ogni non terminale A può essere riscritto in una stringa B indipendentemente dal contesto

ESEMPIO

$G_3 = <\{a, b\}, \{S, A\}, P, S>$ $L(G_3)$ TIPO 0

P: $S \to aAb$

$aA \to aaAb$

$A \to \varepsilon$

$G_4 \Rightarrow P: S \to aSb \mid ab$

$L(G_4)$ linguaggio di tipo 2

$L(G_3) = L(G_4)$

es.

ASSIOMA "E" carattere non terminale

$E \to E + T \mid T$

$T \to T \times F \mid F$

$F \to i \mid (E)$

partendo da E generiamo tutte le espressioni di somma e prodotto in una variabile i

Le grammatiche di tipo 2 generano linguaggi di tipo 2

$G = < \{ (,) \}, \{S\}, P, S >$

$S \to (\mid)$ → linguaggio di parentesi bilanciate

$S \to SS$

$S \to (S)$

GRAMMATICHE DI TIPO 3 (ASF) ⇒ GRAMMATICHE LINEARI DESTRE O REGOLARI

ammettono produzioni

$A \to \delta \quad A \in V_N \quad \delta \in (V_T \circ V_N) \cup V_T = V \cup V_T$

tali linguaggi sono rappresentabili per mezzo di espressioni regolari e quindi deriva il nome di regolari

$G = < \{a, b\}, \{S\}, P, S >$ $\quad$ P: $S \to aS$ $\quad$ G TIPO 3

$S \to b$ $\quad$ $L = \{ a^n b \mid n \geq 0 \}$

$S \to aS \to aaS \to aaaS \to aaab = a^3 b$

Il termine lineare destra deriva dal fatto che il lato destro di ogni produzione compare al più un carattere non terminale

In modo analogo possiamo definire le lineari sinistre, linguaggi regolari caratterizzate dalle seguenti regole

$A \to \delta \quad A \in V_N \quad \delta \in (V_N \circ V_T) \cup V_T = V \cup V_T$

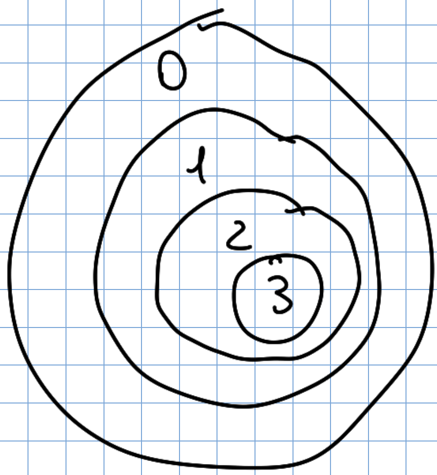
Per ogni grammatica lineare destra ne esiste una lineare sinistra equivalente e viceversa

es.

$S \to Ab \mid b$

$A \to Aa \mid a$

GERARCHIA DI CHOMSKY



Si può verificare che per ogni $0 \leq n \leq 2$ ogni grammatica di tipo $n+1$ è anche di tipo n e pertanto l'insieme dei linguaggi di tipo n contiene tutti i linguaggi che di grammatiche

DEFINIZIONE

Un L è strettamente di tipo n se $\exists$ una G di tipo n che genera L e $\nexists$ alcuna grammatica G' di tipo $m > n$ che possa generarlo

es.

$L = \{ a^n b^n \mid n \geq 0 \}$

G_1 TIPO 0

G_2 TIPO 1

G_3 TIPO 2

$L(G_1)$ TIPO 0

Si può dimostrare che non esiste nessuna grammatica di tipo 3 che lo può generare

$L = \{a^n b^n \mid n \geq 0\}$ è strettamente di tipo 2

Le grammatiche di tipo 3 sono gerarchicamente le più povere

TIPO	PRODUZIONI	DOVE PRENDIAMO LE VARIABILI	DEFINIZIONE
0	$\alpha \to \beta$	$\alpha \in V^* \circ V_N \circ V^*$ $\beta \in V^*$	ho qualsiasi regola
1	$\alpha \to \beta$	$\alpha \in V^* \circ V_N \circ V^*$ $\beta \in V^+$ $\lvert\alpha\rvert \leq \lvert\beta\rvert$	ho regole che non riducono la lunghezza della stringhe
2	$A \to \beta$	$A \in V_N$ $\beta \in V^+$	ogni non terminale può essere riscritto come terminale indipendentemente dal contesto
3	$A \to \delta$	$A \in V_N$ $\delta \in (V_T \circ V_N) \cup V_T$	al lato destro di ogni produzione compare un carattere non terminale

es.

$L = \{ (x\#)^+ \mid x = \text{Permutazioni di } \langle a, b, c \rangle \}$

1. Grammatiche di Tipo 0 (Illimitate)

- Sono le più generali, possono avere **qualsiasi forma** nelle regole di produzione.
- Le regole hanno la forma:

$$\alpha \to \beta$$

  dove **α** e **β** sono qualsiasi sequenza di terminali e non terminali, con la condizione che **α non sia vuoto.**
- **Esempio**
  - $SAB \to ASB$
  - $AB \to BA$
  - $A \to a, B \to b$
- Possono descrivere **qualsiasi linguaggio riconoscibile da una macchina di Turing**, quindi anche problemi molto complessi.

2. Grammatiche di Tipo 1 (Sensibili al contesto)

- Ogni regola ha la forma:

$$\alpha A \beta \to \alpha \gamma \beta$$

  con **A un non terminale** e **α, β, γ qualsiasi stringa (γ non vuoto).**
- Significa che **A** può trasformarsi in **γ,** ma solo se si trova nel contesto di **α e β.**
- **Esempio**
  - $ABC \to ADC$ (B si trasforma in D solo in presenza di A e C)
- Riconosciute da **macchine di Turing con memoria limitata.**

TIPO 1 ammette qualsiasi regola che non riduca la lunghezza delle stringhe

3. Grammatiche di Tipo 2 (Libere dal contesto - CFG)

- Regole della forma:

$$A \to \gamma$$

  con **A un non terminale e γ una sequenza di terminali e/o non terminali.**

- **Esempio**
  - $S \to (S)$
  - $S \to SS$
  - $S \to \varepsilon$
- Utili nei **parser dei linguaggi di programmazione** e riconosciute da **automi a pila.**

4. Grammatiche di Tipo 3 (Regolari)

- **Regole con una forma molto ristretta:**

$$A \to aB$$

  oppure

$$A \to a$$

  con **A e B non terminali, a un terminale.**

- **Esempio** (linguaggio delle stringhe con solo "a" seguite da "b"):
  - $S \to aS$
  - $S \to b$
- Sono usate nei **linguaggi regolari** (es. espressioni regolari) e riconosciute da **automi a stati finiti.**

Riassunto

Tipo	Regole	Potenza
0 (Illimitata)	Nessuna restrizione	Macchina di Turing (qualsiasi problema computabile)
1 (Sensibile al contesto)	$\alpha A \beta \to \alpha \gamma \beta$	Macchina di Turing con memoria limitata
2 (Libera dal contesto - CFG)	$A \to \gamma$	Automi a pila (linguaggi di programmazione)
3 (Regolare)	$A \to aB$ o $A \to a$	Automi a stati finiti (espressioni regolari)