ELEMENTI DI INFORMATICA TEORICA

Parte 2: Linguaggi Formali e Automi

2.1 Linguaggi e grammatiche formali

Giovanni Amendola

Corso di laurea triennale in Informatica
Università della Calabria

30 marzo 2022

Anno Accademico 2021/2022

Gerarchia di Chomsky

- Classificazione delle grammatiche di **Noam Chomsky** (nato a Filadelfia nel 1928)
- La teoria della grammatica generativa risale al 1956.

Gerarchia di Chomsky

- Classificazione delle grammatiche di **Noam Chomsky** (nato a Filadelfia nel 1928)
- La teoria della grammatica generativa risale al 1956.

Gerarchia di Chomsky

Le grammatiche di Chomsky si classificano in base al tipo di regole di produzione che sono usate:

Gerarchia di Chomsky

- Classificazione delle grammatiche di **Noam Chomsky** (nato a Filadelfia nel 1928)
- La teoria della grammatica generativa risale al 1956.

Gerarchia di Chomsky

Le grammatiche di Chomsky si classificano in base al tipo di regole di produzione che sono usate:

- **Linguaggi di tipo** 0: produzioni di qualsiasi tipo.

Gerarchia di Chomsky

- Classificazione delle grammatiche di **Noam Chomsky** (nato a Filadelfia nel 1928)
- La teoria della grammatica generativa risale al 1956.

Gerarchia di Chomsky

Le grammatiche di Chomsky si classificano in base al tipo di regole di produzione che sono usate:

- **Linguaggi di tipo** 0: produzioni di qualsiasi tipo.
- **Linguaggi di tipo** 1: produzioni contestuali (*context sensitive*, *CS*).

Gerarchia di Chomsky

- Classificazione delle grammatiche di **Noam Chomsky** (nato a Filadelfia nel 1928)
- La teoria della grammatica generativa risale al 1956.

Gerarchia di Chomsky

Le grammatiche di Chomsky si classificano in base al tipo di regole di produzione che sono usate:

- **Linguaggi di tipo** 0: produzioni di qualsiasi tipo.
- **Linguaggi di tipo** 1: produzioni contestuali (*context sensitive*, *CS*).
- **Linguaggi di tipo** 2: produzioni non contestuali (*context free*, *CF*).

Gerarchia di Chomsky

- Classificazione delle grammatiche di **Noam Chomsky** (nato a Filadelfia nel 1928)
- La teoria della grammatica generativa risale al 1956.

Gerarchia di Chomsky

Le grammatiche di Chomsky si classificano in base al tipo di regole di produzione che sono usate:

- **Linguaggi di tipo** 0: produzioni di qualsiasi tipo.
- **Linguaggi di tipo** 1: produzioni contestuali (*context sensitive, CS*).
- **Linguaggi di tipo** 2: produzioni non contestuali (*context free, CF*).
- **Linguaggi di tipo** 3: produzioni regolari.

Grammatiche e linguaggi di tipo 3

Definizione: Grammatica lineare destra

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice *lineare destra* se ciascuna produzione è uno dei seguenti tipi:

- $A \to bC$, oppure
- $A \to b$

dove $A, C \in V_N$ e $b \in V_T$.

Grammatiche e linguaggi di tipo 3

Definizione: Grammatica lineare destra

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice *lineare destra* se ciascuna produzione è uno dei seguenti tipi:

- $A \to bC$, oppure
- $A \to b$

dove $A, C \in V_N$ e $b \in V_T$.

- Analogamente, si definiscono le *grammatiche lineari sinistre*

Grammatiche e linguaggi di tipo 3

Definizione: Grammatica lineare destra

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice *lineare destra* se ciascuna produzione è uno dei seguenti tipi:

- $A \to bC$, oppure
- $A \to b$

dove $A, C \in V_N$ e $b \in V_T$.

- Analogamente, si definiscono le *grammatiche lineari sinistre*
- Per ogni grammatica lineare destra ne esiste una sinistra equivalente.

Grammatiche e linguaggi di tipo 3

Definizione: Grammatica lineare destra

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice *lineare destra* se ciascuna produzione è uno dei seguenti tipi:

- $A \to bC$, oppure
- $A \to b$

dove $A, C \in V_N$ e $b \in V_T$.

- Analogamente, si definiscono le *grammatiche lineari sinistre*
- Per ogni grammatica lineare destra ne esiste una sinistra equivalente.
- Il linguaggio generato da una grammatica lineare destra [risp. sinistra] si chiama **linguaggio lineare destro** [risp. sinistro].
- Queste grammatiche e i linguaggi corrispondenti sono anche detti **regolari o di tipo 3**.

Grammatiche e linguaggi di tipo 3

Esempio

Il linguaggio $\{a^n b \mid n \geq 0\}$ è di tipo 3, in quanto può essere generato dalla grammatica avente le seguenti regole di produzione:

- $S \rightarrow aS$
- $S \rightarrow b$

Grammatiche e linguaggi di tipo 3

Esempio

Il linguaggio $\{a^n b \mid n \geq 0\}$ è di tipo 3, in quanto può essere generato dalla grammatica avente le seguenti regole di produzione:

- $S \rightarrow aS$
- $S \rightarrow b$

Esempio

Il linguaggio di tutte le stringhe binarie su $\{0, 1\}$ è di tipo 3, in quanto può essere generato dalla grammatica avente le seguenti produzioni:

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

Grammatiche e linguaggi di tipo 2

Definizione: **Grammatica non contestuale**

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **non contestuale** se tutte le produzioni sono del tipo

$$A \rightarrow \beta$$

dove $A \in V_N$ e $\beta \in (V_T \cup V_N)^+$.

Grammatiche e linguaggi di tipo 2

Definizione: Grammatica non contestuale

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **non contestuale** se tutte le produzioni sono del tipo

$$A \rightarrow \beta$$

dove $A \in V_N$ e $\beta \in (V_T \cup V_N)^+$.

I **linguaggi di tipo 2**, detti anche **non contestuali** o *context-free*, sono i linguaggi generati dalle grammatiche non contesutali.

Grammatiche e linguaggi di tipo 2

Definizione: Grammatica non contestuale

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **non contestuale** se tutte le produzioni sono del tipo

$$A \rightarrow \beta$$

dove $A \in V_N$ e $\beta \in (V_T \cup V_N)^+$.

I **linguaggi di tipo 2**, detti anche **non contestuali** o *context-free*, sono i linguaggi generati dalle grammatiche non contesutali.

Osservazione: Linguaggi di tipo 3 $\subseteq$ Linguaggi di tipo 2

Grammatiche e linguaggi di tipo 2

Definizione: **Grammatica non contestuale**

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **non contestuale** se tutte le produzioni sono del tipo

$$A \to \beta$$

dove $A \in V_N$ e $\beta \in (V_T \cup V_N)^+$.

I **linguaggi di tipo 2**, detti anche **non contestuali** o *context-free*, sono i linguaggi generati dalle grammatiche non contesutali.

Osservazione: Linguaggi di tipo 3 $\subseteq$ Linguaggi di tipo 2

Esempio

Il linguaggio $\{a^n b^n \mid n \geq 1\}$ è di tipo 2, in quanto può essere generato dalla grammatica avente le seguenti regole di produzione:

- $S \to aSb$
- $S \to ab$

Grammatiche e linguaggi di tipo 2

Definizione: Grammatica non contestuale

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **non contestuale** se tutte le produzioni sono del tipo

$$A \to \beta$$

dove $A \in V_N$ e $\beta \in (V_T \cup V_N)^+$.

I **linguaggi di tipo 2**, detti anche **non contestuali** o *context-free*, sono i linguaggi generati dalle grammatiche non contesutali.
Osservazione: Linguaggi di tipo 3 $\subseteq$ Linguaggi di tipo 2

Esempio

Il linguaggio $\{a^n b^n \mid n \geq 1\}$ è di tipo 2, in quanto può essere generato dalla grammatica avente le seguenti regole di produzione:

- $S \to aSb$
- $S \to ab$

Osservazione: Linguaggi di tipo 3 $\subsetneq$ Linguaggi di tipo 2

Grammatiche e linguaggi di tipo 2

Definizione: Grammatica lineare

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ di tipo 2 si dice **lineare** se tutte le produzioni sono del tipo

$$A \to \beta$$

dove $A \in V_N$ e $\beta \in (V_T \cup V_N)^+$ contiene al massimo un solo simbolo non terminale.

Grammatiche e linguaggi di tipo 2

Definizione: Grammatica lineare

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ di tipo 2 si dice **lineare** se tutte le produzioni sono del tipo

$$A \rightarrow \beta$$

dove $A \in V_N$ e $\beta \in (V_T \cup V_N)^+$ contiene al massimo un solo simbolo non terminale.

Le grammatiche lineari generano i **linguaggi lineari**.

Grammatiche e linguaggi di tipo 2

Definizione: **Grammatica lineare**

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ di tipo 2 si dice **lineare** se tutte le produzioni sono del tipo

$$A \to \beta$$

dove $A \in V_N$ e $\beta \in (V_T \cup V_N)^+$ contiene al massimo un solo simbolo non terminale.

Le grammatiche lineari generano i **linguaggi lineari**.

Esempio

Il linguaggio $\{a^n cb^n \mid n \geq 0\}$ è lineare, in quanto può essere generato dalla grammatica lineare avente le seguenti regole di produzione:

- $S \to aSb$
- $S \to c$

Grammatiche e linguaggi di tipo 2

Definizione: Grammatica lineare

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P\rangle$ di tipo 2 si dice **lineare** se tutte le produzioni sono del tipo

$$A \to \beta$$

dove $A \in V_N$ e $\beta \in (V_T \cup V_N)^+$ contiene al massimo un solo simbolo non terminale.

Le grammatiche lineari generano i **linguaggi lineari**.

Esempio

Il linguaggio $\{a^n cb^n \mid n \geq 0\}$ è lineare, in quanto può essere generato dalla grammatica lineare avente le seguenti regole di produzione:

- $S \to aSb$
- $S \to c$

Osservazione: Linguaggi regolari $\subsetneq$ Linguaggi lineari $\subsetneq$ Linguaggi non contestuali

Grammatiche e linguaggi di tipo 1

Definizione: Grammatica contestuale

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **contestuale** se tutte le produzioni sono del tipo

$$\alpha \to \beta$$

dove $\alpha \in V^* \cdot V_N \cdot V^*$ e $\beta \in V^*$ con $|\alpha| \leq |\beta|$ (nota che $V = V_T \cup V_N$).

Grammatiche e linguaggi di tipo 1

Definizione: Grammatica contestuale

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **contestuale** se tutte le produzioni sono del tipo

$$\alpha \to \beta$$

dove $\alpha \in V^* \cdot V_N \cdot V^*$ e $\beta \in V^*$ con $|\alpha| \leq |\beta|$ (nota che $V = V_T \cup V_N$).

- I **linguaggi di tipo 1**, detti anche **contestuali** o *context sensitive*, sono i linguaggi generati dalle grammatiche contestuali.

Grammatiche e linguaggi di tipo 1

Definizione: Grammatica contestuale

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **contestuale** se tutte le produzioni sono del tipo

$$\alpha \to \beta$$

dove $\alpha \in V^* \cdot V_N \cdot V^*$ e $\beta \in V^*$ con $|\alpha| \leq |\beta|$ (nota che $V = V_T \cup V_N$).

- I **linguaggi di tipo 1**, detti anche **contestuali** o *context sensitive*, sono i linguaggi generati dalle grammatiche contestuali.
- **Osservazione**: Le produzioni delle grammatiche contestuali non riducono la lunghezza delle forme di frase.

Grammatiche e linguaggi di tipo 1

Definizione: Grammatica contestuale

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **contestuale** se tutte le produzioni sono del tipo

$$\alpha \to \beta$$

dove $\alpha \in V^* \cdot V_N \cdot V^*$ e $\beta \in V^*$ con $|\alpha| \leq |\beta|$ (nota che $V = V_T \cup V_N$).

- I **linguaggi di tipo 1**, detti anche **contestuali** o *context sensitive*, sono i linguaggi generati dalle grammatiche contestuali.
- **Osservazione**: Le produzioni delle grammatiche contestuali non riducono la lunghezza delle forme di frase.
- **Osservazione**: Linguaggi di tipo 2 $\subseteq$ Linguaggi di tipo 1

Grammatiche e linguaggi di tipo 1

Esempio

Il linguaggio $\{a^n b^n c^n \mid n \geq 1\}$ è di tipo 1, in quanto può essere generato dalla grammatica avente le seguenti regole di produzione:

- $S \rightarrow aSBC \mid aBC$
- $CB \rightarrow BC$
- $aB \rightarrow ab$
- $bB \rightarrow bb$
- $bC \rightarrow bc$
- $cC \rightarrow cc$

Infatti in tali produzioni la lunghezza della stringa sinistra è $\leq$ alla lunghezza della stringa destra.

Grammatiche e linguaggi di tipo 1

Esempio

Il linguaggio $\{a^n b^n c^n \mid n \geq 1\}$ è di tipo 1, in quanto può essere generato dalla grammatica avente le seguenti regole di produzione:

- $S \to aSBC \mid aBC$
- $CB \to BC$
- $aB \to ab$
- $bB \to bb$
- $bC \to bc$
- $cC \to cc$

Infatti in tali produzioni la lunghezza della stringa sinistra è $\leq$ alla lunghezza della stringa destra.

Osservazione: Linguaggi di tipo 2 $\subsetneq$ Linguaggi di tipo 1

Grammatiche e linguaggi di tipo 0

Definizione: Grammatica di tipo 0

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **di tipo** 0 se tutte le produzioni sono del tipo

$$\alpha \to \beta$$

dove $\alpha \in V^* \cdot V_N \cdot V^*$ e $\beta \in V^*$.

Grammatiche e linguaggi di tipo 0

Definizione: Grammatica di tipo 0

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **di tipo** 0 se tutte le produzioni sono del tipo

$$\alpha \to \beta$$

dove $\alpha \in V^* \cdot V_N \cdot V^*$ e $\beta \in V^*$.

- I **linguaggi di tipo 0** sono i linguaggi generati dalle grammatiche di tipo 0, ovvero tutti i possibili linguaggi generabili con le grammatiche formali.

Grammatiche e linguaggi di tipo 0

Definizione: Grammatica di tipo 0

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **di tipo** 0 se tutte le produzioni sono del tipo

$$\alpha \to \beta$$

dove $\alpha \in V^* \cdot V_N \cdot V^*$ e $\beta \in V^*$.

- I **linguaggi di tipo 0** sono i linguaggi generati dalle grammatiche di tipo 0, ovvero tutti i possibili linguaggi generabili con le grammatiche formali.
- **Osservazione**: Le produzioni delle grammatiche di tipo 0 possono avere il lato destro più corto del sinistro, rendendo possibile l'accorciamento di una stringa.

Grammatiche e linguaggi di tipo 0

Definizione: Grammatica di tipo 0

Una grammatica $\mathcal{G} = \langle V_T, V_N, S, P \rangle$ si dice **di tipo** 0 se tutte le produzioni sono del tipo

$$\alpha \to \beta$$

dove $\alpha \in V^* \cdot V_N \cdot V^*$ e $\beta \in V^*$.

- I **linguaggi di tipo 0** sono i linguaggi generati dalle grammatiche di tipo 0, ovvero tutti i possibili linguaggi generabili con le grammatiche formali.
- **Osservazione**: Le produzioni delle grammatiche di tipo 0 possono avere il lato destro più corto del sinistro, rendendo possibile l'accorciamento di una stringa.
- **Osservazione**: Linguaggi di tipo 1 $\subsetneq$ Linguaggi di tipo 0

ϵ-produzioni

Definizione: ϵ-produzioni

Le ϵ-produzioni sono produzioni che generano la stringa vuota ϵ.

ϵ-produzioni

Definizione: ϵ-produzioni

Le ϵ-produzioni sono produzioni che generano la stringa vuota ϵ.

- Se la grammatica è di tipo 0, le ϵ-produzioni possono essere generate (accorciano sempre la stringa di partenza).

ϵ-produzioni

Definizione: ϵ-produzioni

Le ϵ-produzioni sono produzioni che generano la stringa vuota ϵ.

- Se la grammatica è di tipo 0, le ϵ-produzioni possono essere generate (accorciano sempre la stringa di partenza).
- Se la grammatica è di tipo 1, 2 o 3, le ϵ-produzioni non sarebbero permesse.

ϵ-produzioni

Definizione: ϵ-produzioni

Le ϵ-produzioni sono produzioni che generano la stringa vuota ϵ.

- Se la grammatica è di tipo 0, le ϵ-produzioni possono essere generate (accorciano sempre la stringa di partenza).
- Se la grammatica è di tipo 1, 2 o 3, le ϵ-produzioni non sarebbero permesse.
- Senza ϵ-produzioni non si può generare la stringa vuota.

ϵ-produzioni

Definizione: ϵ-produzioni

Le ϵ-produzioni sono produzioni che generano la stringa vuota ϵ.

- Se la grammatica è di tipo 0, le ϵ-produzioni possono essere generate (accorciano sempre la stringa di partenza).
- Se la grammatica è di tipo 1, 2 o 3, le ϵ-produzioni non sarebbero permesse.
- Senza ϵ-produzioni non si può generare la stringa vuota.

Esempio

$\mathcal{L} = \{a^n b^n \mid n \geq 1\}$ è di tipo 2, mentre $\mathcal{L}' = \{a^n b^n \mid n \geq 0\}$ non lo sarebbe. E tuttavia $\mathcal{L}' = \mathcal{L} \cup \{\epsilon\}$.

ϵ-produzioni

Definizione: ϵ-produzioni

Le ϵ-produzioni sono produzioni che generano la stringa vuota ϵ.

- Se la grammatica è di tipo 0, le ϵ-produzioni possono essere generate (accorciano sempre la stringa di partenza).
- Se la grammatica è di tipo 1, 2 o 3, le ϵ-produzioni non sarebbero permesse.
- Senza ϵ-produzioni non si può generare la stringa vuota.

Esempio

$\mathcal{L} = \{a^n b^n \mid n \geq 1\}$ è di tipo 2, mentre $\mathcal{L}' = \{a^n b^n \mid n \geq 0\}$ non lo sarebbe. E tuttavia $\mathcal{L}' = \mathcal{L} \cup \{\epsilon\}$.

- Estendiamo le precedenti definizioni ammettendo anche ϵ-produzioni.

ϵ-produzioni

Teorema

Sia $\mathcal{G}$ una grammatica che genera $\mathcal{L}(\mathcal{G})$ e $\epsilon \notin \mathcal{L}(\mathcal{G})$. Se $\mathcal{L}(\mathcal{G})$ è di tipo 1 o 2, allora è possibile modificare $\mathcal{G}$ in modo che generi $\mathcal{L}(\mathcal{G}) \cup \{\epsilon\}$ usando ϵ-produzioni e senza modificare le altre produzioni.

ϵ-produzioni

Teorema

Sia $\mathcal{G}$ una grammatica che genera $\mathcal{L}(\mathcal{G})$ e $\epsilon \notin \mathcal{L}(\mathcal{G})$. Se $\mathcal{L}(\mathcal{G})$ è di tipo 1 o 2, allora è possibile modificare $\mathcal{G}$ in modo che generi $\mathcal{L}(\mathcal{G}) \cup \{\epsilon\}$ usando ϵ-produzioni e senza modificare le altre produzioni.

Intuizione della dimostrazione

Sia $\mathcal{G} = \langle V_T, V_N, S, P \rangle$. Allora per generare $\mathcal{L}(\mathcal{G}) \cup \{\epsilon\}$ basta usare la grammatica $\mathcal{G}' = \langle V_T, V_N \cup \{S'\}, S', P' \rangle$, dove S' è un nuovo simbolo non terminale ed il nuovo assioma di $\mathcal{G}'$, mentre

$$P' = P \cup \{S' \to S,\ S' \to \epsilon\}.$$

ϵ-produzioni

Teorema

Sia $\mathcal{G}$ una grammatica che genera $\mathcal{L}(\mathcal{G})$ e $\epsilon \notin \mathcal{L}(\mathcal{G})$. Se $\mathcal{L}(\mathcal{G})$ è di tipo 1 o 2, allora è possibile modificare $\mathcal{G}$ in modo che generi $\mathcal{L}(\mathcal{G}) \cup \{\epsilon\}$ usando ϵ-produzioni e senza modificare le altre produzioni.

Intuizione della dimostrazione

Sia $\mathcal{G} = \langle V_T, V_N, S, P \rangle$. Allora per generare $\mathcal{L}(\mathcal{G}) \cup \{\epsilon\}$ basta usare la grammatica $\mathcal{G}' = \langle V_T, V_N \cup \{S'\}, S', P' \rangle$, dove S' è un nuovo simbolo non terminale ed il nuovo assioma di $\mathcal{G}'$, mentre

$$P' = P \cup \{S' \to S,\ S' \to \epsilon\}.$$

Osservazione

Se le ϵ-produzioni sono in **posizione generale**

- le grammatiche di tipo 1 diventano equivalenti a quelle di tipo 0;

ϵ-produzioni

Teorema

Sia $\mathcal{G}$ una grammatica che genera $\mathcal{L}(\mathcal{G})$ e $\epsilon \notin \mathcal{L}(\mathcal{G})$. Se $\mathcal{L}(\mathcal{G})$ è di tipo 1 o 2, allora è possibile modificare $\mathcal{G}$ in modo che generi $\mathcal{L}(\mathcal{G}) \cup \{\epsilon\}$ usando ϵ-produzioni e senza modificare le altre produzioni.

Intuizione della dimostrazione

Sia $\mathcal{G} = \langle V_T, V_N, S, P\rangle$. Allora per generare $\mathcal{L}(\mathcal{G}) \cup \{\epsilon\}$ basta usare la grammatica $\mathcal{G}' = \langle V_T, V_N \cup \{S'\}, S', P'\rangle$, dove S' è un nuovo simbolo non terminale ed il nuovo assioma di $\mathcal{G}'$, mentre

$$P' = P \cup \{S' \to S,\ S' \to \epsilon\}.$$

Osservazione

Se le ϵ-produzioni sono in **posizione generale**

- le grammatiche di tipo 1 diventano equivalenti a quelle di tipo 0;
- le grammatiche di tipo 2 e 3 restano invariate (rispetto a quelle con la ϵ-produzione applicata solo all'assioma).

ϵ-produzioni

Teorema

Una grammatica non contestuale (tipo 2) estesa con ϵ-produzioni:

- se non genera ϵ si può costruire una grammatica equivalente senza ϵ-produzioni;
- se genera ϵ si può costruire una grammatica equivalente solo con ϵ-produzione sull'assioma.

ϵ-produzioni

Teorema

Una grammatica non contestuale (tipo 2) estesa con ϵ-produzioni:

- se non genera ϵ si può costruire una grammatica equivalente senza ϵ-produzioni;
- se genera ϵ si può costruire una grammatica equivalente solo con ϵ-produzione sull'assioma.

Dimostrazione

Supponiamo per semplicità che esista una sola ϵ-produzione $E \to \epsilon$ e l'assioma non appaia mai a destra di una produzione:

- Se E è l'assioma, l'asserto è dimostrato.
- Se E appare a destra delle $k \geq 1$ produzioni $A_i \to \beta_i$, al posto di $E \to \epsilon$ aggiungiamo k produzioni $A_i \to \beta_i[\epsilon/E]$.
- Il procedimento si ripete per le (eventuali) nuove ϵ-produzioni.

ϵ-produzioni

Osservazione

- La precedente dimostrazione mostra anche che una **grammatica di tipo 3** rimane invariata.
- Inoltre, in tal caso, **la dimostrazione si semplifica**: l'eliminazione di una ϵ-produzione non può generare nuove ϵ-produzioni (a destra c'è sempre un solo simbolo non terminale).

ϵ-produzioni

Osservazione

- La precedente dimostrazione mostra anche che una **grammatica di tipo 3** rimane invariata.
- Inoltre, in tal caso, **la dimostrazione si semplifica**: l'eliminazione di una ϵ-produzione non può generare nuove ϵ-produzioni (a destra c'è sempre un solo simbolo non terminale).

Esempio (Grammatiche di tipo 3 ed ϵ-produzioni)

La seguente grammatica di tipo 3 (regolare)

- $S \to bX \mid aB$
- $B \to cX$
- $X \to \epsilon$

ϵ-produzioni

Osservazione

- La precedente dimostrazione mostra anche che una **grammatica di tipo 3** rimane invariata.
- Inoltre, in tal caso, **la dimostrazione si semplifica**: l'eliminazione di una ϵ-produzione non può generare nuove ϵ-produzioni (a destra c'è sempre un solo simbolo non terminale).

Esempio (Grammatiche di tipo 3 ed ϵ-produzioni)

La seguente grammatica di tipo 3 (regolare)

- $S \to bX \mid aB$
- $B \to cX$
- $X \to \epsilon$

si può modificare in quest'altra:

- $S \to b \mid aB$
- $B \to c$

ϵ-produzioni

Esempio (Grammatiche di tipo 2 ed ϵ-produzioni)

La seguente grammatica è di tipo 2 (non contestuale)

- $S \rightarrow AB \mid aB \mid B$
- $A \rightarrow ab \mid aB$
- $B \rightarrow cX \mid X$
- $X \rightarrow \epsilon$

ϵ-produzioni

Esempio (Grammatiche di tipo 2 ed ϵ-produzioni)

La seguente grammatica è di tipo 2 (non contestuale)

- $S \rightarrow AB \mid aB \mid B$
- $A \rightarrow ab \mid aB$
- $B \rightarrow cX \mid X$
- $X \rightarrow \epsilon$

Primo passaggio: eliminazione della ϵ-produzione $X \rightarrow \epsilon$

- $S \rightarrow AB \mid aB \mid B$
- $A \rightarrow ab \mid aB$
- $B \rightarrow c \mid \epsilon$

ϵ-produzioni

Esempio (Grammatiche di tipo 2 ed ϵ-produzioni)

La seguente grammatica è di tipo 2 (non contestuale)

- $S \rightarrow AB \mid aB \mid B$
- $A \rightarrow ab \mid aB$
- $B \rightarrow cX \mid X$
- $X \rightarrow \epsilon$

Primo passaggio: eliminazione della ϵ-produzione $X \rightarrow \epsilon$

- $S \rightarrow AB \mid aB \mid B$
- $A \rightarrow ab \mid aB$
- $B \rightarrow c \mid \epsilon$

Secondo passaggio: eliminazione della ϵ-produzione $B \rightarrow \epsilon$

- $S \rightarrow AB \mid A \mid aB \mid a \mid B \mid \epsilon$
- $A \rightarrow ab \mid aB \mid a$
- $B \rightarrow c$

ϵ-produzioni

Perché le grammatiche di tipo 1 con ϵ-produzioni sono equivalenti alle grammatiche di tipo 0?

ϵ-produzioni

Perché le grammatiche di tipo 1 con ϵ-produzioni sono equivalenti alle grammatiche di tipo 0?

Esempio (Grammatiche di tipo 1 ed ϵ-produzioni)

Consideriamo una grammatica di tipo 0 avente la seguente produzione

- $AB \rightarrow C$

Possiamo costruire una grammatica di tipo 1 (contestuale) con una ϵ-produzione equivalente alla precedente, aggiungendo un nuovo simbolo non terminale e sostituendo la precedente produzione con le seguenti

- $AB \rightarrow CH$
- $H \rightarrow \epsilon$

Gerarchia di Chomsky

Le grammatiche formali di Chomsky si classificano in base al tipo di produzioni utilizzate:

- **Linguaggi di tipo 0**: produzioni qualsiasi ($\alpha \to \beta$, $\alpha \in V^* \cdot V_N \cdot V^*$, $\beta \in V^*$)
- **Linguaggi di tipo 1** o **Linguaggi contestuali**: produzioni contestuali ($\alpha \to \beta$, $\alpha \in V^* \cdot V_N \cdot V^*$, $\beta \in V^*$, $|\alpha| \leq |\beta|$) ed ϵ-produzioni solo sull'assioma.
- **Linguaggi di tipo 2** o **Linguaggi non contestuali**: produzioni non contestuali ($A \to \beta$, $A \in V_N$ e $\beta \in V^+$) ed ϵ-produzioni [Nota: basta richiedere $\beta \in V^*$]
- **Linguaggi di tipo 3** o **Linguaggi regolari**: produzioni lineari destre ($A \to bC \mid b$) o sinistre ($A \to Cb \mid b$) ed ϵ-produzioni.

Gerarchia di Chomsky