Teorema 1 (Chiusure linguaggi regolari)**.** *I linguaggi **regolari** sono chiusi per i seguenti operatori:*

- ***unione***
- ***concatenazione***
- ***ripetizione** (stella di Kleene)*
- ***complementazione***
- ***intersezione***

Teorema 2 (Chiusure linguaggi liberi)**.** *I linguaggi **liberi** sono chiusi per i seguenti operatori:*

- ***unione***
- ***concatenazione***
- ***ripetizione** (stella di Kleene)*

Nota. Si osservi che i linguaggi **liberi deterministici** sono chiusi solo per complementazione.

Lemma 1 (Osservazioni su chiusure interessanti)**.** *Sono provati i seguenti fatti:*

1. *Se L_1 è libero e L_2 è regolare allora $L_1 \cap L_2$ rimane un **linguaggio libero**.*
2. *Se $L_1 \cap L_2$ è non regolare ma L_2 è regolare allora L_1 è non regolare. (per la chiusura dei linguaggi regolari).*
3. *Se $L_1 \cap L_2$ è non libero ma L_2 è regolare allora L_1 è **non libero**. (per 1).*

Teorema 3 (nozioni su linguaggi LL(k))**.** *Ecco una serie di teoremi che descrivono proprietà utili sulle grammatiche $LL(k)$:*

1. *Se L regolare allora $\exists G$ di classe $LL(1)$.*
2. *Una grammatica ricorsiva sinistra non è $LL(k)$ per nessun k.*
3. *Una grammatica ambigua non è $LL(k)$ per nessun k.*
4. *Se G è $LL(k)$ allora essa non è ambigua.*
5. *Se G è $LL(k)$ allora $L(G)$ è libero deterministico.*
6. *$\exists L$ libero deterministico tale che $\nexists G$ di classe $LL(k)$ per cui $L = L(G)$.*

Teorema 4 (nozioni su linguaggi LR(k))**.** *Ecco una serie di teoremi che descrivono proprietà utili sulle grammatiche $LR(k)$:*

1. *Se L è libero deterministico e gode della prefix property allora è sicuramente $LR(0)$.*

2. *Se L è finito e non gode della* prefix property *allora non è $LR(0)$.*

3. *Se G è $LR(k)$ allora G è non ambigua.*

4. *Se G è $LR(k)$ allora $L(G)$ è libero deterministico.*

5. *Un linguaggio è libero deterministico sse è generato da una grammatica $LR(k)$.*

Lemma 2 (Pumping)**.** *Se L è un linguaggio regolare, allora*

$$\exists N > 0, \forall z \in L. |z| \geq N, \exists u, v, w \text{ tali che}$$

1. $z = uvw$

2. $|uv| \leq N$

3. $|v| \geq 1$

4. $\forall k \geq 0. \qquad uv^k w \in L$

Inoltre N è minore o uguale al numero degli stati del DFA minimo *che accetta il linguaggio L.*

Nota. per provare l'appartenenza di un linguaggio L alla classe dei linguaggi regolari si usa il pumping lemma al contrario enunciato al punto 3.

Nota. Se L è finito allora non esiste nessuna $z \in L$ con $|z| \geq N$ e quindi l'implicazione materiale è vera poichè la premessa è falsa.

Nota. Se $\exists z \in L. |z| \geq N$ allora L riconosce un linguaggio *infinito*.

Lemma 3 (Backwards Pumping)**.**

Nota. Dal lemma 7 sappiamo L regolare $\Rightarrow P$. Allora per trovare i casi in cui L **non** è regolare basta usare l'implicazione negata:

$$\neg P \Rightarrow L \text{ non regolare}$$

Se vale la seguente implicazione, allora L non è regolare:
$\forall N > 0, \exists z \in L. |z| \geq N. \forall u, v, w$ *se*

- $z = uvw$
- $|uv| \leq N$
- $|v| \geq 1$

allora $\exists k \geq 0. uv^k w \notin L$

Teorema 5 (Pumping)**.** *Se L è un linguaggio libero allora*

$$\exists N > 0. \forall z \in L. \, |z| \geq N \, \forall u, v, w, x, y \text{ tali che}$$

1. $z = uvwxy$

2. $|vwx| \leq N$

3. $|vx| \geq 1$

4. $\forall k \geq 0. \qquad uv^k wx^k y \in L$

Nota. per provare l'appartenenza di un linguaggio L alla classe dei linguaggi liberi si usa il pumping theorem al contrario enunciato al punto 6.

Teorema 6 (Backwards Pumping)**.**

Nota. Dal lemma 5 sappiamo L libero $\Rightarrow P$. Allora per trovare i casi in cui L **non** è libero basta usare l'implicazione negata:

$$\neg P \Rightarrow L \text{ non libero}$$

Se vale la seguente implicazione, allora L non è libero:
$\forall N > 0, \exists z \in L.|z| \geq N. \forall u, v, w, x, y$ *se*

- $z = uvwxy$
- $|vwx| \leq N$
- $|vx| \geq 1$

allora $\exists k \geq 0.uv^k wx^k y \notin L$

Teorema 7 (Classificazione grammatiche $LL(1)$)**.** *G è una grammatica $LL(1)$ se e solo se per ogni coppia di produzioni distinte* con la stessa testa *del tipo:*

$$A \longrightarrow \alpha \mid \beta$$

si ha che:

1. $\textsc{First}(\alpha) \cap \textsc{First}(\beta) = \emptyset$

2. *(a)* *se* $\varepsilon \in \textsc{First}(\alpha)$ *allora* $\textsc{First}(\beta) \cap \textsc{Follow}(A) = \emptyset$

   (b) *se* $\varepsilon \in \textsc{First}(\beta)$ *allora* $\textsc{First}(\alpha) \cap \textsc{Follow}(A) = \emptyset$

Nota. Questo teorema viene sfruttato per mostrare che un linguaggio generato da una grammatica (spesso non) appartiene alla classe $LL(1)$.

Definizione 1 (Simboli annullabili)**.** *Presa una grammatica G si definisce $N(G) \subseteq NT$ come l'insieme di non terminali annullabili, ovvero tali $\forall A \in N(G).\, A \Rightarrow^+ \varepsilon$.*

Si può calcolare induttivamente come segue:

$$\begin{aligned} N_0(G) &= \{A \in NT \mid A \longrightarrow \varepsilon\} \\ N_i(G) &= \{A \in NT \mid A \longrightarrow c_1 \ldots c_n \\ &\quad \textit{tali che} \qquad c_1, \ldots, c_n \in N_{i-1}(G)\} \end{aligned}$$

Si termina quando si raggiunge un $N_i = N_{i-1}$.

Tecnica 1 (Rimozione produzioni ε)**.** *Avendo a disposizione una grammatica G e l'insieme dei simboli annullabili $N(G)$ calcolato come al punto 1 possiamo riscrivere tutte le produzioni del tipo $A \longrightarrow \alpha$ con $\alpha \neq \varepsilon$ in cui occorrono i simboli annullabili $z_1, \ldots, z_k$ ripiazzandola con $A \longrightarrow \alpha'$ dove α' si ottiene rimuovendo da α ogni possibile sottoinsieme di non terminali $\in N(G)$ (anche $\emptyset$) escluendo il caso in cui $\alpha' = \varepsilon$.*

Definizione 2 (Coppie unitarie)**.** *Presa una grammatica G si definisce $U(G) \subseteq NT \times NT$ come l'insieme di tutte le coppie di non terminali con produzioni unitarie del tipo $A \longrightarrow B$.*

Si può calcolare induttivamente come segue:

$$
\begin{aligned}
U_0(G) &= \{(A, A) \mid A \in NT\} \\
U_i(G) &= U_{i-1} \cup \{(A, C) \mid B \longrightarrow C \\
e & \qquad\qquad\qquad (A, B) \in U_{i-1}(G)\}
\end{aligned}
$$

Si termina quando si raggiunge un $U_i = U_{i-1}$.

Tecnica 2 (Rimozione produzioni unitarie)**.** *Avendo una grammatica G e l'insieme delle coppie delle produzioni unitarie $U(G)$, per ogni $(A, B) \in U(G)$ la grammatica G' risultante conterrà tutte le produzioni $A \longrightarrow \alpha$ per ogni $B \longrightarrow \alpha$ **non unitaria**.*

Tecnica 3 (Rimozione ricorsione sinistra)**.** *Per rimuovere la ricorsione sinistra da una produzione del tipo:*

$$A \longrightarrow \mathbf{A}\alpha_{\mathbf{1}} \mid \ldots \mid \mathbf{A}\alpha_{\mathbf{n}} \mid \beta_1 \mid \ldots \mid \beta_m$$

Si riscrive la produzione aggiungendo una derivazione ausiliaria A' nel seguente modo:

$$
\begin{aligned}
A &\longrightarrow \beta_1 \mathbf{A'} \mid \ldots \mid \beta_m \mathbf{A'} \\
\mathbf{A'} &\longrightarrow \alpha_{\mathbf{1}} \mathbf{A'} \mid \ldots \mid \alpha_{\mathbf{n}} \mathbf{A'} \mid \varepsilon
\end{aligned}
$$