Minimum Pumping Length

Gabriel Rovesti

Testo (tradotto e riassunto dall'esercizio del libro 1.55)

Il *Pumping Lemma* assicura che ogni linguaggio regolare A ammette una "pumping length" p, tale che ogni stringa di A con lunghezza almeno p si possa "pompare" (spezzare $s = xyz$ con $|y| > 0$, $|xy| \leq p$, e $xy^iz \in A$ per ogni $i \geq 0$). Se p è una pumping length per A, allora lo è anche ogni $p' \geq p$. La **minimum pumping length** è la più piccola fra tali p. Si chiede, per i dieci linguaggi seguenti, di trovare la *minima* pumping length e giustificare brevemente la risposta:

a) $000\,1^*$

b) $0^*\,1^*$

c) $001 \;\cup\; 0^*1^*$

d) $0^*1^*0^*1^* \;\cup\; 1\,0^*1$

e) $(01)^*$

f) $\{\varepsilon\}$ (solo la stringa vuota)

g) $1^*0\,1^*0\,1^*$

h) $10\,(11^*0)^*\,0$

i) La stringa fissa " 1011 " (linguaggio con un solo elemento)

j) Σ^* (tutte le stringhe sull'alfabeto)

Soluzioni e minime pumping length

a) $000\,1^*$

Le stringhe ammesse sono " 000 " seguita da zero o più '1'. - La stringa `000` (lunga 3) non è pompabile (se la riduciamo di un `0`, usciremmo dal linguaggio). - Quindi, se volessimo $p \leq 3$, saremmo costretti a pompare anche `000`. Contraddizione. - Invece, con $p = 4$, tutte le stringhe di lunghezza ≥ 4 sono `000` 1^k con $k \geq 1$, e si possono pompare prendendo il loop solo tra le `1`.

$$\boxed{p = 4 \text{ (minimo)}}$$

b) $0^* 1^*$

Qualunque stringa di almeno 1 simbolo (0 o 1) può essere pompata facilmente scegliendo come `y` un solo simbolo (0 o 1). Rimuoverlo o duplicarlo lascia comunque una forma "blocchi di 0 seguiti da blocchi di 1". Anche `0` e `1` (lunghezza 1) sono pompabili.

$$\boxed{p = 1 \text{ (minimo)}}$$

c) $001 \cup 0^*1^*$

Il linguaggio è l'unione della stringa "001" e di 0*1*. - Per 0*1*, come in (b), $p = 1$ funziona. - La stringa "001" (lunga 3) si può pompare: scegli ad es. $x = \texttt{0}, y = \texttt{0}, z = \texttt{1}$. Rimuovendo `0` si ottiene "01", che è in 0*1*; duplicandola si ottiene "0001", pure in 0*1*.

$$\boxed{p = 1 \text{ (minimo)}}$$

d) $0^*1^*0^*1^* \cup 1\,0^*1$

Il linguaggio è piuttosto grande: la prima parte ammette (0–1–0–1) in qualsiasi ordine di blocchi; la seconda parte è "1" seguito da qualche zero, poi "1". - Anche qui, sostanzialmente ogni stringa lunga almeno 1 si può pompare con un `y` monocarattere, conservando la forma. - Si verifica che `1` o `10` non danno problemi con $p = 1$.

$$\boxed{p = 1 \text{ (minimo)}}$$

e) $(01)^*$

Le stringhe sono concatenazioni di '01': ε, "01", "0101", "010101", ... - La stringa "01" di lunghezza 2 non è pompabile (se rimuovi un simbolo, non resta un multiplo di 2, e la forma si rompe). Quindi $p \neq 1$ e $p \neq 2$. - Con $p = 3$, la stringa di lunghezza 2 ('01') è esclusa (non deve essere pompata). La prima stringa ≥ 3 nel linguaggio è "0101" (lunghezza 4), che *è* pompabile scegliendo come 'y' il primo "01".

$$\boxed{p = 3 \text{ (minimo)}}$$

f) $\{\varepsilon\}$

Il linguaggio contiene solo la stringa vuota, di lunghezza 0. - Non esistono stringhe con lunghezza ≥ 1, dunque la condizione del lemma ("tutte le stringhe di lunghezza $\geq p$ si pompano") è triviale.

$$\boxed{p = 1 \text{ (minimo, o qualunque } p > 0)}$$

g) $1^*0\,1^*0\,1^*$

Serve esattamente **due** `0`, in quell'ordine, e un po' di `1` in giro. Alcune stringhe brevi, come "00" (lunghezza 2) e "001" (3) sono nel linguaggio e *non* pompabili se $p \leq 3$. Infatti, es. "00" non puoi rimuovere un `0` e restare con un solo zero. - Con $p = 4$, tutte le stringhe di lunghezza ≥ 4 contengono almeno i due zero e abbastanza `1` per potersi "pompizzare" nel blocco di `1`.

$$\boxed{p = 4 \text{ (minimo)}}$$

h) $10\,(11^*0)^*\,0$

Stringhe del tipo: inizio con "10", poi zero o più blocchi `11*0`, e infine un '0'. - La stringa più breve è "100", lunga 3. Essa non si può pompare con $p \leq 3$. Per es., se rimuovi un simbolo `0`, la stringa non termina più con '0'. - Con $p = 4$, le stringhe ≥ 4 si possono *sempre* pompare (di solito raddoppiando o rimuovendo `1` in quei blocchi).

$$\boxed{p = 4 \text{ (minimo)}}$$

i) $\{$`1011`$\}$

Linguaggio costituito da un'unica stringa "1011" di lunghezza 4. - Se $p \leq 4$, allora la stringa "1011" di lunghezza 4 *deve* poter essere pompata. Ma in un linguaggio di un solo elemento, pompare produce altre stringhe (es. rimuovere 1, o duplicarlo) che non esistono nel linguaggio. Contraddizione. - Se $p = 5$, allora non esistono stringhe di lunghezza ≥ 5 in L, quindi la condizione è vacuamente vera.

$$\boxed{p = 5 \text{ (minimo)}}$$

j) Σ^*

Tutte le stringhe sull'alfabeto (senza restrizioni). - Σ^* è enorme e *qualsiasi* stringa che pompi resta in Σ^*. Non c'è alcun vincolo. Quindi qualunque $p \geq 1$ funziona. - Per definizione, la *minimum pumping length* è 1.

$$\boxed{p = 1 \text{ (minimo)}}$$

Riepilogo

$$\begin{aligned} &a)\ p=4, \quad b)\ p=1, \quad c)\ p=1, \quad d)\ p=1, \\ &e)\ p=3, \quad f)\ p=1, \quad g)\ p=4, \quad h)\ p=4, \\ &i)\ p=5, \quad j)\ p=1. \end{aligned}$$