

1. **(12 punti)** Lo *xor bit a bit* è un'operazione binaria che prende due stringhe binarie di uguale lunghezza ed esegue lo xor logico su ogni coppia di bit corrispondenti. Il risultato è una stringa binaria in cui ogni posizione è 0 se i due bit sono uguali, 1 se sono diversi. Date due stringhe binarie  $x$  e  $y$  di uguale lunghezza,  $x \oplus y$  rappresenta lo xor bit a bit di  $x$  e  $y$ . Per esempio,  $0011 \oplus 0101 = 0110$ .

Dimostra che se  $L$  ed  $M$  sono linguaggi regolari sull'alfabeto  $\{0, 1\}$ , allora anche il seguente linguaggio è regolare:

$$L \oplus M = \{x \oplus y \mid x \in L, y \in M \text{ e } |x| = |y|\}.$$

2. **(12 punti)** Considera il linguaggio

$$L_2 = \{x\#y \mid x, y \in \{0, 1\}^* \text{ sono numeri binari tali che } x \text{ è un multiplo di } y\}.$$

L'alfabeto di questo linguaggio è  $\{0, 1, \#\}$ . Ad esempio,  $100\#10 \in L_2$  perché 4 è multiplo di 2, mentre  $0101\#10 \notin L_2$  perché 5 non è multiplo di 2. Dimostra che  $L_2$  non è regolare.

3. **(12 punti)** Sia  $\sigma$  un alfabeto finito. Data stringa  $w = w_1 \dots w_n$ , dove ogni  $w_i \in \Sigma$ , definiamo  $Rep(w) = w_1 w_1 w_2 w_2 \dots w_n w_n$ . Cioè,  $Rep(w)$  è la stringa ottenuta ripetendo ogni carattere di  $w$ . Dato un linguaggio  $L \subseteq \Sigma^*$ , definiamo il linguaggio

$$Rep(L) = \{Rep(w) \mid w \in L\}.$$

Dimostra che la classe dei linguaggi context free è chiusa per l'operazione  $Rep$ .