

1. **(12 punti)** Una macchina di Turing spreca-nastro è simile a una normale macchina di Turing deterministica a nastro singolo semi-infinito, ma può spostare la testina nella parte non ancora utilizzata del nastro. In particolare, se tutte le celle dopo la cella numero s del nastro sono vuote, e la cella s è non vuota, allora la testina può spostarsi nella cella numero $2s$. A ogni passo, la testina della TM spreca-nastro può spostarsi a sinistra di una cella (L), a destra di una cella (R) o dopo la parte non vuota del nastro (J).
 - (a) Dai una definizione formale della funzione di transizione di una TM spreca-nastro.
 - (b) Dimostra che le TM spreca-nastro riconoscono la classe dei linguaggi Turing-riconoscibili. Usa una descrizione a livello implementativo per definire le macchine di Turing.
2. **(12 punti)** Una variabile A in una grammatica context-free G è *persistente* se compare in ogni derivazione di ogni stringa w in $L(G)$. Data una grammatica context-free G e una variabile A , considera il problema di verificare se A è persistente.
 - (a) Formula questo problema come un linguaggio $PERSISTENT_{CFG}$.
 - (b) Dimostra che $PERSISTENT_{CFG}$ è decidibile.
3. **(12 punti)** Considera le stringhe sull'alfabeto $\Sigma = \{1, 2, \dots, 9\}$. Una stringa w di lunghezza n su Σ si dice *ordinata* se $w = w_1 w_2 \dots w_n$ e tutti i caratteri $w_1, w_2, \dots, w_n \in \Sigma$ sono tali che $w_1 \leq w_2 \leq \dots \leq w_n$. Ad esempio, la stringa 1112778 è ordinata, ma le stringhe 5531 e 44427 non lo sono (la stringa vuota viene considerata ordinata). Diciamo che una Turing machine è *ossessionata dall'ordinamento* se ogni stringa che accetta è ordinata (ma non è necessario che accetti tutte queste stringhe). Considera il problema di determinare se una TM con alfabeto $\Sigma = \{1, 2, \dots, 9\}$ è ossessionata dall'ordinamento.
 - (a) Formula questo problema come un linguaggio SO_{TM} .
 - (b) Dimostra che il linguaggio SO_{TM} è indecidibile.