Automi e Linguaggi Formali – 12/6/2024
Seconda prova intermedia – Secondo Turno

1. **(12 punti)** Una macchina di Turing con "save e restore" (SRTM) è una macchina di Turing deterministica a singolo nastro, che può salvare la configurazione corrente per poi ripristinarla in un momento successivo. Oltre alle normali operazioni di spostamento a sinistra e a destra, una SRTM può effettuare l'operazione di SAVE, che salva la configurazione corrente, e l'operazione di RESTORE che ripristina la configurazione salvata. L'operazione di SAVE sovrascrive una eventuale configurazione salvata in precedenza. Fare il RESTORE in assenza di configurazione salvata non ha effetto: si mantiene inalterata la configurazione corrente.

(a) Dai una definizione formale della funzione di transizione di una SRTM.

(b) Dimostra che le SRTM riconoscono la classe dei linguaggi Turing-riconoscibili. Usa una descrizione a livello implementativo per definire le macchine di Turing.

2. **(12 punti)** Un linguaggio $B \subseteq \{0,1\}^*$ è *palindromo* se ogni stringa in B è palindroma, cioè se $w = w^R$ per ogni $w \in B$. Ad esempio, sia $\{00, 11011, 1001\}$ che $\emptyset$ sono linguaggi palindromi, mentre $\{00, 10\}$ non lo è. Considera il problema di determinare se il linguaggio di una TM M è palindromo.

(a) Formula questo problema come un linguaggio PAL_{TM}.

(b) Dimostra che il linguaggio PAL_{TM} è indecidibile.

3. **(12 punti)** Un ristorante ha necessità di acquistare m ingredienti per preparare tutti i piatti che offre ai suoi clienti. Dato un insieme di n fornitori che forniscono tali ingredienti, si desidera determinare se esiste un piccolo insieme di fornitori che consentano di acquistare tutti gli ingredienti di cui il ristorante ha bisogno. Se ogni fornitore $i = 1, \dots, n$ fornisce un insieme $S_i \subseteq \{1, \dots, m\}$ di ingredienti, una *fornitura valida* è un insieme T di fornitori che, nel loro insieme, forniscono tutti gli m ingredienti: $\bigcup_{i \in T} S_i = \{1, \dots, m\}$. Definiamo il linguaggio

$$SUPPLY = \{\langle S_1, \dots, S_n, k\rangle \mid \text{esiste una fornitura valida } T \text{ di dimensione } |T| = k\}.$$

(a) Dimostra che $SUPPLY$ è un problema NP.

(b) Una copertura di vertici di un grafo G è un insieme di vertici T tale che ogni arco del grafo ha almeno una estremità su un vertice in T. Sappiamo che il linguaggio $VERTEX\text{-}COVER = \{\langle G, k\rangle \mid G \text{ ha una copertura di vertici di dimensione } k\}$ è NP-completo. Dimostra che $SUPPLY$ è NP-hard, usando $VERTEX\text{-}COVER$ come problema NP-hard di riferimento.