Automi e Linguaggi Formali – a.a. 2020/2021
Seconda prova intermedia

1. Una *tag-Turing machine* è una macchina di Turing con un singolo nastro e due testine: una testina può solo leggere, l'altra può solo scrivere. All'inizio della computazione la testina di lettura si trova sopra il primo simbolo dell'input e la testina di scrittura si trova sopra la cella vuota posta immediatamente dopo la stringa di input. Ad ogni transizione, la testina di lettura può spostarsi di una cella a destra o rimanere ferma, mentre la testina di scrittura deve scrivere un simbolo nella cella corrente e spostarsi di una cella a destra. Nessuna delle due testine può spostarsi a sinistra.

Dimostra che che le tag-Turing machine riconoscono la classe dei linguaggi Turing-riconoscibili.

2. Considera il linguaggio $\textsc{AlwaysDiverge} = \{\langle M\rangle \mid M$ è una TM tale che per ogni $w \in \Sigma^*$ la computazione di M su input w non termina$\}$.

(a) Dimostra che il $\textsc{AlwaysDiverge}$ è indecidibile.
(b) $\textsc{AlwaysDiverge}$ è un linguaggio Turing-riconoscibile, coTuring-riconoscibile oppure né Turing-riconoscibile né coTuring-riconoscibile? Giustifica la tua risposta.

3. Considera il problema di pianificare la disposizione dei posti a sedere per un matrimonio con n invitati. Per disporre gli invitati hai a disposizione k tavoli, che possono ospitare un numero arbitrario di invitati. Alcuni degli invitati sono amici tra di loro, altri sono rivali ed altri sono indifferenti l'uno all'altro. Il tuo obiettivo è di trovare una disposizione degli n invitati sui k tavoli che rispetti i seguenti requisiti:

- ogni invitato siede ad un solo tavolo;
- due invitati che sono amici devono sedere allo stesso tavolo;
- due invitati che sono rivali devono sedere a tavoli diversi.

Possiamo rappresentare l'input del problema con una tripla $\langle n, k, R\rangle$ dove:

- n è il numero di invitati
- k è il numero di tavoli
- R è una matrice $n \times n$ che descrive le relazioni tra gli invitati:

$$R[i,j] = \begin{cases} 1 & \text{se gli ospiti } i \text{ e } j \text{ sono amici} \\ -1 & \text{se gli ospiti } i \text{ e } j \text{ sono rivali} \\ 0 & \text{se gli ospiti } i \text{ e } j \text{ sono indifferenti} \end{cases}$$

e definire il seguente linguaggio:

$$\begin{aligned} \text{WSP} = \{\langle n, k, R\rangle \mid & \text{ esiste una disposizione di } n \text{ ospiti su } k \text{ tavoli} \\ & \text{che rispetta le relazioni tra invitati } R\} \end{aligned}$$

(a) Dimostra che WSP è un problema NP
(b) Dimostra che WSP è NP-hard, usando 3-COLOR come problema NP-hard di riferimento.[1]

[1] Vedi Esercizio 32 degli "Esercizi di preparazione all'esame" per la definizione del problema 3-COLOR.