Elementi di Teoria della Computazione

Classe: Resto_2 - Prof.ssa Marcella Anselmo

Tutorato

18/05/2022 ore 14:00-16:00

Seconda Esercitazione

a cura della dott.ssa Manuela Flores

Esercizio: Linguaggio associato al problema DOMINATING-SET
su elearning.informatica.unisa.it

Un sottoinsieme D di vertici di un grafo non orientato G = (V, E) è un **insieme dominante** per G se ogni vertice in V \ D è adiacente a un vertice in D (cioè i due vertici sono connessi mediante un arco in E).

Si consideri il seguente **problema di decisione**:

"Dati un grafo non orientato G = (V, E) e un intero positivo k,
esiste un insieme dominante D di cardinalità k? "

(a) Definire il linguaggio DOMINATING-SET associato a tale problema.

Lezione 24 pag. 32

Linguaggio associato a un problema di decisione

Sia $G = (V, E)$ un grafo non orientato, con insieme V di nodi e insieme E di archi. Un sottoinsieme V' di nodi di G è un *independent set* in G se per ogni u, v in V', la coppia (u, v) non è un arco, cioè u e v non sono adiacenti.

Definire il linguaggio *INDEPENDENT-SET* associato al seguente problema di decisione:

Sia G un grafo non orientato e k un intero positivo. G ha un independent set di cardinalità k?

$INDEPENDENT\text{-}SET = \{\langle G, k\rangle \mid G$ è un grafo non orientato, k è un intero positivo e G ha un independent set di cardinalità $k\}$

Esercizio 13: Linguaggio associato al problema DOMINATING-SET
su elearning.informatica.unisa.it

Un sottoinsieme D di vertici di un grafo non orientato G = (V, E) è un **insieme dominante** per G se ogni vertice in V \ D è adiacente a un vertice in D (cioè i due vertici sono connessi mediante un arco in E).
Si consideri il seguente **problema di decisione**:

"Dati un grafo non orientato G = (V, E) e un intero positivo k,
esiste un insieme dominante D di cardinalità k? "

(a) Definire il linguaggio DOMINATING-SET associato a tale problema.

Linguaggio associato a un problema di decisione

Sia $G = (V, E)$ un grafo non orientato, con insieme V di nodi e insieme E di archi. Un sottoinsieme V' di nodi di G è un *independent set* in G se per ogni u, v in V', la coppia (u, v) non è un arco, cioè u e v non sono adiacenti.

Definire il linguaggio *INDEPENDENT-SET* associato al seguente problema di decisione:

Sia G un grafo non orientato e k un intero positivo. G ha un independent set di cardinalità k?

$INDEPENDENT\text{-}SET = \{\langle G, k\rangle \mid G$ è un grafo non orientato, k è un intero positivo e G ha un independent set di cardinalità $k\}$

Esercizio: Problema accettazione di DFA

su elearning.informatica.unisa.it

(a) Si descriva la relazione esistente tra un problema di decisione e il linguaggio associato.

(b) Dato il problema

> **Problema dell'accettazione di un DFA:** Sia $\mathcal{B}$ un DFA e w una parola. L'automa $\mathcal{B}$ accetta w?

definire il linguaggio associato A_{DFA}, spiegando la corrispondenza.

(c) Si consideri l'automa finito deterministico $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$, dove $Q = \{q_0, q_1\}$, $\Sigma = \{a, b\}$, $F = \{q_1\}$ e δ è tale che $\delta(q_0, a) = q_0$, $\delta(q_0, b) = q_1$, $\delta(q_1, a) = \delta(q_1, b) = q_1$. Precisare quali delle seguenti stringhe sono elementi di A_{DFA}:
$\langle \mathcal{A}, aa \rangle$, $\langle \mathcal{A}, aba \rangle$, $\langle \mathcal{A}, 00 \rangle$.

(a) vedi slide successiva

Lezione 24 pag. 29

Linguaggio associato a un problema di decisione

- Mentre l'insieme delle istanze si divide in due sottoinsiemi (l'insieme delle istanze sì e quello delle istanze no), l'insieme delle stringhe su Σ si divide in **tre** sottoinsiemi:

  1. L'insieme delle stringhe w che codificano istanze con **risposta sì**.
  2. L'insieme delle stringhe w che codificano istanze con **risposta no**.
  3. L'insieme delle stringhe w che **non sono codifiche di istanze**.

- Il linguaggio L **associato** a un problema di decisione $\mathbb{P}$ è il linguaggio delle **codifiche** delle istanze che hanno **risposta sì**.

Lezione 25 pag. 36

Un problema indecidibile

$$A_{TM} = \{\langle M, w\rangle \mid M \text{ è una MdT e } M \text{ accetta } w\}$$

A_{TM} è il linguaggio associato al problema decisionale dell'accettazione di una macchina di Turing.

Teorema

Il linguaggio A_{TM} non è decidibile.

Esercizio: Riduzione da A_TM al complemento di EQ_TM
su elearning.informatica.unisa.it

Esercizio (riduzione da A_{TM} a $\overline{EQ_{TM}}$)

Sia f_{A-NE} la funzione di riduzione esibita per dimostrare che $A_{TM} \leq_m \overline{E_{TM}}$ e sia f_{E-EQ} la funzione di riduzione esibita per dimostrare che $E_{TM} \leq_m EQ_{TM}$.

E' possibile utilizzare f_{A-NE} e f_{E-EQ} per esibire una funzione di riduzione f_{A-NEQ} per dimostrare che $A_{TM} \leq_m \overline{EQ_{TM}}$?

Se sì, la funzione f_{A-NEQ} è la stessa di quella esibita nelle slide precedenti per dimostrare che $A_{TM} \leq_m \overline{EQ_{TM}}$?

Lezione 28 pag. 10

Problema del vuoto

$A_{TM} = \{\langle M, w\rangle \mid M \text{ è una TM e } w \in L(M)\}$
$E_{TM} = \{\langle M\rangle \mid M \text{ è una } TM \text{ e } L(M) = \emptyset\}$

$$A_{TM} \leq_m \overline{E_{TM}}$$

Consideriamo $f : \Sigma^* \to \Sigma^*$ tale che $f(\langle M, w\rangle) = \langle M_1\rangle$ dove

$$M_1 : x \to \boxed{\boxed{x =? w} \to \begin{cases} No & \to \\ Si & w \to \boxed{M} \to accetta \end{cases}} \to \begin{cases} rifiuta \\ accetta \end{cases}$$

$$\text{Quindi } L(M_1) = \begin{cases} \{w\} & \text{se } \langle M, w\rangle \in A_{TM} \\ \emptyset & \text{altrimenti} \end{cases}$$

Lezione 28 pag. 15

EQ_{TM} è indecidibile

$E_{TM} = \{\langle M \rangle \mid M \text{ è una } MdT \text{ e } L(M) = \emptyset\}$
$EQ_{TM} = \{\langle M_1, M_2 \rangle \mid M_1, M_2 \text{ sono MdT e } L(M_1) = L(M_2)\}$

$$E_{TM} \leq_m EQ_{TM}$$

Sia M_1 una macchina di Turing tale che $L(M_1) = \emptyset$.
$f : \langle M \rangle \to \langle M, M_1 \rangle$ è una riduzione di E_{TM} a EQ_{TM}.

Lezione 27 pag. 19

Teoremi

Teorema

$A \leq_m B$ *se e solo se* $\overline{A} \leq_m \overline{B}$.

Dimostrazione

Per ipotesi $A \leq_m B$, quindi esiste una riduzione di A a B.

Poiché f è una riduzione, f è calcolabile e inoltre

$$\forall w \in \Sigma^* \quad w \in A \Leftrightarrow f(w) \in B$$

Proviamo che f è anche una riduzione da $\overline{A}$ a $\overline{B}$.

Esercizio: Riduzione da A_TM al complemento di EQ_TM

su elearning.informatica.unisa.it

Esercizio (riduzione da A_{TM} a $\overline{EQ_{TM}}$)

Sia f_{A-NE} la funzione di riduzione esibita per dimostrare che $A_{TM} \leq_m \overline{E_{TM}}$ e sia f_{E-EQ} la funzione di riduzione esibita per dimostrare che $E_{TM} \leq_m EQ_{TM}$.

E' possibile utilizzare f_{A-NE} e f_{E-EQ} per esibire una funzione di riduzione f_{A-NEQ} per dimostrare che $A_{TM} \leq_m \overline{EQ_{TM}}$?

Se sì, la funzione f_{A-NEQ} è la stessa di quella esibita nelle slide precedenti per dimostrare che $A_{TM} \leq_m \overline{EQ_{TM}}$?

f: ⟨M,w⟩ → ⟨M1⟩ è una riduzione di A_{TM} a ‘E_{TM}.

$$M_1 : x \to \boxed{\boxed{x =? w} \to \begin{cases} No & \to \\ Si & w \to \boxed{M} \to accetta \end{cases}} \to \begin{cases} rifiuta \\ accetta \end{cases}$$

$$\text{Quindi } L(M_1) = \begin{cases} \{w\} & \text{se } \langle M, w\rangle \in A_{TM} \\ \emptyset & \text{altrimenti} \end{cases}$$

Sia M_1 una macchina di Turing tale che $L(M_1) = \emptyset$.
$f : \langle M\rangle \to \langle M, M_1\rangle$ è una riduzione di E_{TM} a EQ_{TM}.

Esercizio: Riduzione da A_TM al complemento di EQ_TM
su elearning.informatica.unisa.it

Esercizio (riduzione da A_{TM} a $\overline{EQ_{TM}}$)
Sia f_{A-NE} la funzione di riduzione esibita per dimostrare che $A_{TM} \leq_m \overline{E_{TM}}$ e sia f_{E-EQ} la funzione di riduzione esibita per dimostrare che $E_{TM} \leq_m EQ_{TM}$.

E' possibile utilizzare f_{A-NE} e f_{E-EQ} per esibire una funzione di riduzione f_{A-NEQ} per dimostrare che $A_{TM} \leq_m \overline{EQ_{TM}}$?

Se sì, la funzione f_{A-NEQ} è la stessa di quella esibita nelle slide precedenti per dimostrare che $A_{TM} \leq_m \overline{EQ_{TM}}$?

f: $\langle M,w \rangle \rightarrow \langle M1 \rangle$ è una riduzione di A_{TM} a ‘E_{TM}.

$$M_1 : x \rightarrow \boxed{\boxed{x =?w} \rightarrow \begin{cases} No & \rightarrow \\ Si & w \rightarrow \boxed{M} \rightarrow accetta \end{cases}} \rightarrow \begin{cases} rifiuta \\ accetta \end{cases}$$

$$\text{Quindi } L(M_1) = \begin{cases} \{w\} & \text{se } \langle M, w \rangle \in A_{TM} \\ \emptyset & \text{altrimenti} \end{cases}$$

Sia T1 una macchina di Turing tale che L(T1) =∅ ,
g: $\langle T \rangle \rightarrow \langle T, T1 \rangle$ è una riduzione di ‘E_{TM} a ‘EQ_{TM}.

Lezione 28 pag. 17

Riduzione da A_{TM} al complemento di EQ_{TM}

$f : \langle M, w\rangle \to \langle M_1, M_2\rangle$ è riduzione che prova $A_{TM} \leq_m EQ_{TM}$.

$$L(M_1) = \Sigma^*;\ L(M_2) = \begin{cases} \Sigma^* & \text{se } \langle M, w\rangle \in A_{TM} \\ \emptyset & \text{se } \langle M, w\rangle \notin A_{TM} \end{cases}$$

Possiamo modificare f per dimostrare che $A_{TM} \leq_m \overline{EQ_{TM}}$?

Lasciamo la stessa M_2 e cambiamo M_1 in $\mathbf{M_3}$.
$g : \langle M, w\rangle \to \langle M_3, M_2\rangle$

$$\mathbf{L(M_3)} = \emptyset;\ L(M_2) = \begin{cases} \boxed{\Sigma^*} & \text{se } \langle M, w\rangle \in A_{TM} \\ \emptyset & \text{se } \langle M, w\rangle \notin A_{TM} \end{cases}$$

Esercizio: Linguaggio L_x

Sia L_x l'insieme di stringhe binarie che rappresentano la codifica delle coppie (*M*,w) tali che *M* è una codifica di una macchina di Turing e w è la codifica di una stringa binaria di input accettata da tale macchina *M*.

Definire formalmente tale linguaggio e confrontarlo con quello riconosciuto da una MdT Universale U e con A_{TM}

Lezione 26 pag. 36

Macchina di Turing Universale

- Una MdT universale U simula la computazione di una qualsiasi MdT M
- U riceve in input una **rappresentazione** $\langle M, w\rangle$ di M e di un possibile input w di M
- È chiamata universale perchè la computazione di una qualsiasi MdT può essere simulata da U

$$\langle M, w\rangle \to \boxed{U} \to \begin{cases} \textit{accetta} & \text{se } M \text{ accetta } w \\ \textit{rifiuta} & \text{se } M \text{ rifiuta } w \\ \textit{non termina} & \text{se } M \text{ non termina} \end{cases}$$

Lezione 25 pag. 36

Un problema indecidibile

$$A_{TM} = \{\langle M, w\rangle \mid M \text{ è una MdT e } M \text{ accetta } w\}$$

A_{TM} è il linguaggio associato al problema decisionale dell'accettazione di una macchina di Turing.

Teorema

Il linguaggio A_{TM} non è decidibile.

Esercizio: Riduzione da HALT_TM
su elearning.informatica.unisa.it

Si consideri il linguaggio

$$L = \{\langle M\rangle \mid M \text{ è una MdT che si arresta su 11 e non si arresta su 00}\}.$$

Definire il linguaggio $HALT_{TM}$ e dimostrare che $HALT_{TM} \leq_m L$.

Lezione 27 pag. 17

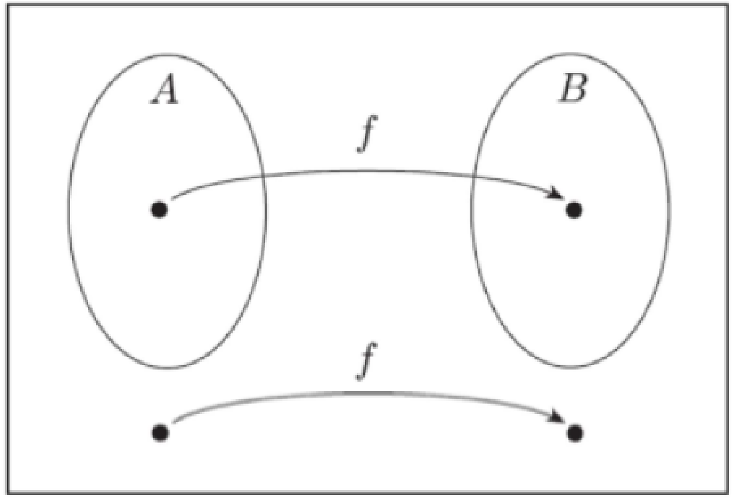
Riducibilità mediante funzione

Definizione

Un linguaggio $A \subseteq \Sigma^*$ è riducibile mediante funzione a un linguaggio $B \subseteq \Sigma^*$, e scriveremo $A \leq_m B$, se esiste una funzione calcolabile $f : \Sigma^* \to \Sigma^*$ tale che $\forall w \in \Sigma^*$

$$w \in A \Leftrightarrow f(w) \in B$$

La funzione f è chiamata una riduzione da A a B.



Lezione 27 pag. 25

Indecidibilità del problema della fermata

$$A_{TM} = \{\langle M, w\rangle \mid M \text{ è una MdT e } M \text{ accetta } w\}$$

$$\boxed{HALT_{TM} = \{\langle M, w\rangle \mid M \text{ è una MdT e } M \text{ si arresta su } w\}}$$

Teorema
$A_{TM} \leq_m HALT_{TM}$.

Dimostrazione
Occorre definire una funzione calcolabile $f\ : \Sigma^* \rightarrow \Sigma^*$ tale che $f(\langle M, w\rangle) = \langle M', w\rangle$ e

$$\langle M, w\rangle \in A_{TM} \quad \text{sse} \quad \langle M', w\rangle \in HALT_{TM}$$

Prossimo tutorato

Ci vediamo mercoledì prossimo ore 14-16

sempre su questo canale del Team…

…buono studio ☺