Elementi di Teoria della Computazione

Classe: Resto_2 - Prof.ssa Marcella Anselmo

Tutorato

11/05/2022 ore 14:00-16:00

Prima Esercitazione

a cura della dott.ssa Manuela Flores

Esercizio: Linguaggio associato al problema CALC
su elearning.informatica.unisa.it

Linguaggio associato al problema CALC

1. Fornire il concetto di problema di decisione e di linguaggio associato.

2. Si consideri il problema di decisione CALC

Input: $f : \Sigma* \rightarrow \Sigma*$ funzione.

Domanda: f è calcolabile?

(2.a) Fornire un esempio di istanza al problema CALC con risposta sì

(2.b) Definire il linguaggio C associato al problema di decisione CALC.

Lezione 24 pag. 3

Problemi di decisione

Un problema di decisione è un problema che ha come soluzione una risposta sì o no.

Esempi:

- Problema PRIMO: Dato un numero x intero e maggiore di uno, x è primo?
- Problema CONNESSO: Dato un grafo G, G è connesso?
- Problema A_{DFA}: Dato un DFA $\mathcal{B}$ e una stringa w, $\mathcal{B}$ accetta w?

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 24 pag. 30

Linguaggio associato a un problema di decisione

- Il linguaggio L **associato** a un problema di decisione $\mathbb{P}$ è il linguaggio delle **codifiche** delle istanze che hanno **risposta sì**.
- Se esiste una macchina di Turing che decide L il problema viene detto decidibile. Altrimenti il problema viene detto indecidibile.
- Se esiste una macchina di Turing che riconosce L il problema viene detto semidecidibile.
- In questo modo esprimiamo un problema computazionale come un problema di riconoscimento di un linguaggio. La macchina corrisponde a un algoritmo per il problema.

Esercizio: Linguaggio associato al problema CALC
su elearning.informatica.unisa.it

Linguaggio associato al problema CALC

1. Fornire il concetto di problema di decisione e di linguaggio associato.

2. Si consideri il problema di decisione CALC

Input: $f : \Sigma* \rightarrow \Sigma*$ funzione.

Domanda: f è calcolabile?

(2.a) Fornire un esempio di istanza al problema CALC con risposta sì

(2.b) Definire il linguaggio C associato al problema di decisione CALC.

Lezione 24 pag. 8

Problemi di decisione
istanze

Un problema di decisione considera elementi di un insieme.

Tali elementi sono anche chiamati le **istanze** del problema.

Quindi un istanza di un problema è un particolare input per quel problema.

- Esempio: 3, 4, 6 sono istanze per il problema PRIMO.
- Esempio: $(x_1 \vee x_2) \wedge (\overline{x_1} \vee x_2)$ è un istanza per il problema SAT.

Lezione 24 pag. 9

Problemi di decisione
istanze sì e istanze no

L'insieme delle istanze (per un problema di decisione) è unione del sottoinsieme delle **istanze con risposta sì** e del sottoinsieme delle **istanze con risposta no**.

- Esempio: 3 è un'istanza sì per il problema PRIMO,
  4 e 6 sono istanze no per il problema PRIMO.
- Esempio: $(x_1 \vee x_2) \wedge (\overline{x_1} \vee x_2)$ è un istanza sì per il problema SAT (prendere $x_1 = x_2 = 1$).

$$(x_1 \vee x_2) \wedge (x_1 \vee \overline{x_2}) \wedge (\overline{x_1} \vee x_2) \wedge (\overline{x_1} \vee \overline{x_2})$$

è un istanza no per il problema SAT.

Lezione 18 pag. 17

Funzioni calcolabili

Le MdT possono essere utilizzate per il calcolo di funzioni.

Definizione

Una funzione $f : \Sigma^* \to \Sigma^*$ *è calcolabile se esiste una macchina di Turing* $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$ *tale che*

$$\forall w \in \Sigma^* \quad q_0 w \to^* q_{accept} f(w)$$

Lezione 18 pag. 19

Funzioni calcolabili

- Definire una macchina di Turing deterministica M che calcoli la funzione $f(x, y) = x + y$, con x, y interi positivi.
- Si assuma che l'input sia della forma

$$w0z$$

dove w è la rappresentazione unaria di x e z è la rappresentazione unaria di y.

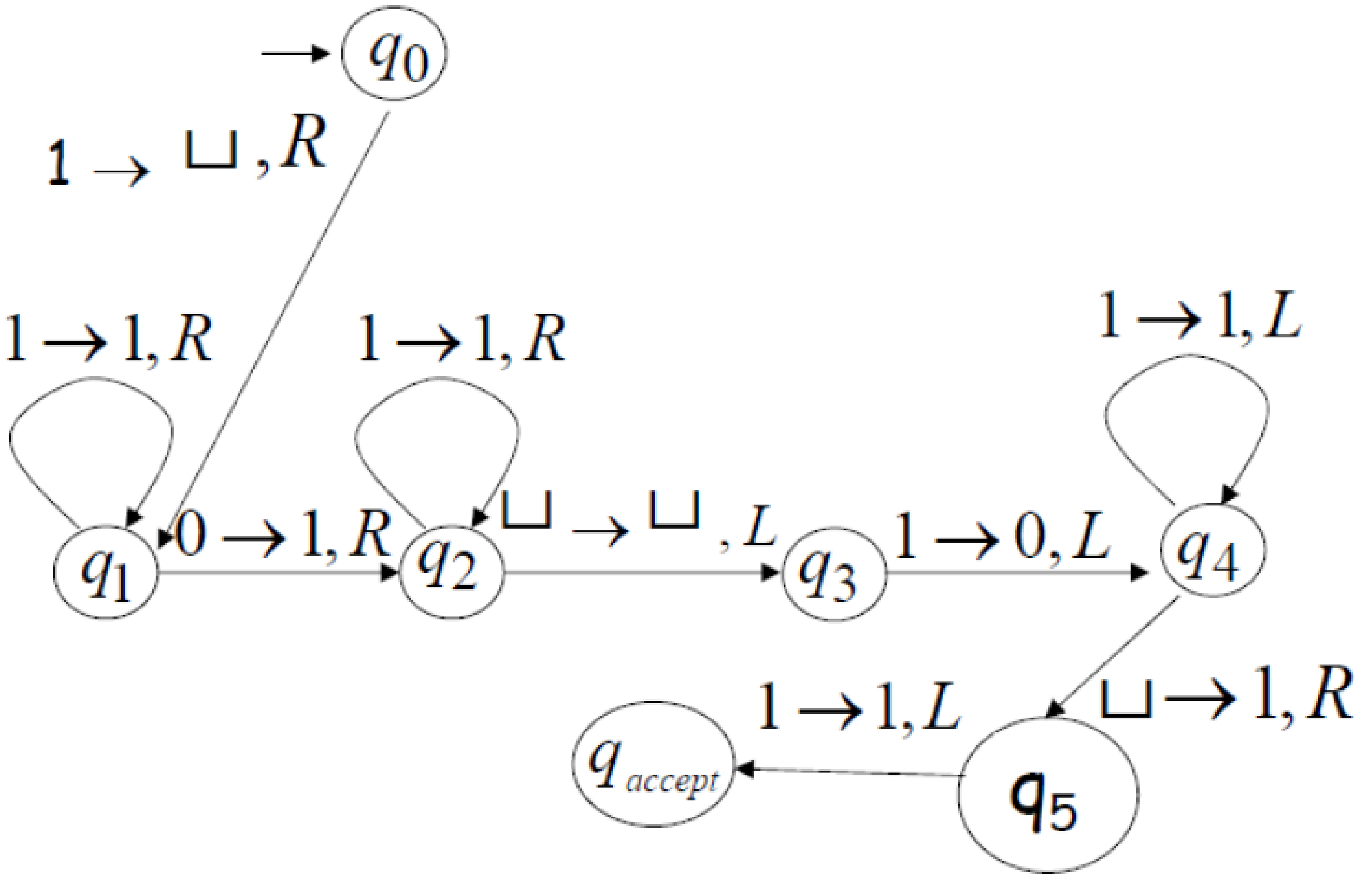
- Si definisca M in modo che l'output sia della forma

$$wz0$$

Per esempio se $x = 3$ e $y = 2$, l'input sarà 111011 e M dovrà fermarsi con 111110 sul nastro.

Lezione 18 pag. 20

Macchina di Turing per $f(x,y)=x+y$



Esercizio: Linguaggio associato al problema CALC
su elearning.informatica.unisa.it

Linguaggio associato al problema CALC

1. Fornire il concetto di problema di decisione e di linguaggio associato.

2. Si consideri il problema di decisione CALC

Input: $f : \Sigma* \to \Sigma*$ funzione.

Domanda: f è calcolabile?

(2.a) Fornire un esempio di istanza al problema CALC con risposta sì

(2.b) Definire il linguaggio C associato al problema di decisione CALC.

Lezione 24 pag. 37

Problemi di decisione

Il problema di decisione ispirato dal decimo problema di Hilbert

Input: *p polinomio a coefficienti interi.*
Domanda: *p ha una radice intera?*

Il corrispondente linguaggio associato è

$$\begin{aligned} D &= \{\langle p\rangle \mid p \text{ è un polinomio a coefficienti interi} \\ &\quad \text{avente una radice intera}\} \end{aligned}$$

Sappiamo che D non è decidibile e quindi il corrispondente problema di decisione non ammette una soluzione algoritmica, è indecidibile.

Esercizio: Linguaggio associato al problema CALC

su elearning.informatica.unisa.it

2. Si consideri il problema di decisione CALC
Input: f : $\Sigma* \rightarrow \Sigma*$ funzione.
Domanda: f è calcolabile?
(2.b) Definire il linguaggio C associato al problema di decisione CALC.

Problemi di decisione

Il problema di decisione ispirato dal decimo problema di Hilbert

Input: *p polinomio a coefficienti interi.*
Domanda: *p ha una radice intera?*

Il corrispondente linguaggio associato è

$$D = \{\langle p\rangle \mid p \text{ è un polinomio a coefficienti interi avente una radice intera}\}$$

Sappiamo che D non è decidibile e quindi il corrispondente problema di decisione non ammette una soluzione algoritmica, è indecidibile.

Esercizio: MdT per il successore in ordine radix
su elearning.informatica.unisa.it

MdT per il successore in ordine radix

Descrivere una MdT che data una stringa w sull'alfabeto {1, 2} calcoli la successiva di w nell'ordine radix.

Ordine radix = ordine per lunghezza e a parità di lunghezza ordine numerico secondo 1 < 2.

Esempio

w = 1221 successore di w = 1222

w' = 222 successore di w' = 1111

Esercizio: MdT per il successore in ordine radix
su elearning.informatica.unisa.it

MdT per il successore in ordine radix

Descrivere una MdT che data una stringa w sull'alfabeto {1, 2} calcoli la successiva di w nell'ordine radix.

Ordine radix = ordine per lunghezza e a parità di lunghezza ordine numerico secondo 1 < 2.

Esempio

w = 1221 successore di w = 1222

w' = 222 successore di w' = 1111

Esercizio: Computazione MdT (1)

su elearning.informatica.unisa.it

Computazione MdT (1)

Esempio: $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$, con
$Q = \{q_0, q_{accept}, q_{reject}\}$, $\Sigma = \{a, b\}$, $\Gamma = \{a, b, \sqcup\}$,
$\delta(q_0, a) = (q_0, a, R)$, $\delta(q_0, b) = (q_0, b, L)$,
$\delta(q_0, \sqcup) = (q_{accept}, \sqcup, L)$.

Mostrare la computazione (come sequenza di configurazioni) di M sull'input w = aaaa.

Lezione 16 pag. 36

Computazione di una MdT

Siano C, C' *configurazioni.*
$C \to^* C'$ *se esistono configurazioni* $C_1, \ldots, C_k$, $k \geq 1$ *tali che*

1. $C_1 = C$,
2. $C_i \to C_{i+1}$, *per* $i \in \{1, \ldots, k-1\}$,
   (ogni C_i *produce* C_{i+1}*)*
3. $C_k = C'$.

Diremo che $C \to^* C'$ *è una* **computazione** *(di lunghezza* $k-1$*).*

Quando k = 1?

Lezione 16 pag. 37

Configurazioni

Una configurazione C si dice:

- iniziale su input w se $C = q_0 w$, con w $\in \Sigma^*$
- di accettazione se $C = u\ q_{accept}\ v$
- di rifiuto se $C = u\ q_{reject}\ v$

Poiché non esistono transizioni da q_{accept} e da q_{reject}, allora le configurazioni di accettazione e di rifiuto sono dette configurazioni di arresto.

Lezione 16 pag. 35

Esempio

$$
\begin{aligned}
\delta(q_0, 0) &= (q_0, 0, R), \quad \delta(q_0, 1) = (q_0, 1, R),\\
\delta(q_0, \sqcup) &= (q_1, \sqcup, L),\\
\delta(q_1, 1) &= (q_2, 1, L), \quad \delta(q_2, 0) = (q_3, 0, L),\\
\delta(q_3, 1) &= (q_{accept}, 1, L)
\end{aligned}
$$

$$q_0 11 \to 1 q_0 1 \to 11 q_0 \to 1 q_1 1 \to q_2 11 \to q_{reject} 11$$

$$q_0 101 \to 1 q_0 01 \to 10 q_0 1 \to 101 q_0 \to 10 q_1 1 \to 1 q_2 01 \to q_3 101 \to q_{accept} 101$$

Esercizio: Transitività delle riduzioni
su elearning.informatica.unisa.it

Dimostrare che se $A \leq_m B$ e $B \leq_m C$, allora $A \leq_m C$. La dimostrazione deve essere costruttiva

Prossimo tutorato

Ci vediamo mercoledì prossimo ore 14-16

sempre su questo canale del Team…

…buono studio ☺