ELEMENTI D'INFORMATICA TEORICA

prova scritta dell'11 febbraio 2010

1. $A = \{a, b\}$, $L_1 = \{w \mid w \in A^* \text{ e } w$ <u>non</u> ha la forma $u\,bba\,v\}$
Esibire un DFA M tale che $L_1 = L(M)$ e scrivere un'espressione regolare per il linguaggio $A^* - L_1$.

2. Mostrare che $L_2 = \{0^\ell 1^m \mid \ell \geq m > 0\}$ non è regolare

3. Esibire il diagramma di transizione di stati di un automa a pila M che riconosca il linguaggio dell'esercizio 2.

4. Sia $A = \{0, 1\}$ e $L_3 = (A^* - L_2) - \{\varepsilon\}$ con L_2 il linguaggio dell'es. 2. Scrivere una grammatica CF Γ tale che $L_3 = L(\Gamma)$.

5. Sia $f(x)$ una funzione primitiva ricorsiva.
Mostrare che $g(x, z) = f^z(x) = \underbrace{f(\cdots f}_{z \text{ volte}}(x)\cdots)$
è una funzione primitiva ricorsiva (assumendo che $g(x, 0) = x$).

6. Sia $B = \{z \mid \exists x\, P(x, z)\}$ e sia P un predicato calcolabile. Mostrare che (a) B è r.e.;
(b) per una scelta opportuna di P, $\overline{B}$ non è r.e.

Elementi di Informatica teorica
Prova scritta del 19·3·2010

1. Sia $A = \{0,1\}$ e sia:
$$L \doteq \{w \mid w \in A^* \ \& \ w \text{ è la rappresentazione binaria, senza zeri iniziali, di un numero pari e maggiore di } 8\}$$
L è regolare? Dimostrare o refutare.

2. Sia $A = \{a,b\}$. Scrivere l'espressione regolare α tale che
$$\langle \alpha \rangle = \{w \mid w \in A^* \ \& \ a, b \text{ occorrono entrambe in } w\}$$

3. Sia $T = \{a,b,c\}$. Determinare una grammatica CF, Γ, tale che:
$$L(\Gamma) = \{w \mid |w| > 0 \ \& \ w = xcy \text{ con } x, y \in \{a,b\}^* \ \& \ |x| \text{ pari} \ \& \ |y| \text{ dispari}\}$$

4. Sia $\Psi_{\mathcal{P}}(x) = f(x)$ con $\#(\mathcal{P}) = z$ una funzione totale computabile e $C = \{z \mid \exists x [f(x) = z]\}$. Dimostrare che C è ricorsivamente numerabile.

5. Dimostrare che la funzione:
$$h(x) = \text{il più piccolo fattore primo, maggiore di 1, di } x.$$
è primitiva ricorsiva.

6. Sia $R = \{x \mid \forall t \neg STP(x, m, t)\}$. Trovare un numero m tale $R = \{0\}$.

Elementi di Informatica teorica

Prova scritta

20 settembre 2010

---

1. $A = \{0, 1\}$. $L = \{w \mid |w| \geq 2$ e il secondo simbolo di w è '0'$\}$

Esibire una espressione regolare α tale che $\langle\alpha\rangle = L$.

2. Mostrare che $L = \{a^n b^m \mid n, m \geq 0;\ n \neq m\}$ non è regolare. Sia inoltre $L_1 = \{a^n b^n \mid n \geq 0\}$. Mostrare che $L \cup L_1$ è regolare.

3. Mostrare che $L = \{a^n b^m \mid n > m;\ m \geq 0\}$ è un linguaggio CF.

4. Sia $P(x,y) = \begin{cases} 1 & \text{se vi sono almeno } y \text{ numeri primi} \leq x \\ 0 & \text{altrimenti} \end{cases}$

Mostrare che $P(x,y)$ è un predicato primitivo ricorsivo.

5. Per ogni insieme ricorsivo A ed ogni insieme B sottoinsieme di A $(B \subset A)$, B è ricorsivo. Dimostrare o confutare

6. Sia f definita nel modo seguente:

$$f(x) = \begin{cases} 1 & \text{se } \min_t STP(x,x,t) < k \\ 0 & \text{se } \min_t STP(x,x,t) \geq k \\ \uparrow & \text{altrimenti} \end{cases}$$

con k costante fissata. Mostrare che f è parzialmente calcolabile.

Elementi di Informatica teorica
Prova scritta del 3.3.2011

1. Sia $A = \{a, b\}$ e
$L \subseteq A^*$, $L = \{w \mid |w| \geq 4$ & il terzo simbolo di w è $b\}$.
Scrivere una espressione regolare α tale che $\langle\alpha\rangle = L$.

2. Sia Γ una grammatica con produzioni:
$$S \to XYX \mid XX, \quad Y \to XYX \mid XX,$$
$$X \to aa.$$
Descrivere $L(\Gamma)$ e porre Γ in CNF.

3. Sia Δ una grammatica con produzioni:
$$S \to XYX \mid XX, \quad Y \to XYX \mid XX,$$
$$X \to bb.$$
Mostrare che $L(\Gamma) \cdot L(\Delta)$, con Γ la grammatica dell'es. 2, è un linguaggio regolare.

4. Sia $L_1 \cup L_2$ un linguaggio regolare. Allora L_1 è regolare o L_2 è regolare. Dimostrare o confutare.

5. Sia $P(x,t)$ un predicato primitivo ricorsivo. Sia $g(x,y)$ il più grande numero $t \leq y$ tale che $P(x,t)$. Mostrare che $g(x,y)$ è una funzione primitiva ricorsiva.

6. Mostrare che non esiste nessuna funzione calcolabile f tale che: $f(x) = 1 \iff x \in W_x$.

Elementi di informatica teorica
prova scritta del 19.7.2017

1. Dimostrare o confutare le seguenti asserzioni:
   (a): Se $L_1 \subseteq L_2$ ed L_1 non è regolare allora L_2 non è regolare
   (b): Se L non è regolare allora $\bar{L} = A^* - L$ non è regolare
2. Sia $L = \{u \mid u = a^n b^{n+m}, \text{ con } n, m \geq 0\}$. Mostrare che L non è regolare
3. Sia G la grammatica CF che ha le regole di produzione
   $$S \to aS \mid aSbS \mid c$$
   Mostrare che G è ambigua
4. Sia $K_{<>} = \{<z_1, z_2> \mid z_1 \in W_{z_2} \text{ e } z_1 = z_2\}$. Mostrare che $K_{<>}$ è un insieme r.e.
5. Sia c una costante data. $D(x)$ è la somma dei divisori di x se $x > c$; altrimenti $D(x) = 0$. Mostrare che $D(x)$ è una funzione primitiva ricorsiva.
6. Sia $f(x_1, x_2) = \begin{cases} 1 & \text{se } STP(x_1, x_1, x_2) \\ 0 & \text{altrimenti} \end{cases}$

   f è una funzione calcolabile? Dimostrare o confutare.

Elementi di Informatica Teorica
prova scritta del 4.10.2017

Sia $A=\{a,b\}$, $L \subseteq A^*$, $L=\{w \mid$ la lunghezza di w è un multiplo di 3 o w contiene la stringa $bbbb\}$. Scrivere un'espressione regolare α tale che $\langle \alpha \rangle = L$.

Sia $L=\{a^n b^m a^m b^n \mid m,n>0\}$. Mostrare che L è CF.

Siano $L_1=\{a^n b^m \mid n,m>0 \text{ ed } m \neq n\}$ ed $L_2=\{a^n b^n \mid n \geq 0\}$. Dimostrare o confutare: (i) $L_1 \cap L_2$ è un linguaggio regolare; (ii) $L_1 \cup L_2$ è un linguaggio regolare.

Sia $g(x)$ una funzione calcolabile e sia $H_g=\{x \mid HALT(g(x),x)\}$. H_g è un insieme r.e.? Dimostrare o confutare

Sia $P(x)$ un predicato primitivo ricorsivo e sia $R(z,y)$ il predicato che è vero se e solo se vi sono almeno y numeri t, con $t \leq z$, tali che $P(t)$.
Mostrare che $R(z,y)$ è un predicato primitivo ricorsivo.

6. Siano A e B insiemi di numeri naturali. Dimostrare o confutare la seguente asserzione:
Se $A \subseteq B$ e B è un insieme r.e. allora anche A è r.e.

ELEMENTI D'INFORMATICA TEORICA
8 gennaio 2018

1. Mostrare che $L = \{a b^i c^j d \mid i > j > 0\}$ non è un linguaggio regolare

2. Mostrare che il linguaggio dell'es. 1 è context-free

3. (a) Siano L_1 ed L_2 linguaggi su $A = \{a, b\}$ e sia $L_1 \subsetneq L_2$. Dimostrare o confutare la seguente asserzione: se L_2 è regolare allora L_1 è regolare

(b) Siano L_1 ed L_2 linguaggi CF su $A = \{a, b, c\}$. Dimostrare o confutare: $L_1 \cap L_2$ è un linguaggio CF

4. Il predicato $R(x, y)$ è vero se e solo se vi sono esattamente y numeri primi $\leq x$. Mostrare che $R(x, y)$ è un predicato primitivo ricorsivo

5. Sia $K_0 = \{\langle z_1, z_2 \rangle \mid z_1 \in W_{z_2}\}$. Mostrare che K_0 è r.e.

6. Sia f una funzione di numeri naturali tale che

$$f(x)\downarrow \iff HALT(x, x) < 1$$

Mostrare che f non è una funzione parzialmente calcolabile

ELEMENTI DI INFORMATICA TEORICA
Prova scritta del 8.2.2018

1. Sia $A = \{a, b\}$, $L \subset A^*$, $L = \{u \mid u = a v a$ oppure $u = b v b\}$. Esibire il diagramma di transizione di un automa finito $\mathcal{M}$ tale che $L = L(\mathcal{M})$ e scrivere un'espressione regolare α tale che $\langle \alpha \rangle = L$.

2. Sia $A = \{a, b, c\}$, $L \subset A^*$ ed $L = \{W \mid W = W^R\}$. Scrivere le regole di una grammatica CF G tale che $L = L(G)$.

3. Siano L_1 ed L_2 linguaggi CF su uno stesso alfabeto A. Sia $L_3 = A^* - ((A^* - L_1) \cup (A^* - L_2))$. L_3 è CF? Dimostrare o confutare.

4. Sia $f(x) = 2x$ se x è un numero primo gemello e sia $f(x) = 2x + 1$ altrimenti. Mostrare che f è una funzione primitiva ricorsiva

5. Sia $R(x)$ un predicato calcolabile e sia

$$f(x) = \begin{cases} 0 & \text{se } R(x) \text{ è falso} \\ 1 & \text{se } R(x) \text{ è vero e } x \in W_x \\ \uparrow & \text{altrimenti.} \end{cases}$$

Mostrare che f è una funzione parzialmente calcolabile

6. Sia $A = \{x \mid f(x) \in B\}$, f una funzione calcolabile e B un insieme ricorsivo. Mostrare che A è ricorsivo.

ELEMENTI DI INFORMATICA TEORICA

prova scritta del 1° marzo 2018

1. Sia $A = \{a, b\}$, $L \subseteq A^*$, $L = \{ u \mid u = s_1 \cdot s_2 \cdot \ldots \cdot s_n,\ |u| \geq 3$ ed $s_{n-2} = b \}$. Scrivere un'espressione regolare α tale che $L = \langle \alpha \rangle$

2. Siano L_1 ed L_2 due linguaggi tali che L_1 è regolare, $L_1 \cup L_2$ è regolare ed $L_1 \cap L_2 = \emptyset$. Mostrare che L_2 è regolare.

3. Sia $L = \{ a^n b^m \mid n \neq m \text{ ed } n, m > 0 \}$. Mostrare che L è un linguaggio context-free.

4. Sia $P(x) = \begin{cases} x & \text{se } x \text{ è il prodotto di due numeri primi} \\ 0 & \text{altrimenti} \end{cases}$

Mostrare che $P(x)$ è una funzione primitiva ricorsiva

5. Sia $R(x)$ un predicato primitivo ricorsivo. Mostrare che

$$A_R(x) = \begin{cases} 1 & \text{se vi sono almeno } x \text{ numeri } n \text{ tali che } R(n) = 1 \\ \uparrow & \text{altrimenti} \end{cases}$$

è una funzione parzialmente calcolabile

~~6. Sia B = { [illegible]~~

6. Sia $B = \{ [z_1, z_2, z_3] \mid z_1 \in K \text{ e } z_2 \in W_{z_3} \}$.

Mostrare che B è un insieme r.e.

ELEMENTI DI INFORMATICA TEORICA
prova del 2.7.18

1. Sia $A=\{a,b\}$, $L\subseteq A^*$, $L=\{w \mid |w|\geq 4$ e il terzo simbolo di w è una a se w inizia con una b ed è b altrimenti$\}$. Scrivere un'espressione regolare α tale che $\langle\alpha\rangle = L$.

2. Mostrare che $L=\{a^n b^m c^k \mid k>m>n>0\}$ non è un linguaggio regolare.

3. Specificare due linguaggi L_1 ed L_2 tali che:
   (i) $L_1 \neq L_2$
   (ii) L_1 ed L_2 sono entrambi CF ma non regolari
   (iii) $L_1 \cap L_2$ è CF ma non regolare

4. Sia $g(x)$ la somma di tutti i numeri naturali $n \leq x$ tali che n è un numero primo o n è un numero pari. Mostrare che g è una funzione primitiva ricorsiva.

5. Sia $B=\{(x_1,\dots,x_m)\in \mathbb{N}^m \mid g(x_1,\dots,x_m)\downarrow\}$ con g funzione parzialmente calcolabile. Mostrare che
$$B' = \{[x_1,\dots,x_m]\in \mathbb{N} \mid (x_1,\dots,x_m)\in B\}$$
è un insieme r.e.

Mostrare che non esiste nessuna funzione calcol
$f(x)$ tale che $f(x)=0$ se e solo se $x \in$

ELEMENTI DI INFORMATICA TEORICA

prova scritta del 18.7.18

1. Sia $L = \{a^{2n} b^m \mid n, m \geq 0\}$ Esibire il diagramma di transizione di stato di un DFA M tale che $L = L(M)$ e scrivere un'espressione regolare α tale che $\langle \alpha \rangle = L$.

2. Siano L_1 ed L_2 dei linguaggi, con L_1 regolare, $L_1 \cup L_2$ regolare ed $L_1 \cap L_2 = \emptyset$. Dimostrare che L_2 è regolare.

3. Sia $L = \{a^n b^m c^J \mid n, m, J > 0 \text{ ed } n = m \text{ o } n = J\}$. Scrivere le regole di una grammatica CF G tale che $L = L(G)$ e mostrare che L è una grammatica ambigua.

4. Sia $f(x, y)$ il minimo comune multiplo di x ed y. Mostrare che f è una funzione primitiva ricorsiva.

5. Sia $R(x, z)$ un predicato calcolabile. Mostrare che $A = \{x \mid \text{esiste almeno un numero } z \text{ dispari tale che } R(x, z)\}$ è un insieme ricorsivamente enumerabile.

6. Sia

$$g(x) = \begin{cases} 0 & \text{se } x \in \mathbb{N} - K \\ \uparrow & \text{altrimenti} \end{cases}$$

g è una funzione parzialmente calcolabile? Dimostrare o confutare.

ELEMENTI DI INFORMATICA TEORICA

prova scritta del 12.9.18

1. Se $L_1 \cup L_2$ è un linguaggio regolare, allora L_1 è regolare o L_2 è regolare. VERO o FALSO? Motivare la risposta.

2. Sia $L = \{a^n b^m a^m b^n \mid m, n > 0\}$. L è un linguaggio CF? Dimostrare o confutare.

3. Sia $L = \{a^n b^m \mid n > m > 0\}$. Mostrare che L non è un linguaggio regolare.

Sia $A = \{\langle x, y\rangle \mid x \text{ e } y \text{ sono coprimi } (\text{primi tra loro})\}$. Mostrare che A è un insieme primitivo ricorsivo.

Siano f e g funzioni unarie e calcolabili. Sia $B = \{x \mid f(x) \in K \text{ o } g(x) \in K\}$. Mostrare che B è un insieme r.e.

Sia $h(x) = 0$ se $\exists y\, STP(x, x, y)$ e sia $h(x) = x+1$ altrimenti. Mostrare che h non è una funzione calcolabile.

ELEMENTI DI INFORMATICA TEORICA

prova scritta del 7.1.2019

1. Sia $A = \{a, b\}$ ed $L \subseteq A^*$, con $L = \{ w \mid |w| > 0$ e in ogni posizione dispari di w c'è una $a \}$. Scrivere un'espressione regolare α tale che $\langle \alpha \rangle = L$.

2. Siano B e C i due seguenti linguaggi su $A = \{a, b\}$:
   $B = \{ w \mid w$ ha lo stesso numero di a e di $b \}$;
   $C = \{ a^n b^m \mid n, m \geq 0 \}$.
   (i) $abba \in B$? (ii) descrivere $B \cap C$ e stabilire se B è regolare oppure no usando esclusivamente le proprietà di chiusura dei linguaggi regolari.

3. Scrivere le regole di una grammatica context-free G tale che $L(G) = \{ a^i b^j c^k \mid$ con $i = j$ o $j = k$ e $i, j, k > 0 \}$. Inoltre si esibiscano due derivazioni estreme a sinistra di G per la stringa $a^2 b^2 c^2$ e due alberi di derivazione diversi per la medesima stringa.

4. Siano B e C due insiemi primitivi ricorsivi. Mostrare che $B - C$ è un insieme primitivo ricorsivo.

5. Sia $B = \{ \langle z_1, z_2 \rangle \mid HALT(z_1, z_2) \}$. B è un insieme r.e.? Dimostrare o confutare.

6. Sia f una funzione unaria tale che $f(x) > k$, per qualche costante k, se e solo se $\exists z\, STP(x, x, z)$. La funzione f è una funzione calcolabile? Dimostrare o confutare.

ELEMENTI DI INFORMATICA TEORICA

prova scritta del 1° marzo 2019

1. Sia $\alpha = ((a\cdot(a\cup b))^* \cup ((a\cdot(a\cup b))^* \cdot a))$.
Scrivere il diagramma di transizione di un DFA M_0 tale che $\langle\alpha\rangle = L(M_0)$.

2. Descrivere due linguaggi L_1 ed L_2 e descrivere inoltre la loro unione nel rispetto delle seguenti condizioni:
(a) L_1 ed L_2 sono entrambi CF, ma nessuno dei due è regolare;
(b) $L_1 \neq L_2$;
(c) $L_1 \cup L_2$ non è regolare.

3. Scrivere le regole di una grammatica CF in forma normale di Chomsky G, tale che $L(G) = \{a^n b^m \mid n > m > 0\}$.

4. Sia $A = \{X \mid X$ è la somma di due numeri primi gemelli$\}$.
Mostrare che A è un insieme primitivo ricorsivo.

5. Sia $B = \{[z_1, z_2, z_3] \mid \langle z_1, z_2\rangle \in W_{z_3}\}$. Mostrare che B è un insieme ricorsivamente enumerabile.

6. Mostrare che non esiste nessuna funzione calcolabile $f(x)$ tale che $f(x) > x$ se e solo se $x \in K$.

ELEMENTI DI INFORMATICA TEORICA
prova scritta del 20.6.2019

1. Siano L_1 ed L_2 linguaggi regolari. Mostrare che $L_3 = L_1 - L_2$ è un linguaggio regolare

2. Mostrare che $L_4 = \{a^n b^m \mid m > n > 0\}$ non è un linguaggio regolare

3. Sia $L_5 = \{a^{2n} b^n c^m \mid n, m > 0\}$. Scrivere le regole di una grammatica CF G tale che $L_5 = L(G)$ e costruire in G sia una derivazione della stringa $a^4 b^2 c^1$ sia il corrispondente albero di derivazione -

4. Sia
$$P(x,y) = \begin{cases} 1 & \text{se esistono almeno } y \text{ numeri naturali } n \text{ tali che } n \leq x \text{ ed } n \text{ è un quadrato perfetto} \\ 0 & \text{altrimenti} \end{cases}$$
Mostrare che $P(x,y)$ è un predicato primitivo ricorsivo.

5. Sia $f(x) > x^2$ se e solo se $x \notin \overline{K}$. Mostrare che f non è una funzione calcolabile -

6. Sia $R = \{\langle r_1, r_2 \rangle \mid r_1 \in W_{(r_1 + r_2)}\}$. Mostrare che R è un insieme r.e.

ELEMENTI DI INFORMATICA TEORICA

prova scritta del 17.7.2019

1. Siano $A = \{a, b\}$, $L_1 \subseteq A^*$ ed $L_1 = \{u \in A^* \mid u$ non contiene tre a consecutive$\}$.
   (i) Scrivere un'espressione regolare α tale che $\langle \alpha \rangle = L_1$
   (ii) Disegnare il diagramma di transizione di un DFA M tale che $L_1 = L(M)$.
2. Scrivere le regole di una grammatica regolare G tale che $L(G)$ è il linguaggio L_1 dell'es. 1.
3. Siano $A = \{a, b\}$, $L_2 \subseteq A^*$, $L_2 = \{u \mid$ la lunghezza di u è un numero dispari e il suo simbolo centrale è una a$\}$
   (i) Scrivere le regole di una grammatica CF G tale che $L_2 = L(G)$.
   (ii) Disegnare un albero derivazione per una derivazione in G della stringa $abaaa \in L_2$.
4. Sia $A = \{(x_1, x_2) \mid f(x_1, x_2)\downarrow\}$ con f funzione parzialmente calcolabile. Mostrare che $B = \{\langle x_1, x_2 \rangle \mid (x_1, x_2) \in A\}$ è un insieme r.e.
5. Sia c una costante e sia $f(x)$ il più grande numero $y \leq x$ che è multiplo di c. Mostrare che f è una funzione primitiva ricorsiva.
6. Sia $f(x)$ una funzione che soddisfa la condizione:

$$f(x)\downarrow \text{ se e solo se } l(x) \notin W_{r(x)}$$

$f(x)$ è una funzione parzialmente calcolabile? Dimostrare o confutare.

ELEMENTI DI INFORMATICA TEORICA
prova scritta del 6.9.2019

√1. Sia $A = \{a, b\}$ e sia $L = \{u \mid u \in A^*$ e u contiene almeno due occorrenze della stringa aa che siano separate da almeno una $b\}$. Scrivere un'espressione regolare α tale che $\langle\alpha\rangle = L$.

√2. Mostrare che non esiste nessuna espressione regolare α tale che $\langle\alpha\rangle = \{a^n b^m \mid n > m > 0\}$.

~~3~~. Sia L il linguaggio su $A = \{a, b\}$ tale che $L = \{w \mid w = w^R\}$: (1) Scrivere le regole di una grammatica CF G tale che $L = L(G)$; (2) Spiegare perché G non è una grammatica regolare.

√4. Sia $P(z)$ un predicato primitivo ricorsivo e sia $Q(x)$ il predicato che è vero se e solo se esiste esattamente un numero naturale m tale che $m \leq x$ e $P(m)$. Mostrare che $Q(x)$ è un predicato primitivo ricorsivo.

√5. Sia $B = \{[z_1, z_2, z_3, z_4] \mid \langle z_1, z_2\rangle \in W_{\langle z_3, z_4\rangle}\}$. Mostrare che B è ricorsivamente enumerabile.

√6. Sia dato un insieme $A = \{a_1, \dots, a_n\}$ tale che, per $1 \leq i \leq n$, $\Phi(a_i, a_i)\uparrow$. Sia

$$f(x) = \begin{cases} 1 & \text{se } \Phi(x,x)\downarrow \\ 0 & \text{se } x \in A \\ \uparrow & \text{altrimenti} \end{cases}$$

Mostrare che f è una funzione parzialmente calcolabile

ELEMENTI DI INFORMATICA TEORICA

prova scritta del 15.10.2019

1. (a) Disegnare il diagramma di transizione di stato di un NDFA $\mathcal{M}$ sull'alfabeto $A = \{a, b\}$ che accetta tutte e sole le parole che terminano con bab oppure con aaba. (b) Scrivere un'espressione regolare α tale che $\langle \alpha \rangle = L(\mathcal{M})$.

2. (a) Scrivere le regole di una grammatica context-free G tale che $L(G) = \{a^n b^{2n} \mid n > 0\}$. (b) Disegnare l'albero di derivazione per una derivazione in G della parola $a^2 b^4$.

3. Sia $f(x) = \begin{cases} 0 & \text{se } x \in W_x \\ \uparrow & \text{altrimenti} \end{cases}$

   f è parzialmente calcolabile? Dimostrare o confutare

4. Sia $g(x) = \begin{cases} 0 & \text{se } \neg \exists y\ STP(x, x, y) \\ \uparrow & \text{altrimenti} \end{cases}$

   g è parzialmente calcolabile? Dimostrare o confutare

5. Sia
$$h(x) = \begin{cases} 0 & \text{se } x \text{ è un numero primo} \\ 1 & \text{se } x \text{ non è primo ed è pari} \\ x & \text{se } x \text{ non è primo ed è dispari} \end{cases}$$
   Mostrare che h è una funzione primitiva ricorsiva.

6. Sia A un insieme r.e. Sia $B = \{x \mid 2x \in A\}$.
   Mostrare che B è un insieme r.e.

Elementi di informatica teorica

Prova scritta

21 gennaio 2020

1. Sia $L_1 = \{a^n b^{n+m} a^m \mid n, m > 0\}$. Scrivere le regole di una grammatica context-free G tale che $L_1 = L(G)$ e costruire in G un albero di derivazione per la stringa $abbbaa$.

2. Sia L_2 il linguaggio formato dalle stringhe contenenti un numero pari di a ed esattamente una b. Scrivere il diagramma delle transizioni di stato di un DFA $\mathcal{M}$ tale che $L_2 = L(\mathcal{M})$, e un'espressione regolare α tale che $L_2 = \langle \alpha \rangle$.

3. Sia L un linguaggio regolare. Dimostrare che $L^+ := \{w_1 w_2 \cdots w_n \mid n \geq 1,\ w_1, \ldots, w_n \in L\}$ è anch'esso regolare. ?

4. Sia $\omega(n)$ il numero di fattori primi distinti di un numero naturale n (con $\omega(1) = 0$, e $\omega(0) = 0$ per convenzione). Dimostrare che la funzione ω è ricorsiva primitiva.

5. Sia f una funzione ricorsiva primitiva, e

$$B = \left\{ 1 + \sum_{n=0}^{x} f(n) \mid x \in \mathbb{N} \right\}.$$

Dimostrare che B è r.e.

6. Sia $H = \{\langle x, y \rangle \mid x \in W_{2y}\}$. L'insieme H è r.e.? Dimostrare o confutare.

Elementi di informatica teorica

Prova scritta

20 gennaio 2022

1. Scrivere le regole di una grammatica context-free G che generi il linguaggio delle $w \in \{a, b\}^*$ di lunghezza dispari, con una a come lettera centrale e contenenti almeno una b.
2. Determinare due linguaggi L e L' sull'alfabeto $\{a, b\}$ con L regolare, $L \cap L'$ *non* regolare e $L' \not\subseteq L$.
3. Disegnare il diagramma di transizione di un DFA $\mathcal{M}$ tale che

$$L(\mathcal{M}) = \langle(((\varepsilon \cup b) \cdot (a \cup (a \cdot b))^*)\rangle .$$

4. Per $n \geq 0$, sia $f(n)$ il numero di divisori primi distinti di $n + 1$. Mostrare che f è una funzione ricorsiva primitiva.
5. Dimostrare che non esiste alcuna funzione calcolabile g tale che:

$$g(x) \in \{0, 1\} \iff \Phi(x, x)\downarrow .$$

6. Mostrare che $B = \{\langle k, \langle m, n\rangle\rangle \mid k \in W_m \cap W_n\}$ è ricorsivamente enumerabile.

Elementi di informatica teorica

Prova scritta

16 febbraio 2022

1. Sia $A = \{a, b\}$, $L \subset A^*$,

   $$L = \{w \mid \text{la lunghezza di } w \text{ è un multiplo di 3 e } w \text{ termina con } ba \text{ o con } ab\}.$$

   Scrivere un'espressione regolare α tale che $\langle\alpha\rangle = L$.

2. Sia $L_1 = \{a^n b^m \mid n, m > 0 \ \text{ e } \ n \neq m\}$. Scrivere le regole di una grammatica CF G tale che $L(G) = L_1^*$.

3. Mostrare che il linguaggio L_1 dell'esercizio 2 non è regolare.

4. Sia $S = \{\langle m, n\rangle \mid m \text{ ed } n \text{ sono coprimi}\}$. Mostrare che S è un insieme ricorsivo primitivo. (NB: Due numeri naturali si dicono *coprimi* se il loro massimo comun divisore è 1.)

5. Specificare un predicato calcolabile $P(x, y)$ tale che $B = \{x \mid \forall y\ P(x, y)\}$ non è un insieme r.e. e $\overline{B}$ è un insieme r.e.

6. Sia $f(x)$ una funzione calcolabile e sia

   $$C = \{x \mid f(x) \in D\}$$

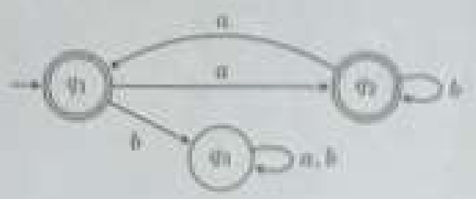
   dove D è un insieme ricorsivo. Mostrare che anche C è un insieme ricorsivo.

Elementi di informatica teorica

Prova scritta

16 marzo 2022

1. Sia $\mathcal{M}$ il DFA il cui diagramma di transizione è riprodotto in figura. Scrivere un'espressione regolare α tale che $L(\mathcal{M}) = \langle \alpha \rangle$.



2. Siano $L_1 = \langle ((a \cdot a^*) \cdot b^*) \rangle$ e

$$L_2 = \{w \in \{a, b\}^* \mid w \text{ inizia per } b \text{ e contiene almeno una } a\}.$$

Scrivere le regole di una grammatica **regolare** G tale che $L(G) = L_1 \cup L_2$.

3. Specificare due linguaggi non regolari L e L', sull'alfabeto $\{a, b\}$, tali che $L \cap L'$ sia infinito e regolare.

4. Sia $B \subseteq \mathbb{N}$ un insieme ricorsivo primitivo, e Q il predicato definito da

$$Q(x) \iff \text{tutti i divisori di } x + 1 \text{ appartengono a } B.$$

Mostrare che anche Q è ricorsivo primitivo.

5. La funzione f definita da

$$f(x) = \begin{cases} x + 1 & \text{se } \neg(\exists z : \mathrm{STP}(x, x, z)) \\ \uparrow & \text{altrimenti.} \end{cases}$$

è parzialmente calcolabile? Dimostrare o confutare.

6. Mostrare che l'insieme $\{[i, j, k] \mid \mathrm{HALT}(i, k) \wedge \mathrm{HALT}(j, k)\}$ è ricorsivamente enumerabile.

ESERCIZIO 1

A

Un linguaggio A si dice T-r se esiste una macchina di Turing M che lo riconosce. Dire che è un linguaggio è riconosciuto da una Turing machine M, vuol dire che tutte le stringhe w appart. A sono accettate da M, quindi L(M) = A.

Una Tm M accetta l'input w se esiste una sequenza di configurazioni c1, c2,…ck dove:

- C1 è una configurazione iniziale (quindi della forma q0w, con 10 stato accettante la testina nell'estrema sinistra del nastro.
- Ck è una configurazione di accettazione(contiene lo stato qacc)
- Ogni configurazione c_i -1 segue ci

Una configurazione non è nient'altro che una tripla formata da uno stato della macchina, la posizione della sua testina sul nastro e la stringa corrente presente sul nastro.

Un linguaggio A si dice co-T-r se è il complemento di un linguaggio T-R.

Un linguaggio che è sia TR che Co-TR si dice decidibile. Possiamo dimostrarlo nel seguente modo:

Siano M1 il riconoscitore di un linguaggio A e sia M2 il riconosciutore del linguaggio !A, possiamo costruire una macchina M che decide il linguaggio A.

M prende in input una stringa w:

- Esegue in simultanea M1 ed M2 su w
- Se M1 accetta, M accetta, se M2 accetta, M rifiuta.

Dato che M si ferma su tutti gli input, perché una stringa o appart. Ad A o a !A, allora è un decisore.

B

Si, un esempio è il linguaggio EQ_tm.

EQ_tm = {<M1,M2>| sono due TM t.c. L(M1) = L(M2)}

Sappiamo che se A<B ed A non è TR allora B non è TR, e sappiamo anche che A < B = !A < !B.

Dimostriamo che EQ_TM non è TR iniziando con il dimostrare che A_TM è riducibile mediante funzione ad !EQ_tm. Avremo quindi che Atm < !EQ => !Atm < EQ.

La funzione di riduzione è la seguente.

F prende in input <M,w>, M TM e w stringa:

- Costruisce due TM M1 ed M2.
- M1 rifiuta su tutti gli input
- M2 su tutti gli input esegue M su w ed accetta se M accetta
- Ritorna <M1,M2>

Dato che M1 non accetta nessuna stringa mentre M2 accetta tutte le stringhe se M accetta w, le due macchine non sono equivalenti. Sono però equivalenti quando M non accetta w, perché entrambe non accettano nulla. Per questo F riduce A_TM ad !EQ_tm.

Viceversa, per dimostrare che !EQ non è TR riduciamo mediante funzione A_tm al suo complemento, quindi Avremo che !Atm < EQ => !Atm < !EQ.

La funzione di riduzione è la seguente.

G prende in input <M,w>, M TM e w stringa:

- Costruisce due TM M1 ed M2.
- M1 accetta su tutti gli input
- M2 su tutti gli input esegue M su w ed accetta se M accetta
- Ritorna <M1,M2>

Dato che M1 accetta tutte le stringhe mentre M2 accetta tutte le stringhe se M accetta w, le due macchine sono equivalenti. Non sono però equivalenti quando M non accetta w, perché M1 accetta tutto mentre M2 non accetta nulla. Per questo G riduce A_TM ad EQ_tm.

C

Si e possiamo dimostrarlo pe costruzione.

Siano L1 ed L2 due linguaggi decidibili, e siano M1 ed M2 le tm che li decidono. Costruiamo una TM M che decide il linguaggio L = L1 intersecato L2.

M su input w:

- Esegue M1 ed M2 su w simultaneamente. Se entrambe accettano, accetta. Se M1 rifiuta, rifiuta. Se M2 rifiuta, rifiuta.

Dato che M1 ed M2 si fermeranno su ogni imput, in quanto decisori, sappiamo che M si fermerà anch'esso su tutti gli input e quindi è anch'esso un decisore.

D

Un LBA è un tipo di TM in cui la testina non può andare oltre le celle di nastro occupate dall'input. Quando la testina tenta di spostarsi al di fuori della parte di nastro che non contiene input, essa non si muove. Avendo una quantità di memoria limitata (che dipende dalla lunghezza della stringa), esso avrà un numero limitato di configurazioni differenti per un nastro lungo n.

Il numero di configurazioni distinte è dato da qng^n dove q è il numero di stati, n la lunghezza della stringa in input e g il numero di simboli nell'alfabeto del nastro (con g^n indichiamo tutte le possibili stringhe sul nastro di lunghezza n e con g simboli nell'alfabeto del nastro).

E

A_LBA = {<M, w> | M è una LBA che accetta w}

Si, è decidibile. Possiamo dimostrare che è decidibile costruendo un decisore per questo linguaggio. Sia S il decisore.

S prende in input <M,w> con M LBA e w stringa

- Simula M su input w per qng^n passi o fino a che M non raggiunge uno stato di halting.
- Se M si ferma in uno stato accettante, accetta. Se M si ferma in uno stato rifiutante, rifiuta. Se M non si ferma, quindi va in loop, rifiuta.

È facile determinare se M cicla w perché se una delle qng^n configurazioni si ripete, allora vuol dire che M è entrato in un loop.

ESERCIZIO 2

Sia M la TM multinastro che decide L.

M su input x:

- **Esegue V su input x presente sul primo nastro, se V accetta proseguo. Se V rifiuta, rifiuto.**
- **Inserisco all'inizio secondo nastro "#", poi copio dall'inizio del primo nastro sul secondo nastro fino al simbolo "-"(escluso).**
- **Copio sul terzo nastro dalla cella successiva al simbolo "-" fino al primo "blank" (escluso).**
- **Eseguo la TM S sul secondo nastro, se S rifiuta, salto al passo 6.**
- **Eseguo D sul terzo nastro, se D accetta, ripeto dal passo 4, se D rifiuta, M rifiuta.**
- **Eseguo C su input z(la stringa presente sul 3 nastro), se C accetta, M accetta, se C rifiuta, M rifiuta.**

V verifca la correttezza dell'input. Se V accetta, l'input è formattato correttamente.

V su input x:

- Se il simbolo non è "0" o "1" rifiuta.
- Se il simbolo è "0" o "1" sposto la testina a dx e ripeto il passo 2.
- Se il simbolo è "-", mi sposto alla sua dx.
- Se il simbolo è "a", sposto la testina a dx e ripeto il passo 4, se non è "a" rifiuto.
- Se il simoblo è "blank" accetto, altrimenti rifiuto.

S decrementa il numero binario.

S su input x:

- Sposto la testina all'estrema destra del nastro, fino al simbolo precedente al primo "blank".
- Se il simbolo è 1, lo rimpiazzo con 0 e accetto. Se il simbolo è 0, lo rimpiazzo con 1, mi sposto a sx, ripeto il passo 2. Se il simbolo è "#", rifiuto.

C verifica se esistono altre. Se esistono, rifiuta, perché w non è un rappresentazione binaria di n (numero di a)

C su input z:

- Se il simbolo è “X” sposto la testina a dx e ripeto il passo 1.
- Se il simbolo è “blank”, accetto.
- Se il simbolo è “a” rifiuto.

D decrementa il numero di a.

D su input z:

- Se il simbolo è una “x”, sposto la testina a dx e ripeto.
- Se il simbolo è una “a”, la rimpiazzo con la “x” e accetto.
- Se simbolo è un “blank”, rifiuto.

ESERCIZIO 3

Per poter dimostrare che L sia indecidibile, utilizziamo il metodo per riducibilità. Questo metodo utilizza una riduzione. Una riduzione è un modo di convertire un problema in un altro problema della quale è nota la soluzione e che potrà poi essere utilizzata per risolvere il primo problema.

Supponiamo p.a. che esista un decisore R per L. Costruiamo un decisore S per A_tm.

S su input <M,w> dove M è una TM e w è una stringa:

- **Usa la descrizione di M e w, per costruire una TM M1(Modifica di M).**
- **Esegue R su input <M1>**
- **Se R accetta, S accetta, Se R rifiuta, S rifiuta.**

M1 è una TM che non sarà mai eseguita, verrà solo utilizzata per determinare se M accetta w. Ha all’interno dei propri stati la stringa 001.

M1 è costruita in modo tale che accetta 001 se e solo se M accetta w. Se M non accetta w, M1 non accetta 001.

M1 su input x:

- **Se x != 001, rifiuta.**
- **Se x = 001, esegue M su w ed accetta se M accetta w. Rifiuta se M rifiuta.**

Se M accetta w, R accetta <M1> perché la stringa 001 appartiene al linguaggio di M1, e quindi S accetta. Se invece M non accetta w, R non accetta <M1> perché la stringa 001 non appartiene al linguaggio di M1, e quindi S rifiuta.

Prova Edit del 23/06/2023

[32 punti]

[Tempo: 2 ore e 15 min]

1) [4] Disegnare un DFA che legga un intero non negativo scritto in binario, con il bit meno significativo per primo, e che accetti esattamente quando l'ingresso è divisibile per quattro. Specificate quali sono Q, Σ, δ, q0 e F per il DFA.
2) [6] Sia L il linguaggio di tutte le stringhe dell'alfabeto Σ = {a, b} che non contengono la sottostringa "aa" e non contiene la sottostringa "bb" e ha lunghezza minima 1. Disegnate un NFA che riconosca L e utilizzi solo tre stati. Scrivere poi il DFA ottenuto per sottoinsiemi
3) [4] Sia Σ = {a} e $L = \{a^{k^3} \mid k \geq 0\}$. Si provi che L non è regolare.
4) [2] Intuitivamente, il linguaggio $L = \{a^n\, b^m\colon n \geq 100 \text{ e } m < 100\}$ è regolare?
5) [4] Si descriva la prova del seguente lemma: Se un pushdown riconosce un linguaggio L allora L è un context-free (Lemma 2.27 del Sipser)
6) [4] Si dia una CFG per il seguente linguaggio $L = \{wa^nb^nw^R\colon n \geq 0\}$ dove w è una stringa, anche vuota dell'alfabeto Σ= {a,b}. Si dia la derivazione della stringa abaabbba ottenuta con la grammatica data
7) [4] Dopo una breve ma formale introduzione della macchina di Turing deteriministica e nondeterministica, si introducano brevemente le LBA (tutto in massimo 2 pagine).
8) [4] Dimostrare che il problema di determinare se due macchine di Turing accettano lo stesso linguaggio è indecidibile

Prova Edit del 20/07/2023

[32 punti]

[Tempo: 2 ore e 15 min]

1) [8] Disegnare un NFA sull'alfabeto $\Sigma = \{0, 1\}$ che
   a. accetta il Linguaggio L delle stringhe che hanno il terzultimo e l'ultimo simbolo uguale.
   b. Sia L' il linguaggio delle stringhe che termina con 1. Si disegni l'NFA per L intersecato con L'.
   c. Si converta l'ultimo NFA in un DFA utilizzando la costruzione per sottoinsieme. Consiglio: usare una versione minimale dell'NFA e del DFA.
   d. Dell'ultimo automa ottenuto, si costruisca l'espressione regolar del linguaggio.
2) [6] Disegnare un PDA per i seguenti due linguaggi
   a. $L1= \{a^i b^j c^k \mid i, j, k \geq 0, \text{ and } i = j \text{ or } j = k \}$
   b. $L2= \{a^i b^j c^k \mid i, j, k \geq 0, \text{ and } i + k=j\}$.
3) [4] Scrivere la CFG che genera L2
4) [6] Dopo aver formalizzato il pumping lemma, si provi utilizzando lo stesso che il seguente linguaggio non è regolare: $\{0^n10^n10^n \mid n \geq 0 \}$
5) [4] Dopo una breve <u>ma formale</u> introduzione della macchina di Turing deterministica e nondeterministica, si parli del problema della fermata.
6) [4] Si parli brevemente della numerabilità/non-numerabilità di Q e di R.