LINGUAGGI FORMALI
Esercizi

✎ ✎ ✎ ✎ ✎ ✎

Nicola Fanizzi

LINGUAGGI DI PROGRAMMAZIONE
Corso di Informatica T.P.S.

Dipartimento di Informatica
Università di Bari "Aldo Moro"

[2014/01/28-13:30:23]

Indice

1

Introduzione ai Linguaggi Formali

1.1 Operazioni

1.1.1 – Dato $L = \{\mathsf{a}^n\mathsf{b}^n \mid n > 0\}$, determinare L^2 e L^3 ✍

1.1.2 – Dato $L = \{00, 01, 100\}$, stabilire se appartengono a L^* le stringhe

- 000010000
- 100000100001
- 00000100

Quali di queste appartengono a L^4 ? ✍

Suggerimento: scomporre nelle varie sotto-stringhe

1.1.3 – Dato $L = \{0^n1^n \mid n > 0\}$, determinare $\bar{L}$ ✍

Suggerimento: scomporre in sotto-insiemi (disgiunti) definiti anche ricorsivamente

1.2 Grammatiche

1.2.1 – Data $G = (\Sigma, V, S, P)$, con

- $\Sigma = \{\mathsf{a}, \mathsf{b}\}$
- $V = \{S, B\}$
- $P = \{S \longrightarrow \mathsf{a}B,\ B \longrightarrow \mathsf{b}S|\mathsf{b}\}$

indicare

1. il tipo di G
2. $\mathbf{L}(G)$ ✍

1.2.2 – Data $G = (\Sigma, V, S, P)$, con

- $\Sigma = \{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d}\}$
- $V = \{S, E, F\}$
- $P = \{S \longrightarrow ESF|EF,\ E \longrightarrow \mathsf{ab},\ F \longrightarrow \mathsf{cd}\}$

indicare

1. il tipo di G
2. $\mathbf{L}(G)$ ✍

1.2.3 – Data $G = (\{\mathsf{a}\}, \{S\}, S, P)$, con

- $P = \{S \longrightarrow \mathsf{a}S\mathsf{a}|\mathsf{aa}|\mathsf{a}\}$

indicare

1. il tipo di G
2. $\mathbf{L}(G)$ ed il suo tipo ✍

1.2.4 – Dato $G = (\{\mathsf{a}, \mathsf{b}, \mathsf{c}\}, \{S, A\}, S, P)$ dove

- $P = \{S \longrightarrow S\mathsf{c}|A,\ A \longrightarrow \mathsf{a}A\mathsf{b}|\mathsf{ab}\}$

indicare $\mathbf{L}(G)$ ✍

1.2.5 – Dato $G = (\{\mathsf{a}, \mathsf{b}\}, \{S\}, S, P)$ dove

- $P = \{S \longrightarrow \epsilon|SS|\mathsf{a}S\mathsf{b}|\mathsf{b}S\mathsf{a}\}$

indicare $\mathbf{L}(G)$ ✍

1.2.6 – Dato $G = (\{0\}, \{S, A, B\}, S, P)$ dove

- $P = \{S \longrightarrow A0,\ A \longrightarrow B,\ B \longrightarrow A0\}$

indicare $\mathbf{L}(G)$ ed il suo tipo ✍

1.2.7 – Dato $\Sigma = \{0, 1\}$, definire G tale che $L = \mathbf{L}(G)$ nei seguenti casi:

i. L delle stringhe con uno ed un solo 1

ii. L delle stringhe con almeno uno 0

iii. L delle stringhe con non più di tre 1

iv. L delle stringhe con almeno tre 0 ✍

1.2.8 – Dato $\Sigma = \{0, 1\}$, definire G per i seguenti linguaggi:

i. $L_1 = \{0^n 1^m \mid m > n \geq 0\}$

ii. $L_2 = \{0^n 1^{2n} \mid n \geq 0\}$

iii. $L_3 = \{0^n 1^{n+2} \mid n \geq 1\}$

iv. $L_4 = \{0^n 1^{n-3} \mid n \geq 3\}$

v. $L_1 \cup L_2$

vi. $L_1 \cdot L_2$

vii. L_1^3

viii. L_1^*

ix. $L_1 - \bar{L}_4$

1.2.9 – Dato $L = \{w \in \{0, 1\} \mid |w| \bmod 3 = 1\}$, definire G tale che $L = \mathbf{L}(G)$

1.2.10 – Dato $L = \{0^n 1^k \mid k = n + 1, n \geq 0\}$, definire G tale che $L = \mathbf{L}(G)$

1.3 Automi

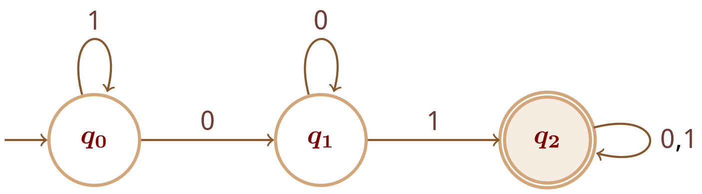
1.3.1 – Dato l'automa M definito da:

	a	b
$\rightarrow q_0$	q_0	q_1
$*q_1$	q_0	q_2
q_2	q_2	q_1

Quali tra a^{3}b, abaab, a^{4}bba, a^{6}b^{8}ab sono accettate?

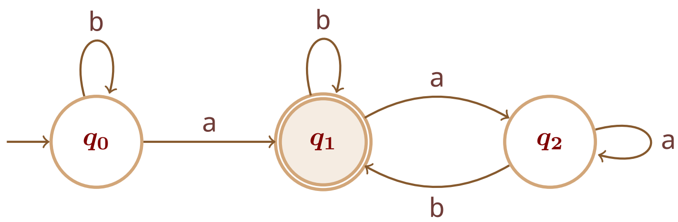
1.3.2 – Dato l'automa definito nell'Esercizio 1.3.1, fornire il diagramma di transizione

1.3.3 – Dato l'automa definito dal diagramma di transizione



fornire la sua tabella di transizione

1.3.4 – Dato l'automa definito dal diagramma di transizione



indicare il linguaggio accettato

2 Linguaggi Regolari

2.1 Grammatiche Lineari Destre

2.1.1 – Definire grammatiche lineari destre che generino ognuno linguaggi seguenti:

i. L delle stringhe binarie con uno ed un solo 1

ii. L delle stringhe binarie con almeno uno 0

iii. L delle stringhe binarie con non più di tre 1

iv. L delle stringhe binarie con almeno tre 0 ✍

2.1.2 – Definire una grammatica lineare destra che generi $\{1^n0 \mid n > 0\} \cup \{0^k1 \mid k > 0\}$ ✍

2.2 Automi

2.2.1 DFA

2.2.1 – Trovare un DFA M tale che accetti l'insieme delle stringhe binarie che abbiano 11 come prefisso ✍

2.2.2 – Trovare un DFA tale che accetti $\{(10)^k \mid n > 0\}$ ✍

2.2.3 – Dato $L = \{w \in \{a, b\}^* \mid |w| \bmod 3 = 1\}$, definire un DFA M tale che $L = \mathbf{L}(M)$ ✍

2.2.4 – Definire un DFA M che accetta tutte le stringhe binarie tranne quelle che contengono 001 come sottostringa ✍

2.2.5 – Dato $L = \{awa \mid w \in \{a, b\}^*\}$, definire il DFA M tale che $L = \mathbf{L}(M)$ ✍

2.2.6 – Dato L dell'esercizio 2.2.5, definire il DFA M tale che $L^2 = \mathbf{L}(M)$ ✍

2.2.7 – Trovare un DFA per i seguenti linguaggi sull'alfabeto binario:

i. $\{w \in \{\mathsf{a},\mathsf{b}\}^* \mid |w| \bmod 3 = 0\}$

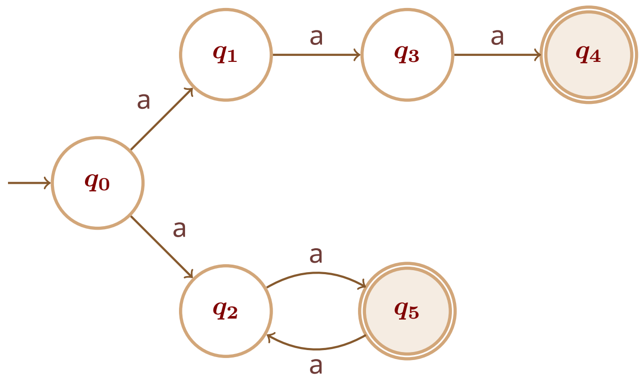
ii. $\{w \in \{\mathsf{a},\mathsf{b},\mathsf{c}\}^* \mid |w| \bmod 5 \neq 0\}$

iii. $\{w \in \{0,1\}^* \mid n_1(w) \bmod 3 = 0\}$

iv. $\{w \in \{0,1\}^* \mid [n_0(w) \bmod 3] > [n_1(w) \bmod 3]\ \}$ ✍

2.2.2 NFA e trasformazioni

2.2.8 – Trovare un DFA equivalente al NFA

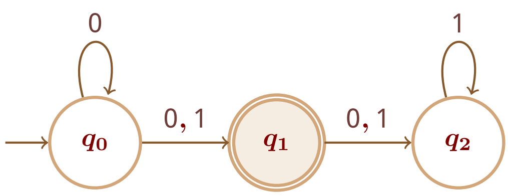


✍

2.2.9 – Definire un NFA con massimo 5 stati per

$$\{\mathsf{baba}^n \mid n \geq 0\} \cup \{\mathsf{bab}^m \mid m \geq 0\}$$ –

2.2.10 – Trovare un DFA equivalente a:



✍

2.3 Espressioni Regolari

2.3.1 – Data $R = (00)^*(11)^*1$, definire $\mathbf{L}(R)$ come insieme ✍

2.3.2 – Definire un'espressione regolare per ognuno linguaggi seguenti:

i. L delle stringhe binarie con uno ed un solo 1

ii. L delle stringhe binarie con almeno uno 0

iii. L delle stringhe binarie con non più di tre 1

iv. L delle stringhe binarie con almeno tre 0

2.3.3 – Definire un'espressione regolare R per il linguaggio $\{w \in \{0,1\}^* \mid w \text{ non contiene zeri consecutivi }\}$

2.3.4 – Definire un'espressione regolare R per il linguaggio $\{a^i b^j \mid i + j \text{ pari }\}$

2.3.5 – Definire un'espressione regolare R per il linguaggio $\{w \in \{0,1\}^* \mid n_0(w) + n_1(w) \text{ dispari }\}$

2.4 Applicazioni del Teorema di Kleene

2.4.1 – Definire un DFA per ognuna delle espressioni regolari ottenute nell'Es. 2.3.2.

2.4.2 – Definire espressioni regolari per i linguaggi di cui all'Es. 2.2.7 trasformando ognuno dei DFA.

2.4.3 – Data $R = (00)^*(11)^*1$, definire un DFA M ed una grammatica G tali che $\mathbf{L}(R) = \mathbf{L}(M) = \mathbf{L}(G)$

2.4.4 – Dato l'automa M dell'Es. 2.2.8, trovare una R tale che $\mathbf{L}(R) = \mathbf{L}(M)$.

2.4.5 – Dato l'automa M dell'Es. 2.2.9, trovare una R tale che $\mathbf{L}(R) = \mathbf{L}(M)$.

2.4.6 – Dato l'automa M dell'Es. 2.2.10, trovare una R tale che $\mathbf{L}(R) = \mathbf{L}(M)$.

2.4.7 – Definire un NFA che accetti il linguaggio denotato da: $((10+1)(11)^*) + (01)^*$

2.5 Proprietà di chiusura

2.5.1 – Definire una grammatica lineare destra che generi: $\{w \in \{a,b\}^* \mid w = a^k n \vee b^k a,\ k > 0\}$

2.5.2 – Definire un DFA per il complemento dei linguaggi di cui all'Es. 2.2.7.

2.5.3 – Definire una grammatica regolare per ognuno dei DFA ottenuti nell'Es. 2.5.2.

2.5.4 – Sfruttando la chiusura rispetto all'unione, definire una grammatica lineare destra ed un DFA per il linguaggio $\{\ldots \mid \ldots\}$ *da completare*

2.5.5 – Trovare gli NFA che accettino

i. $\mathbf{L}((0+1)1^*) \cap \mathbf{L}(011^*)$

ii. $\mathbf{L}(ab^*a^*) \cap \mathbf{L}(a^*b^*a)$

2.6 Pumping Lemma – Linguaggi regolari

Esercizi da risolvere attraverso il Pumping Lemma su $\mathcal{L}_{\mathbf{reg}}$.

2.6.1 – Provare che non sono regolari i linguaggi:

i. $\{0^i 1^j \mid i, j > 0,\ i < j\}$

ii. $\{a^i b^j \mid i, j > 0,\ i > j\}$

iii. $\{0^i 1^j \mid i, j > 0,\ 2i > j\}$

iv. $\{a^i b^j \mid i, j > 0,\ i \neq j\}$

v. $\{w \in \{0,1\}^* \mid i, j > 0,\ 2i > j\}$

vi. $\{w \in \{0,1\}^* \mid n_0(w) \neq n_1(w)\}$

vii. $\{w \in \{a,b\}^* \mid n_a(w) \neq n_b(w)\}$

2.6.2 – Provare che non sono regolari i linguaggi:

i. $\{a^i b a^i \mid i > 0\}$

ii. $\{ww^R \mid w \in \{a,b\}^*\}$

iii. $\{ww \mid w \in \{0,1\}^*\}$

2.6.3 – Provare che non sono regolari i linguaggi:

i. $\{a^k b^k c^k \mid k \geq 0\}$

ii. $\{a^h b^k c^k d^h \mid h, k \geq 0\}$

2.6.4 – Provare che non sono regolari i linguaggi:

i. $\{0^k \mid k = 2^i,\ i \geq 0\}$

ii. $\{0^h \mid h = i^2,\ i \geq 0\}$

iii. $\{w \in \{a,b\}^* \mid |w| = i^2,\ i \geq 0\}$

2.6.5 – Stabilire se possa essere regolare $\{w_1 c w_2 \mid w_1, w_2 \in \{a,b\}^*,\ w_1 \neq w_2\}$

3 Linguaggi Liberi

3.1 Grammatiche

3.1.1 – Determinare una grammatica G per i seguenti linguaggi:

1. $\{ww^R \mid w \in \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}^*\}$
2. $\{w \in \{\mathsf{a}, \mathsf{b}\}^* \mid n_{\mathsf{a}}(w) = n_{\mathsf{b}}(w)\}$

3.2 Pumping Lemma

3.2.1 – Dimostrare che i seguenti linguaggi non sono liberi da contesto:

i. $\{\mathsf{a}^n\mathsf{b}^n\mathsf{c}^n \mid n > 0\}$

ii. $\{\mathsf{a}^n\mathsf{b}^m\mathsf{c}^p \mid 1 \leq n \leq m \leq p\}$

iii. $\{\mathsf{a}^i\mathsf{b}^j\mathsf{c}^i\mathsf{d}^j \mid i, j \geq 1\}$

iv. $\{ww \mid w \in \{0, 1\}^+\}$

3.2.2 – Dimostrare che i seguenti linguaggi non sono liberi da contesto:

i. $\{\mathsf{a}^t \mid t \text{ numero primo}\}$

ii. $\{\mathsf{a}^{n^2} \mid n \geq 0\}$

iii. $\{\mathsf{a}^i\mathsf{b}^j \mid i = 2^j,\ i, j \geq 0\}$

iv. $\{\mathsf{a}^n\mathsf{b}^m \mid n > 2^m,\ n, m \geq 0\}$

v. $\{\mathsf{a}^k\mathsf{b}^r \mid k > 0,\ r > k^2\}$

vi. $\{\mathsf{a}^{2^n} \mid n \geq 1\}$

3.3 Proprietà di chiusura

3.3.1 – Determinare una grammatica G per i seguenti linguaggi:

1. $\{0^i1^j \mid i \neq j,\ i, j > 0\}$
2. $\{w \in \{a, b\}^* \mid n_a(w) \neq n_b(w)\}$ ✍

3.3.2 – Dimostrare che i seguenti linguaggi sono liberi da contesto:

i. $\{0^n1^n \mid n = 2k + 1,\ k > 0\}$

ii. $\{a^nb^n \mid n > 0,\ n \neq 7\}$

iii. $\{w \in \{a, b\}^* \mid n_a(w) = n_b(w) \text{ e } w \text{ non contiene bbb}\}$

iv. $\{c^na^ib^kc^m \mid n, m, i, k > 0,\ i \neq k\}$ ✍

3.3.3 – Dimostrare che sono liberi da contesto i linguaggi complemento $\bar{L}$ dei seguenti linguaggi:

i. $\{ww \mid w \in \{0, 1\}^*\}$

ii. $\{a^ib^ic^i \mid i \geq 0\}$ ✍

3.3.4 – Dimostrare che i seguenti linguaggi non sono liberi da contesto:

i. $\{a^nb^nc^n \mid n > 0\}$

ii. $\{w \in \{a, b, c\}^* \mid n_a(w) = n_b(w) = n_c(w)\}$ ✍

3.4 Forme Normali

3.4.1 – Date le grammatiche di cui all'Esercizio 3.1.1, trasformarle in GNF. ✍

3.5 Automi a Pila

3.5.1 – Date le grammatiche di cui all'Esercizio 3.4.1, ricavare i PDA che riconoscano gli stessi linguaggi. ✍

Bibliografia

[1] Giorgio Ausiello, Fabrizio D'Amore, e Giorgio Gambosi. *Linguaggi, Modelli, Complessità*. FrancoAngeli, 2003.

[2] Daniel I.A. Cohen. *Introduction to Computer Theory*. Wiley, 1996.

[3] John E. Hopcroft, Rajeev Motwani, e Jeffrey D. Ullman. *Automi, Linguaggi e Calcolabilità*. Pearson Italia, 3a edizione, 2009.

[4] Peter Linz. *An Introduction to Formal Languages and Automata*. Jones & Bartlett, 5a edizione, 2012.

[5] Robert N. Moll, Michael A. Arbib, e Assaf J. Kfoury. *An introduction to formal language theory*. Springer, 1988.

[6] Michael Sipser. *Introduction to the theory of computation*. Thomson, 2a edizione, 2005.

[7] Thomas Sudkamp. *Languages and Machines*. Addison-Wesley, 3a edizione, 2006.