Corso di Laurea in Sicurezza dei sistemi e delle reti informatiche

Fondamenti di informatica per la sicurezza

anno accademico 2004–2005

docente: Stefano Ferrari

25.01.2005 — Soluzione della seconda parte — vers. A

valutazioni **1** (4) _____ **2** (4) _____ **3** (4) _____ **4** (6) _____ **5** (6) _____ **6** (8) _____

Cognome _____

Nome _____

Matricola _____ **Firma** _____

Esercizio 1

Siano dati i linguaggi L_1 e L_2:

- $L_1 = \{a, ba, ab\}$
- $L_2 = \{x, y\}$

Descrivere i linguaggi:

a) $L_3 = L_1 \cap L_2$

b) $L_4 = L_1 \cup L_2$

c) $L_5 = L_1 L_2$

d) $L_6 = {L_1}^2$

e) $L_7 = {L_1}^* {L_2}^*$

f) $L_8 = ({L_1}^2 L_2)^*$

Per quegli insiemi di cui sia troppo lungo (o impossibile) dare una descrizione estensionale, elencare almeno tre elementi, indicando le caratteristiche degli elementi che li compongono. In particolare, chiarire se la stringa vuota ϵ appartiene al linguaggio.

Soluzione

a) $L_3 = L_1 \cap L_2 = \emptyset$
Gli insiemi L_1 e L_2 non hanno elementi in comune, quindi la loro intersezione è vuota.

Nota: L'insieme vuoto $\emptyset$ è diverso dall'insieme costituito dalla sola stringa vuota, $\{\epsilon\}$.

b) $L_4 = L_1 \cup L_2 = \{a, ba, ab\,x, y\}$

c) $L_5 = L_1 L_2 = \{ax, bax, abx\,ay, bay, aby\}$

d) $L_6 = {L_1}^2 = \{aa, baa, aba\,baba, abba\,aab, baab, abab\}$

e) $L_7 = {L_1}^* {L_2}^*$
L'insieme L_7 è dato dalle stringhe formate come concatenazione di un numero arbitrario (eventualmente nullo) di elementi di L_1 seguito da una concatenazione di un numero arbitrario (eventualmente nullo) di elementi di L_2. Poiché sia ${L_1}^*$ che ${L_2}^*$ sono composto da infiniti elementi, anche L_7 avrà infiniti elementi. L'insieme $\{\epsilon, abaaba, xxyxyx, aabyxxx\}$ è un sottoinsieme di L_7.

f) $L_8 = ({L_1}^2 L_2)^*$
L'insieme L_8 è formato dalla concatenazione di un numero arbitrario (eventualmente nullo) di stringhe composte da due elementi di L_2 e da un elemento di L_1. Pertanto, L_8 è composto da infiniti elementi. L'insieme $\{\epsilon, aax, baabyaabx\}$ è un sottoinsieme di L_8.

Esercizio 2

Sia data la seguente grammatica, $G = \langle T, V, P, S\rangle$, definita su $\Sigma = \{a, b, c, d\}$:

- insieme dei simboli terminali, T: $T = \Sigma$
- insieme dei metasimboli, V: $V = \{K, H\}$
- insieme delle regole di produzione, P: $P = \{S ::= K, K ::= d|Hb|Hc, H ::= c|Kd|Ha\}$

Quali fra le seguenti stringhe vengono generate da G?

a) *cacdb*

b) *dab*

c) *ddac*

d) *cddaac*

e) *dbdac*

Riportare la successione di regole da applicare per la generazione di tali stringhe e le stringhe parziali ottenute, spiegando perché non si possono ottenere le stringhe che eventualmente non risultassero appartenere al linguaggio generato da G.

Soluzione

a) *cacdb*

	S
$S ::= K$	K
$K ::= Hb$	Hb
$H ::= Kd$	Kdb
$K ::= Hc$	$Hcdb$
$H ::= Ha$	$Hacdb$
$H ::= c$	$cacdb$

La stringa *cacdb* è generata da G: $cacdb \in L(G)$.

b) *dab*

	S
$S ::= K$	K
$K ::= Hb$	Hb
$H ::= Ha$	Hab
$H ::= Kd$	$Kdab$

Non è possibile eliminare il metasimbolo K senza aggiungere un altro simbolo.

La stringa *dab* non è generata da G: $dab \notin L(G)$.

c) *ddac*

	S
$S ::= K$	K
$K ::= Hc$	Hc
$H ::= Ha$	Hac
$H ::= Kd$	$Kdac$
$K ::= d$	$ddac$

La stringa *ddac* è generata da G: $ddac \in L(G)$.

d) *cddaac*

	S
$S ::= K$	K
$K ::= Hc$	Hc
$H ::= Ha$	Hac
$H ::= Ha$	$Haac$
$H ::= Kd$	$Kdaac$
$K ::= d$	$ddaac$

Non esistono altri metasimboli da espandere e manca un simbolo per ottenere la stringa data.

La stringa *cddaac* non è generata da G: $cddaac \notin L(G)$.

e) *dbdac*

	S
$S ::= K$	K
$K ::= Hc$	Hc
$H ::= Ha$	Hac
$H ::= Kd$	$Kdac$
$K ::= Hb$	$Hbdac$
$H ::= Kd$	$Kdbdac$

Non è possibile eliminare il metasimbolo K senza aggiungere un altro simbolo.

La stringa *dbdac* non è generata da G: $dbdac \notin L(G)$.

Esercizio 3

Sia dato il seguente automa a stati finiti, A, $A = \langle Q, \Sigma, \delta, q_0, F \rangle$:

- insieme degli stati, Q: $Q = \{q_0, q_1, q_2, q_3\}$
- alfabeto di input, Σ: $\Sigma = \{a, b, c, d, e\}$
- funzione di transizione δ:

	a	b	c	d	e
q_0	q_2	q_1	q_2	q_3	q_1
q_1	q_2	q_0	q_1	q_0	q_3
q_2	q_1	q_2	q_1	q_1	q_1
q_3	q_3	q_2	q_0	q_0	q_2

- stato iniziale, q_0
- insieme di stati finali, F: $F = \{q_1\}$

Indicare:

a) quattro stringhe accettate da A

b) quattro stringhe rifiutate da A

Soluzione

a) quattro stringhe accettate da A:

- *abcae*
- *ebadc*
- *bbae*
- *ddb*

b) quattro stringhe rifiutate da A:

- *dacba*
- *abcd*
- *dab*
- *eabce*

Esercizio 4

Modellare, tramite un automa a stati finiti deterministico, il funzionamento di una automobile.

L'automobile dispone di una portiera e di un blocco di avviamento a chiave. Quando la chiave viene inserita, l'automobile si mette in moto. L'automobile non deve essere avviata se la portiera non è chiusa.

Ipotizzare che non si possano verificare contemporaneamente più azioni. Modellare l'automa in modo che esso accetti solo le stringhe che descrivono il funzionamento sicuro dell'automobile. In particolare, individuare possibili situazioni fisicamente irrealizzabili o pericolose e formalizzarle in modo che l'automa rifiuti le successioni di azioni che porterebbero l'automobile in tali situazioni.

Stati e simboli riportati nel testo sono solo indicativi: possono essere modificati, ridotti ed estesi a secondo delle esigenze del progetto.

Soluzione

L'automa deve modellare un sistema fisico. L'insieme dei simboli di input modella quindi gli stimoli che il sistema riceve dall'esterno (o le azioni che esso subisce) e gli stati descrivono le situazioni in cui il sistema viene a trovarsi.

Questo permette di vedere l'automa come un simulatore del sistema in esame: l'automa deve accettare le stringhe che rappresentano le sequenze di stimoli (o azioni) fisicamente realizzabili oppure quelle che rappresentano una sequenza di eventi di particolare interesse.

Il sistema è descrivibile come insieme di due sottosistemi indipendenti: la portiera e il blocco di avviamento. La portiera può trovarsi in due stati (aperta o chiusa), così come il blocco di avviamento (chiave inserita, chiave non inserita). Gli stati dell'automobile si possono derivare dalle combinazioni degli stati dei due sottosistemi. Pertanto, l'insieme degli stati, Q, può essere:

$$Q = \{ma,\ mc,\ fa,\ fc\}$$

dove la lettera m indica se il motore è in moto, in contrapposizione a f che indica che il motore è fermo, mentre le lettere a e c indicano che la portiera è aperta e chiusa, rispettivamente.

Le azioni che possono essere effettuate sul sistema sono l'apertura e la chiusura della portiera e l'inserimento o il disinserimento della chiave. Pertanto, insieme dei simboli, Σ, può essere:

$$\Sigma = \{a,\ c,\ i,\ d\}$$

dove a e c indicano, rispettivamente, l'azione di apertura e di chiusura della portiera, mentre i e d indicano, rispettivamente, l'inserimento e l'estrazione della chiave dal blocco di avviamento.

Le specifiche descrivono i seguenti comportamenti:

- l'inserimento della chiave provoca la messa in moto del motore;
- l'avviamento del motore a portiera aperta è una situazione non accettabile.

Poiché esiste almeno una situazione ritenuta non accettabile, è opportuno aggiungere un altro stato, *errore*, tale per cui una volta raggiunto non lo si possa più lasciare. Questa caratteristica formalizza il fatto che la situazione di errore è irreversibile, cioè non esiste una sequenza di azioni che permetta di riassorbire una situazione non accettabile.

Si può inoltre ipotizzare che il tentativo di apertura (chiusura) della portiera già aperta (chiusa) generi errore. Analoghe considerazioni valgono per l'inserimento/disinserimento della chiave. In tal caso, l'automa viene portato nello stato *errore*.

Ogni sequenza di azioni che non comporti il raggiungimento dello stato *errore* rappresenta il normale funzionamento dell'automobile. Pertanto, qualsiasi sequenza di simboli che non porti nello stato *errore* deve venire accettata, e, quindi, tutti gli stati tranne *errore* compongono l'insieme degli stati finali, F.

Si può ipotizzare che lo stato iniziale sia quello relativo alla macchina ferma con portiera chiusa, fc.

Le ipotesi aggiuntive rendono di fatto irraggiungibile lo stato ma, il quale, pertanto, può essere eliminato. Quindi, l'insieme degli stati, Q, qui utilizzato sarà:

$$Q = \{mc, fa, fc, errore\}$$

Con le ipotesi fatte, dovrebbero essere accettate, per esempio, le seguenti sequenze di azioni: $acidac$, $idac$. Al contrario, verrebbero rifiutate, tra le altre, le seguenti sequenze di azioni: c, $acaa$, $aicd$. Va notato che aggiungendo un qualsiasi suffisso ad una stringa rifiutata, si ottiene sempre una stringa rifiutata: se una certa sequenza di azioni porta in uno stato non accettabile, qualsiasi sequenza di azioni ad essa successiva non può renderla accettabile.

La tabella delle transizioni, $\delta : Q \times \Sigma \to Q$ può essere quella riportata in Tabella 1.

Esercizio 5

Sia data l'espressione regolare E, definita su $\Sigma = \{a, b, c\}$:

- $E = c^2(cb + a)^* + ab^*c^2$

Quali fra le seguenti stringhe vengono descritte da E?

a) $ccaaacb$

b) $ccbcabcc$

c) $ccacbaa$

d) $abbbcc$

e) acc

f) $aabcbcc$

Soluzione

Le espressioni regolari denotano degli insiemi di stringhe. In tal senso, possiamo applicare l'operatore di relazione insiemistica $\subseteq$ alle espressioni regolari per indicare che l'insieme denotato da un'espressione contiene l'insieme denotato da una seconda espressione regolare. Per esempio, $E_1 \subseteq E_2$ significa che tutte le stringhe descritte da E_1 sono descritte anche da E_2.

Ricordando che l'espressione regolare s descrive l'insieme di stringhe composto dalla sola s, $\{s\}$, si può dimostrare che tale stringa viene descritta da un'espressione regolare E derivando una catena di inclusioni del tipo $s \subseteq E_1 \subseteq E_2 \subseteq \cdots \subseteq E_k \equiv E$.

Osserviamo innanzitutto che le stringhe che vengono descritte da E sono descritte, in alternativa, o dalla sottoespressione $E_1 = c^2(cb + a)^*$ o dalla sottoespressione $E_2 = ab^*c^2$. Questa premessa semplificherà la spiegazione delle soluzioni di seguito riportate.

a) $ccaaacb$
$ccaaacb \subseteq (cc)(a)(a)(a)(cb) \subseteq c^2(a + cb)^4 \subseteq c^2(cb + a)^* \subseteq c^2(cb + a)^* + ab^*c^2$

La stringa $ccaaacb$ viene descritta da E: $ccaaacb \in \mathcal{L}(E)$.

b) $ccbcabcc$
Le stringhe ottenute da E_1 avranno cc nel prefisso, ma non nel suffisso, mentre le stringhe ottenute da E_2 avranno cc nel suffisso, ma non nel prefisso.

La stringa $ccbcabcc$ ha cc sia come prefisso che come suffisso. Pertanto, non può essere descritta da E: $ccbcabcc \notin \mathcal{L}(E)$.

c) $ccacbaa$
$ccacbaa \subseteq (cc)(a)(cb)(a)(a) \subseteq c^2(cb + a)^4 \subseteq c^2(cb + a)^* \subseteq c^2(cb + a)^* + ab^*c^2$

La stringa $ccacbaa$ viene descritta da E: $ccacbaa \in \mathcal{L}(E)$.

d) $abbbcc$
$abbbcc \subseteq (a)(bbb)(cc) \subseteq ab^3c^2 \subseteq ab^*c^2 \subseteq c^2(cb + a)^* + ab^*c^2$

La stringa $abbbcc$ viene descritta da E: $abbbcc \in \mathcal{L}(E)$.

e) acc
$acc \subseteq (a)(cc) \subseteq ac^2 \subseteq ab^*c^2 \subseteq c^2(cb + a)^* + ab^*c^2$

La stringa acc viene descritta da E: $acc \in \mathcal{L}(E)$.

f) $aabcbcc$
La stringa $aabcbcc$ non può essere descritta da E_1 perché non ha cc come prefisso, e non può essere descritta da E_2 perché ha più di un simbolo a come prefisso.

La stringa $aabcbcc$ non viene descritta da E: $aabcbcc \notin \mathcal{L}(E)$.

$\boldsymbol{\delta}$	a	c	i	d
mc	*errore*	*errore*	*errore*	fc
fa	*errore*	fc	*errore*	*errore*
fc	fa	*errore*	mc	*errore*
errore	*errore*	*errore*	*errore*	*errore*

Tabella 1: Tabella delle transizioni dell'automa dell'esercizio 4.

Esercizio 6

Indicare una espressione regolare (non banale) definita su $\Sigma = \{a, b, c\}$ che descriva le seguenti stringhe:

- *abbabba*
- *abba*
- *cc*
- *caacc*

ma non le seguenti:

- *babbab*
- *cbaabcc*
- *abaccba*
- *ccaacb*

Soluzione

Si può notare che tutte le stringhe da includere o non contengono il simbolo b o non contengono il simbolo c. Inoltre, le stringhe che non contengono c iniziano per a. Queste caratteristiche possono essere descritte dall'espressione regolare $a(a+b)^* + (a+c)^*$:

- stringhe che non contengono b: $a(a+b)^* + \underline{(a+c)^*}$;
- stringhe che non contengono c: $\underline{a(a+b)^*} + (a+c)^*$;
- a come prefisso: $\underline{a}(a+b)^* + (a+c)^*$.

Nessuna delle stringhe del secondo gruppo viene descritta da tale espressione regolare:

- *babbab*: non contiene c, ma inizia per b;
- *cbaabcc*: contiene sia b che c;
- *abaccba*: contiene sia b che c;
- *ccaacb*: contiene sia b che c.

Altre espressioni regolari che rispettano le specifiche del problema sono:

- $a(b^2a)^* + c(a^2c)^*c$
- $c^*(a+b^2)^*c^*$
- $(ab^2)^*(c+a)^*$
- $(c+(abb)^*)a^*c^*$
- $(a+c)(b^*a+c)(b^2a+ac^2)^*$
- $c^*(ab^2)^*a^*c^*$
- $a(a+b^2)^* + (a+c)^*$