Corso di Laurea in Sicurezza dei sistemi e delle reti informatiche

Fondamenti di informatica per la sicurezza

anno accademico 2004–2005

docente: Stefano Ferrari

23.06.2005 — 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:

- $L_1 = \{a, b, ba\}$
- $L_2 = \{cd, z\}$

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) $L_4 = L_1 \cup L_2 = \{a, b, ba, cd, z\}$

c) $L_5 = L_1 L_2 = \{acd, bcd, bacd, az, bz, baz\}$

d) $L_6 = {L_1}^2 = \{aa, ab, aba, ba, bb, bba, baa, bab, baba\}$

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 $\{\epsilon, aabba, cdzcd, baazcdcd\}$ 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_1 e da un elemento di L_2. Pertanto, L_8 è composto da infiniti elementi. L'insieme $\{\epsilon, aabz, babcdbaaz\}$ 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: $P = \{S ::= K,\ K ::= a|bH|cH,\ H ::= b|dK|cH\}$

Quali fra le seguenti stringhe vengono generate da G?

a) $bcda$

b) *bdda*

c) *cdbb*

d) *cccc*

e) *bdca*

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)

bcda	
	S
$S ::= K$	K
$K ::= bH$	bH
$H ::= cH$	bcH
$H ::= dK$	$bcdK$
$K ::= a$	$bcda$

La stringa *bcda* è generata da G: $bcda \in \mathcal{L}(G)$.

b)

bdda	
	S
$S ::= K$	K
$K ::= bH$	bH
$H ::= dK$	bdK

Non esiste regola che generi il simbolo d dal metasimbolo K.

La stringa *bdda* non è generata da G: $bdda \notin \mathcal{L}(G)$.

c)

cdbb	
	S
$S ::= K$	K
$K ::= cH$	cH
$H ::= dK$	cdK
$K ::= bH$	$cdbH$
$H ::= b$	$cdbb$

La stringa *cdbb* è generata da G: $cdbb \in \mathcal{L}(G)$.

d)

cccc	
	S
$S ::= K$	K
$K ::= cH$	cH
$H ::= cH$	ccH
$H ::= cH$	$cccH$
$H ::= cH$	$ccccH$

La stringa *cccc* è stata ottenuta, ma non è possibile eliminare il metasimbolo H.

La stringa *cccc* non è generata da G: $cccc \notin \mathcal{L}(G)$.

e)

bdca	
	S
$S ::= K$	K
$K ::= bH$	bH
$H ::= dK$	bdK
$K ::= cH$	$bdcH$

Non esiste regola che generi il simbolo a dal metasimbolo H.

La stringa *bdca* non è generata da G: $bdca \notin \mathcal{L}(G)$.

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_1	q_3	q_2	q_1	q_1
q_2	q_3	q_2	q_0	q_0	q_2
q_3	q_3	q_0	q_1	q_0	q_3

- 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:

- *acb*
- *aedb*
- *ba*
- *caadb*

b) quattro stringhe rifiutate da A:

- *deb*
- *aaa*
- *db*
- *abcd*

Esercizio 4

Modellare il funzionamento di un termostato mediante un automa a stati finiti.

Il termostato controlla un sistema di condizionamento dotato sia di apparato di riscaldamento che di raffreddamento.

L'utente può regolare due soglie (alta, t_a, e bassa, t_b) e la modalità (*estate* e *inverno*). Il termostato può tenere attivo solo uno dei due apparati alla volta. Se in modalità *estate*, il termostato attiva l'apparato raffreddante quando viene superata la soglia alta, t_a, per spegnerlo solo quando è stata superata la soglia bassa, t_b. Al contrario, in modalità *inverno*, il termostato attiva l'apparato riscaldante quando viene superata la soglia bassa, t_b, per spegnerlo solo quando è stata superata la soglia alta, t_a.

Ipotizzare che non si possano verificare contemporaneamente più eventi. Modellare l'automa in modo che esso accetti solo le stringhe che descrivono il funzionamento dell'termostato. In particolare, individuare possibili situazioni fisicamente irrealizzabili o pericolose e formalizzarle in modo che l'automa rifiuti le successioni di azioni che porterebbero il termostato 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 il funzionamento di un dispositivo. L'insieme dei simboli di input modella quindi i segnali che il dispositivo riceve (generati da qualche evento) e gli stati descrivono le condizioni. le situazioni in cui tale dispositivo viene a trovarsi.

Questo permette di vedere l'automa come un simulatore del dispositivo considerato: l'automa deve accettare le stringhe che rappresentano le sequenze di segnali fisicamente realizzabili oppure quelle che rappresentano una sequenza di eventi di particolare interesse.

L'insieme degli stati del termostato è descrivibile come l'insieme delle situazioni in cui si trova ad operare. Tali situazioni sono date dalla combinazione di diversi fattori:

- è stata selezionata la modalità estiva o invernale;
- lo stato di funzionamento degli apparati controllati (riscaldamento, raffreddamento).

Poichè i fattori sono tra loro indipendenti, gli stati risulteranno dalla combinazione dei singoli fattori. Per formalizzare gli stati, useremo una composizione di simboli che indichino i diversi fattori in gioco. I simboli E e I per indicare, rispettivamente la modalità estiva e invernale, i simboli C e F per indicare gli apparati di riscaldamento e raffreddamento, e, infine, i simboli a e d ma apporre come pedice ai simboli degli apparati per indicare se tale apparato è attivo e disattivo, rispettivamente. Pertanto, lo stato EF_aC_d indicherà:

- modalità estiva ($\underline{E}F_aC_d$)
- raffreddamento attivato ($E\underline{F_a}C_d$)
- riscaldamento disattivato ($EF_a\underline{C_d}$)

Pertanto, l'insieme degli stati, Q, può essere:

$$\begin{aligned} Q \;=\; & \{EF_aC_a,\ IF_aC_a,\ EF_dC_a,\ IF_dC_a,\\ & EF_aC_d,\ IF_aC_d,\ EF_dC_d,\ IF_dC_d\} \end{aligned}$$

I simboli che il dispositivo deve gestire (gli eventi ai quali deve rispondere) riguardano la modalità di funzionamento e la temperatura ambientale. Indicheremo con:

- e, la richiesta di passare in modalità estiva;
- i, la richiesta di passare in modalità invernale;
- t_a, la segnalazione del raggiungimento della temperatura soglia alta;
- t_b, la segnalazione del raggiungimento della temperatura soglia bassa.

Per una descrizione maggiormente dettagliata, sarebbe necessario distinguere tra il superamento di temperatura soglia per eccesso e per difetto. In altri termini, potrebbe essere necessario sapere se un temperatura soglia è stata raggiunta mentre la temperatura ambientale sta aumentando o mentre sta diminuendo. Pertanto, i simboli potrebbero essere:

- t_{ae}, temperatura più elevata della temperatura soglia alta;
- t_{ad}, temperatura inferiore alla temperatura soglia alta;
- t_{be}, temperatura più elevata della temperatura soglia bassa;

- t_{bd}, temperatura inferiore alla temperatura soglia bassa.

Quindi, l'insieme dei simboli, Σ, sarà:

$$\Sigma = \{e, i, t_{ae}, t_{ad}, t_{be}, t_{bd}\}.$$

Dalle specifiche, il comportamento del sistema prevede che:

- se la modalità è estiva e la temperatura è più elevata di t_a, deve entrare in funzione l'apparto raffreddante;
- se la modalità è estiva e la temperatura è inferiore a t_b, deve essere disattivato l'apparto raffreddante;
- se la modalità è invernale e la temperatura è inferiore di t_b, deve entrare in funzione l'apparto riscaldante;
- se la modalità è invernale e la temperatura è superiore di t_a, deve essere disattivato l'apparto riscaldante.

Inoltre è logico ipotizzare che:

- se la modalità è estiva, l'apparato riscaldante sia disattivato;
- se la modalità è invernale, l'apparato raffreddante sia disattivato.

Ogni azione che violi le precedenti condizioni dovrebbe generare una situazione di errore, che può essere rappresentata da uno stato particolare, *err*. Tale stato, una volta raggiunto, non può più essere lasciato. In tal modo, lo stato *err* può essere utilizzato per discriminare le sequenze di azioni (simboli) accettabili. Ogni sequenza di azioni che non comporti il raggiungimento dello stato *err* rappresenta il normale comportamento dell'utente. Pertanto, qualsiasi sequenza di simboli che non porti nello stato *err* deve venire accettata, e, quindi, tutti gli stati tranne *err* compongono l'insieme degli stati finali, F.

Con le ipotesi fatte, dovrebbero essere accettate, per esempio, le seguenti sequenze di azioni:

- $e - t_{ae} - i - t_{ad} - t_{bd}$,
- $i - e - e - e - i$.

Si può ipotizzare che lo stato iniziale sia quello relativo al dispositivo impostato sulla modalità invernale, con entrambi i sistemi disattivati, IF_dC_d.

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

Diverse varianti sono possibili. Per esempio, si può ampliare l'automa per considerare anche le azioni e gli stati necessari per descrivere la possibilità di attivare o disattivare l'intero dispositivo. Oppure si può operare una semplificazione dell'automa, unificando gli eventi riguardanti le soglie (t_a sostituisce t_{ae} e t_{ad}, e, analogamente, t_b sostituisce t_{ba} e t_{bd}) e, dando per inteso la mutua esclusività degli apparati di condizionamento, l'insieme degli stati può semplificarsi in $Q = \{C_a, C_d, F_a, F_d\}$. In questo caso, la tabella delle transizioni, $\delta : Q \times \Sigma \to Q$ può essere quella riportata in Tabella 2.

Esercizio 5

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

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

Quali fra le seguenti stringhe vengono descritte da E?

a) *ccbccb*

b) *ccaaabc*

c) *abbbcc*

d) *acc*

e) *ccabccb*

f) *aaabcc*

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 $E_1 = c^2(bc + a)^*$

$\boldsymbol{\delta}$	e	i	t_{ae}	t_{ad}	t_{be}	t_{bd}
EF_aC_a	err	err	err	err	err	err
IF_aC_a	err	err	err	err	err	err
EF_dC_a	err	err	err	err	err	err
IF_dC_a	EF_dC_d	IF_dC_a	IF_dC_d	IF_dC_d	IF_dC_a	IF_dC_a
EF_aC_d	EF_aC_d	IF_dC_d	EF_aC_d	EF_aC_d	EF_dC_d	EF_dC_d
IF_aC_d	err	err	err	err	err	err
EF_dC_d	EF_dC_d	IF_dC_d	EF_aC_d	EF_aC_d	EF_dC_d	EF_dC_d
IF_dC_d	EF_dC_d	IF_dC_d	IF_dC_d	IF_dC_d	IF_dC_a	IF_dC_a
err	err	err	err	err	err	err

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

$\boldsymbol{\delta}$	e	i	t_a	t_b
C_a	F_d	C_a	C_d	C_a
C_d	F_d	C_d	C_d	C_a
F_a	F_a	C_d	F_a	F_d
F_d	F_d	C_d	F_a	F_d

Tabella 2: Tabella delle transizioni alternativa dell'automa dell'esercizio 4.

o dalla sottoespressione $E_2 = ab^*c^2$. Questa premessa semplificherà la spiegazione delle soluzioni di seguito riportate.

a) *ccbccb*
Le stringhe generate da E iniziano con la doppia c (quando sono generate da E_1) oppure per a (quando sono generate da E_2). La stringa data può appartenere solo al primo caso, però dopo le due c iniziali, il simbolo c può apparire solo se preceduto da un simbolo b.

Poiché questa condizione non viene verificata, la stringa *ccbccb* non può essere descritta da E: $ccbccb \notin \mathcal{L}(E)$.

b) *ccaaabc*
$ccaaabc \subseteq (cc)(aaa)(bc) \subseteq c^2a^3(bc) \subseteq c^2(a+bc)^4 \subseteq c^2(a+bc)^* \subseteq c^2(bc+a)^* + ab^*c^2$

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

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

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

d) *acc*
$acc \subseteq (a)(cc) \subseteq ac^2 \subseteq ab^*c^2 \subseteq c^2(bc+a)^* + ab^*c^2$

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

e) *ccabccb*
La stringa *ccabccb* non può essere generata da E_2 perché non ha il simbolo a come prefisso. Le stringhe generate da E_1 hanno cc come suffisso, ma la rimanente sottostringa deve essere composta da simboli a o da stringhe bc e pertanto devono terminare solo per a o c.

La stringa *ccabccb* termina per b e quindi non può essere descritta da E: $ccabccb \notin \mathcal{L}(E)$.

f) *aaabcc*
Le stringhe generate da E iniziano con la doppia c (quando sono generate da E_1) oppure per a (quando sono generate da E_2). In quest'ultimo caso, però, il simbolo a compare solo una volta.

La stringa *aaabcc* ha più di un simbolo a, e quindi non può essere descritta da E: $aaabcc \notin \mathcal{L}(E)$.

Esercizio 6

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

- *ababbbc*
- *ccbbbbc*
- *ccc*
- *ababbbbc*

ma non le seguenti:

- $acbbbc$
- $acccbc$
- acc
- $cabbba$

Soluzione

Si può notare che tutte le stringhe da includere hanno ab o c^2 come prefisso. Questa caratteristica può essere descritta dall'espressione regolare $(ab + c^2)(a + b + c)^*$.

Nessuna delle stringhe del secondo gruppo viene descritta da tale espressione regolare, in quanto tutte le stringhe hanno i primi due simboli diversi.

Altre espressioni regolari che rispettano le specifiche del problema sono:

- $(ab + c)^2 b^* c$