Corso di Laurea in Sicurezza dei sistemi e delle reti informatiche

Fondamenti di informatica per la sicurezza

anno accademico 2004–2005

docente: Stefano Ferrari

22.01.2005 — 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:

- $L_1 = \{a, b, c\}$
- $L_2 = \{x, y\}$

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

c) $L_5 = L_1 L_2 = \{ax, bx, cx, ay, by, cy\}$

d) $L_6 = {L_1}^2 = \{aa, ab, ac, ba, bb, bc, ca, cb, cc\}$

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 $\{\epsilon, ab, xxyx, bccbcbay\}$ 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 $\{\epsilon, acxbby, cbyccxaax\}$ 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) $ccdb$

b) *daac*

c) *acdab*

d) *adab*

e) *ccadab*

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) *ccdb*

	S
$S ::= K$	K
$K ::= Hb$	Hb
$H ::= Kd$	Kdb
$K ::= Hc$	$Hcdb$
$H ::= c$	$ccdb$

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

b) *daac*

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

Non è possilbile eliminare il metasimbolo K senza aggiungere un altro simbolo. La stringa *daac* non è generata da G: $daac \notin \mathcal{L}(G)$.

c) *acdab*

	S
$S ::= K$	K
$K ::= Hb$	Hb
$H ::= Ha$	Hab
$H ::= Kd$	$Kdab$
$K ::= Hc$	$Hcdab$
$H ::= Ha$	$Hacdab$

Non è possilbile eliminare il metasimbolo H senza aggiungere un altro simbolo. La stringa *acdab* non è generata da G: $acdab \notin \mathcal{L}(G)$.

d) *acdab*

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

Non è possibile ottenere il simbolo a dal metasimbolo K. La stringa *adab* non è generata da G: $adab \notin \mathcal{L}(G)$.

e) *ccadab*

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

Non è possibile ottenere il simbolo a dal metasimbolo K. La stringa *ccadab* non è generata da G: $ccadab \notin \mathcal{L}(G)$.

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_3	q_0	q_3	q_0	q_3
q_2	q_1	q_2	q_1	q_2	q_1
q_3	q_3	q_2	q_1	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:

- eac
- cbde
- aabb
- acde

b) Quattro stringhe rifiutate da A:

- abcd

- ea
- ccd
- ebea

Esercizio 4

Modellare, tramite un automa a stati finiti deterministico, il funzionamento di un forno a microonde.

Il forno a microonde può emettere onde alle seguenti potenze: 400 W e 700 W.

Il forno dispone di una manopola a tre posizioni per selezionare la potenza (0, 400 e 700), di un tasto di attivazione (start), di uno di stop (stop) e di un sensore per rilevare la posizione dello sportello (aperto/chiuso). Quando il forno è in funzione, all'apertura dello sportello interrompe l'emissione di radiazioni, e, alla successiva richiusura, riprende l'emissione alla stessa potenza selezionata.

Ipotizzare che non si possano verificare contemporaneamente più azioni. Modellare l'automa in modo che esso accetti solo le stringhe che descrivono il funzionamento del forno. In particolare, individuare possibili situazioni fisicamente irrealizzabili e formalizzarle in modo che l'automa rifiuti le successioni di azioni che porterebbero il forno 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 in esame può essere visto come composto da tre sottosistemi: la manopola di regolazione della potenza, il tasto di attivazione/spegnimento e lo sportello del forno. Questi tre sottosistemi sono tra loro indipendenti: un'azione su uno di essi non ha effetto sugli altri. Pertanto, gli stati del sistema sono dati dalle combinazioni dei sottosistemi. La manopola ha tre stati (0, 400 e 700), mentre hanno due stati il tasto di spegnimento (on e off) e lo sportello (a e c, aperto e chiuso).

In quest'ottica, l'insieme degli stati dell'automa, Q, sarebbe quindi:

$$Q = \{a0on,\ a0off,\ c0on,\ c0off,\ a400on,\ a400off,\ c400on,\ c400off,\ a700on,\ a700off,\ c700on,\ c700off\}$$

dove la prima lettera identifica lo stato dello sportello, il numero la potenza erogata e la stringa finale l'attività del forno. Gli stati *cXon* identificano gli stati in cui il forno eroga le microonde alla potenza *X*. Gli stati *aXon* identificano gli stati in cui il forno ha lo sportello aperto e non eroga le microonde, ma se lo sportello venisse richiuso erogherebbe microonde alla potenza *X*. Questi stati, quindi sono necessari per tenere traccia della potenza da erogare.

Le azioni che si possono effettuare sul forno sono le regolazioni della potenza (0, 400 e 700), l'apertura e la chiusura dello sportello (a e c) e l'attivazione o lo spegnimento (on/off). Quindi, l'alfabeto di input, Σ sarà:

$$\Sigma = \{a,\ c,\ on/off,\ 0,\ 400,\ 700\}$$

La descrizione del funzionamento del forno data nel problema non è molto dettagliata e non precisa il comportamento in alcune situazioni critiche. Ciò lascia spazio ad ipotesi aggiuntive. Per esempio, si può ipotizzare che l'attivazione dell'emissione di microonde sia pericolosa quando lo sportello è aperto e che quindi le sequenze di azioni che portano a questa situazione vadano rifiutate dall'automa. A tal fine, si può aggiungere un altro stato, *errore*, tale per cui una volta raggiunto non lo si possa più lasciare. Analogamente, si può assumere che il tentativo di aprire lo sportello se già fosse aperto e di chiuderlo se già fosse chiuso comportano un errore e, pertanto, portano l'automa nello stato *errore*. Ogni sequenza di azioni che non comporti il raggiungimento dello stato *errore* rappresenta il normale funzionamento del forno. 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 a 0 W, sportello aperto e forno non attivo, *a0off*.

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

δ	*a*	*c*	*on/off*	*0*	*400*	*700*
a0on	*errore*	*c0on*	*a0off*	*a0on*	*a400on*	*a700on*
a0off	*errore*	*c0off*	*a0on*	*a0off*	*a400off*	*a700off*
c0on	*a0on*	*errore*	*c0off*	*c0on*	*c400on*	*c700on*
c0off	*a0off*	*errore*	*c0on*	*c0off*	*c400off*	*c700off*
a400on	*errore*	*c400on*	*a400off*	*a0on*	*a400on*	*a700on*
a400off	*errore*	*c400off*	*a400on*	*a0off*	*a400off*	*a700off*
c400on	*a400on*	*errore*	*c400off*	*c0on*	*c400on*	*c700on*
c400off	*a400off*	*errore*	*c400on*	*c0off*	*c400off*	*c700off*
a700on	*errore*	*c700off*	*a700on*	*a0off*	*a400off*	*a700off*
a700off	*errore*	*c700off*	*a700on*	*a0off*	*a400off*	*a700off*
c700on	*a700on*	*errore*	*c700off*	*c0on*	*c400on*	*c700on*
c700off	*a700off*	*errore*	*c700on*	*c0off*	*c400off*	*c700off*
errore	*errore*	*errore*	*errore*	*errore*	*errore*	*errore*

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

Le specifiche sono abbastanza vaghe da contemplare anche una variante che, sebbene più limitata nelle funzionalità, consente di ridurre il numero di stati dell'automa. Rilassando l'ipotesi di indipendenza tra i sottosistemi del forno, si possono infatti aggiungere le seguenti ipotesi:

- la potenza si può regolare solo dopo aver attivato il forno;
- a forno spento, la potenza vale 0 W.

Con queste ipotesi, il numero di possibili combinazioni diventa molto più piccolo in quanto gli stati *XYoff* collassano negli stati *X0off*.

Con queste ipotesi aggiuntive, l'insieme degli stati dell'automa, Q, sarebbe quindi:

$$Q = \{a,\ c,\ a0,\ c0,\ a400,\ c400,\ a700,\ c700\}$$

dove gli stati a e c indicano la situazione in cui il forno è disattivato (e quindi a potenza 0 W), mentre gli stati aX e cX indicano la situazione in cui il forno è attivo alla potenza X, con lo sportello, rispettivamente, aperto e chiuso.

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 $\Sigma = \{a,\ b,\ c\}$:

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

Quali fra le seguenti stringhe vengono descritte da E?

a) *ccaabccabaa*

b) *cbaacbbca*

c) *cccaabbcca*

d) *bcccc*

e) *baaabbcca*

f) *abbbbccccabaa*

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 $s \subseteq E_1 \subseteq E_2 \subseteq \cdots \subseteq E_k \equiv E$.

a) *ccaabccabaa*
$ccaabccabaa \subseteq (cca)(ab)(cca)(b)(aa) \subseteq (cca+ab+cca)^3(b)(a)^2 \subseteq (c^2a+ab)^3(b)(a+c^2)^2 \subseteq (c^2a+ab^*)^*b(a+c^2)^2$

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

b) *cbaacbbca*
Le stringhe descritte da E devono avere un suffisso appartenente all'insieme $\{aa,\ acc,\ cca,\ cccc\}$ (descritto dalla sottoespressione $(a + c^2)^2$). La stringa *cbaacbbca* non gode di questa proprietà e pertanto

$\boldsymbol{\delta}$	*a*	*c*	*on/off*	*0*	*400*	*700*
a	*errore*	*c*	*a0*	*a*	*a*	*a*
c	*a*	*errore*	*c0*	*c*	*c*	*c*
a0	*errore*	*c0*	*a*	*a0*	*a400*	*a700*
c0	*a0*	*errore*	*c*	*c0*	*c400*	*c700*
a400	*errore*	*c400*	*a*	*a0*	*a400*	*a700*
c400	*a400*	*errore*	*c*	*c0*	*c400*	*c700*
a700	*errore*	*c700*	*a*	*a0*	*a400*	*a700*
c700	*a700*	*errore*	*c*	*c0*	*c400*	*c700*
errore	*errore*	*errore*	*errore*	*errore*	*errore*	*errore*

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

non può essere descritta da E: *cbaacbbca* $\notin \mathcal{L}(E)$.

c) *cccaabbcca*
Le stringhe descritte da E devono avere un prefisso appartenente all'insieme $\{cca, a, b\}$ (descritto dalla sottoespressione $(c^2a + ab^*)^*b$).

La stringa *cccaabbcca* non gode di questa proprietà e pertanto non può essere descritta da E: *cccaabbcca* $\notin \mathcal{L}(E)$.

d) *bcccc*
$bcccc \subseteq (b)(cc)(cc) \subseteq (b)(cc)^2 \subseteq (b)(c^2)^2 \subseteq (b)(a+c^2)^2 \subseteq b(a+c^2)^* \subseteq (c^2a+ab^*)^*b(a+c^2)^2$

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

e) *baaabbcca*
Una stringa descritta da E può iniziare per b solo se la sottoespressione $(c^2a+ab^*)^*$ genera la stringa vuota. In tal caso, però, dopo la b iniziale, non possono esserci più di due a.

La stringa *baaabbcca* non gode di questa proprietà e pertanto non può essere descritta da E: *baaabbcca* $\notin \mathcal{L}(E)$.

f) *abbbbccccabaa*
Una stringa descritta da E non può avere sottostringhe interne composte esclusivamente da simboli c consecutivi che siano più lunghe di due. Infatto i simboli c possono trovarsi solo intervallati da un simbolo a (c^2a) oppure come suffisso ($(a + c^2)^2$).

La stringa *abbbbccccabaa* contiene quattro c consecutivi che non costituiscono il suo suffisso e non può essere perciò descritta da E: *abbbbccccabaa* $\notin \mathcal{L}(E)$.

Esercizio 6

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

- *bbbcbcaaabc*
- *bcbcbcbc*
- *ccbbbabc*
- *bbcbbaaaabcaaabc*

ma non le seguenti:

- *abcccbabc*
- *aaacca*
- *bbcbabb*
- *bcbbbaac*

Soluzione

Si può notare che le stringhe da includere hanno tutte *bc* come suffisso. Inoltre, il primo simbolo non è mai una a. Queste caratteristiche possono essere descritte dall'espressione regolare $(b + c)(a + b + c)^*bc$:

- prefisso diverso da a: $\underline{(b + c)}(a + b + c)^*bc$;
- bc per suffisso: $(b + c)(a + b + c)^*\underline{bc}$;
- qualsiasi sequenza di simboli all'interno: $(b + c)\underline{(a + b + c)^*}bc$.

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

- *abcccbabc*: inizia per a;
- *aaacca*: inizia per a;
- *bbcbabb*: non termina per *bc*;
- *bcbbbaac*: non termina per *bc*.

Un'altra espressione regolare che rispetta le specifiche del problema è $(b + c)^*(a^*bc)^*$.