Corso di Laurea in Sicurezza dei sistemi e delle reti informatiche

Fondamenti di informatica per la sicurezza

anno accademico 2004–2005
docente: Stefano Ferrari

14.01.2005 — Soluzione del secondo compitino — vers. D

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:

- $L_1 = \{ab, b, c\}$
- $L_2 = \{a, c, ca\}$

Descrivere i linguaggi:

a) $L_3 = L_1 \cap L_2$

b) $L_4 = L_1 \cup L_2$

c) $L_5 = L_2 L_1$

d) $L_6 = {L_1}^2$

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

f) $L_8 = (L_1 L_2)^*$

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) $L_3 = L_1 \cap L_2 = \{c\}$

b) $L_4 = L_1 \cup L_2 = \{ab, b, c, a, ca\}$

c) $L_5 = L_2 L_1 = \{aab, ab, ac, cab, cb, cc, caab, cac\}$

d) $L_6 = {L_1}^2 = \{abab, abb, abc, bab, bb, bc, cab, cb, cc\}$

e) $L_7 = {L_2}^* L_1$
L'insieme L_7 è formato da stringhe che hanno un elemento di L_1 preceduto da una concatenazione di un numero arbitrario (eventualmente nullo) di elementi di L_2. Poiché ${L_2}^*$ è composto da infiniti elementi, anche L_7 avrà infiniti elementi. L'insieme $\{b, acacaab, cc\}$ è un sottoinsieme di L_7.

f) $L_8 = (L_1 L_2)^*$
L'insieme L_8 è la chiusura dell'insieme concatenazione di L_1 e L_2. Pertanto, L_8 è composto da infiniti elementi. L'insieme $\{\epsilon, abcab, bcaabc\}$ 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: $P = \{S ::= K,\ K ::= a|Hb|Hc,\ H ::= c|Kd|Hb\}$

Quali fra le seguenti stringhe vengono generate da G?

a) $adbb$

b) $aadb$

c) $ccdbbc$

d) adc

e) cdc

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) *adbb*

	S
$S ::= K$	K
$K ::= Hb$	Hb
$H ::= Hb$	Hbb
$H ::= Kd$	$Kdbb$
$K ::= a$	$adbb$

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

b) *aadb*

	S
$S ::= K$	K
$K ::= Hb$	Hb
$H ::= Kd$	Kdb
$K ::= a$	adb

Non è possibile aggiungere altri simboli, in quanto non ci sono metasimboli nella stringa generata fino a questo punto. La stringa *aadb* non è generata da G: $adbb \notin \mathcal{L}(G)$.

c) *ccdbbc*

	S
$S ::= K$	K
$K ::= Hc$	Hc
$H ::= Hb$	Hbc
$H ::= Hb$	$Hbbc$
$H ::= Kd$	$Kdbbc$
$K ::= Hc$	$Hcdbbc$
$H ::= c$	$ccdbbc$

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

d) *adc*

	S
$S ::= K$	K
$K ::= Hc$	Hc
$H ::= Kd$	Kdc
$K ::= a$	adc

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

e) *cdc*

	S
$S ::= K$	K
$K ::= Hc$	Hc
$H ::= Kd$	Kdc
$K ::= Hc$	$Hcdc$

Non è possibile eliminare il metasimbolo H. Pertanto, la stringa *cdc* non è generata da G: $cdc \notin \mathcal{L}(G)$.

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

- *edccba*
- *aacb*
- *ababc*
- *ab*

b) quattro stringhe rifiutate da A:

- *dacc*
- *abbc*
- *ac*
- *adec*

Esercizio 4

Modellare, tramite un automa a stati finiti deterministico, l'utilizzo di una lavatrice.

La lavatrice può essere sottoposta alle seguenti azioni: programmazione, caricamento, svuotamento, avviamento, terminazione.

Una lavatrice non può essere avviata se non è stata programmata. Avviare una lavatrice che è già in moto non sortisce alcun effetto su di essa, così come terminare una lavatrice già inattiva.

Modellare un automa a stati finiti deterministico dove gli stati rappresentano le condizioni in cui una lavatrice si può trovare (sia in termini di carico che di attività), e la stringa di input rappresenta la sequenza di azioni che possono essere operate sulla lavatrice stessa.

Ipotizzare che non si possano verificare contemporaneamente più azioni. Modellare l'automa in modo che esso accetti solo le stringhe che descrivono un utilizzo accettabile (ed economico: avviare una lavatrice vuota è possibile, ma non ragionevole) della lavatrice.

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.

Nel caso in esame, sono date le seguenti informazioni:

- insieme degli stati, Q:
  $Q = \{p, a, i, v, c\}$
  dove p sta per "programmata", u sta per "attiva", i sta per "inattiva", v sta per vuota e c sta per carica;

- alfabeto di input, Σ:
  $\Sigma = \{a, t, s, c, p\}$
  per indicare, rispettivamente, le azioni di avviamento, di terminazione, di svuotamento di caricamento e di programmazione.

Dalle specifiche è possibile dedurre i seguenti comportamenti:

- la lavatrice può passare allo stato a solo dallo stato p;

- l'avviamento di una lavatrice che si trova in a e la terminazione di una lavatrice nello stato i non modificano lo stato in cui la lavatrice si trova;

- avviare una lavatrice che si trova in v non è corretto.

Per poter catturare le sequenze di azioni ritenute non corrette è necessario introdurre un nuovo stato, e ("errore"), dal quale, una volta raggiunto, l'automa non possa più uscirne.

Si può ipotizzare che lo stato iniziale sia quello relativo alla lavatrice vuota ($q_0 = v$), e che gli stati finali siano tutti, escluso e ($F = \{p, a, i, v, c\}$).

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

Molte varianti sono possibili. Per esempio, per rendere il modello più dettagliato e vicino alla realtà, si possono ritenere le operazioni di caricamento e di programmazione indipendenti tra loro (e l'avviamento ha effetto solo se entrambe sono state realizzate). Altre varianti possono essere costruite non considerando alcune operazioni come causa di errore (per esempio, una lavatrice carica può essere svuotata).

Esercizio 5

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

- $E = (ab + cb)^* + (ac)^3 a^*$

Quali fra le seguenti stringhe vengono descritte da E?

a) *abcbabab*

b) *acacbabb*

c) *bababacc*

d) *acacac*

e) *acacacaaa*

f) *acac*

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

$\boldsymbol{\delta}$	a	t	s	c	p
p	a	c	e	e	p
a	a	i	e	e	e
i	e	i	v	e	e
v	e	e	v	c	e
c	e	e	e	e	p
e	e	e	e	e	e

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

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) *abcbabab*
$abcbabab \subseteq (ab)(cb)(ab)(ab) \subseteq (ab+cb)^4 \subseteq (ab+cb)^* \subseteq (ab+cb)^*(ab+cb)^* + (ac)^3a^*$

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

b) *acacbabb*
L'espressione regolare E descrive stringhe che sono scomponibili in sequenze di ab e cb $(\underline{(ab+cb)^*}+(ac)^3a^*)$. oppure sono sequenze (di lunghezza indefinita) di a prefissate dalla stringa *acacac* $((ab+cb)^* + \underline{(ac)^3a^*})$.

La stringa *acacbabb* non possiede alcuna di queste caratteristiche, e, quindi, non può essere descritta da E: $acacbabb \notin \mathcal{L}(E)$.

c) *babababcc*
Le stringhe descritte da E devono avere come primo simbolo a o c $((\underline{a}b + \underline{c}b)^* + (\underline{a}c)^3a^*)$.

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

d) *acacac*
$acacac \subseteq (ac)(ac)(ac) \subseteq (ac)^3 \subseteq (ac)^3a^* \subseteq (ab+cb)^* + (ac)^3a^*$

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

e) *acacacaaa*
$acacacaaa \subseteq (ac)(ac)(ac)(a)(a)(a) \subseteq (ac)^3a^3 \subseteq (ac)^3a^* \subseteq (ab+cb)^* + (ac)^3a^*$

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

f) *acac*
L'espressione regolare E descrive stringhe che sono scomponibili in sequenze di ab e cb $(\underline{(ab+cb)^*}+(ac)^3a^*)$. oppure sono sequenze (di lunghezza indefinita) di a prefissate dalla stringa *acacac* $((ab+cb)^* + \underline{(ac)^3a^*})$.

Non possedendo alcuna di queste caratteristiche, la stringa *acac* non può essere descritta da E: $acac \notin \mathcal{L}(E)$.

Esercizio 6

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

- *aaacbacb*
- *babbbb*
- *abbbacac*
- *bbbacacb*

ma non le seguenti:

- *cacaacb*
- *bcbacb*
- *cbbacba*
- *abcacab*

Soluzione

Si può notare che le stringhe da includere hanno tutte il prefisso di lunghezza 2 composto da a e b. Inoltre, nella rimanente sottostringa il simbolo c è sempre preceduto da a.Queste caratteristiche possono essere descritte dall'espressione regolare $(a+b)^2(ac+a+b)^*$:

- a e b come prefisso di lunghezza 2: $\underline{(a+b)^2}(ac+b)^*$;

- c preceduto da a: $(a+b)^2\underline{(ac+a+b)^*}$.

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

- *cacaacb*: c'è c nel prefisso di lunghezza 2;
- *bcbacb*: c'è c nel prefisso di lunghezza 2;
- *cbbacba*: c'è c nel prefisso di lunghezza 2;
- *abcacab*: c'è c non preceduto da a nella sottostringa ottenuta eliminando i primi due simboli.

Altre espressioni regolari che rispettano le specifiche del problema possono essere:

- $(a+b)^2(ac+b)^*$,
- $(a^*b^*)^*(acb^*)^*$,
- $(a+b)^2b^*((ac)^*b^*)^2$,
- $(a+b)^3(a+b+c)^*$.