

## Chapter 9

# Linguaggi Formali e Compilatori - Prof. Breveglieri e Crespi Reghizzi - Soluzione Prova scritta 07/02/2005

Tempo 2 ore per chi è esonerato dalle prime due parti, 3 ore per gli altri.

AVVERTENZA: L'esame è diviso in 5 parti

1. Espr. reg. e aut. finiti: *Esonerato chi ha superato il compito*
2. Grammatiche *Esonerato chi ha superato il compito*
3. Esercitazioni Flex Bison (fascicolo separato) *Obbligatorio per tutti*
4. Grammatiche e analisi sintattica *Obbligatorio per tutti*
5. Traduzione e semantica *Obbligatorio per tutti*

## 9.1 Espressioni regolari e automi finiti 20%

1. Trovare la (o le) stringa più breve che appartiene al linguaggio di alfabeto  $\{a, b\}$  generato dall'espressione regolare  $R_1$ :

$$R_1 = (ab^+) (\neg(ab \mid b^*) ((b^*ab^*a)^+ \cap (aaa)^*))$$

**Soluzione**

$$\text{min di } (ab^+) = ab$$

$$\text{min di } (\neg(ab \mid b^*)) = a$$

$$\text{min di } ((b^*ab^*a)^+ \cap (aaa)^*) = aaaaaa$$

La frase più breve è pertanto  $aba^7$

2. Trovare la (o le) stringa più breve che appartiene al linguaggio di alfabeto  $\{a, b\}$  generato dall'espressione regolare  $R_2$ :

$$R_2 = \neg(a^* \mid (a^*b^+)^+)^* \mid b(b \mid a)^+bb^+$$

**Soluzione** essendo il primo termine

$$\neg(a^* \mid (a^*b^+)^+)^*$$

il complemento del linguaggio universale, esso è vuoto. Dal secondo termine

$$b(b \mid a)^+bb^+$$

si ha:

$$\min \text{ di } b = b$$

$$\min \text{ di } (b \mid a)^+ = a, b$$

$$\min \text{ di } bb^+ = bb$$

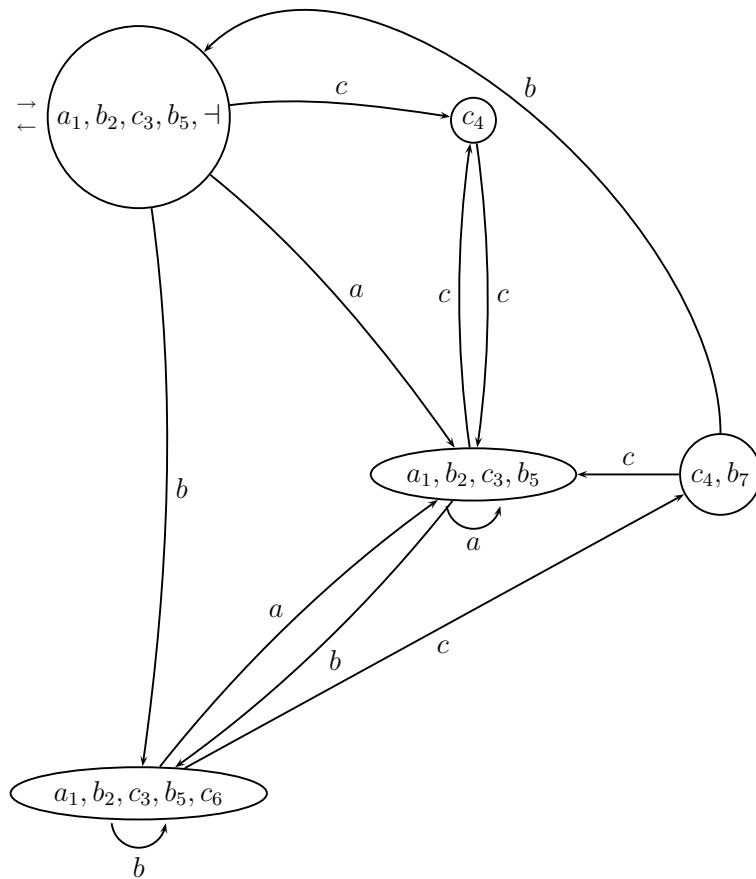
La risposta è pertanto *babb* e *bbbb*

3. È data l'espressione regolare seguente:

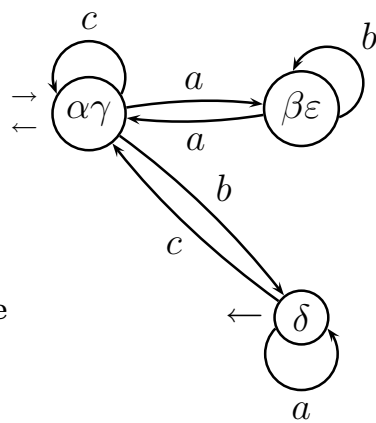
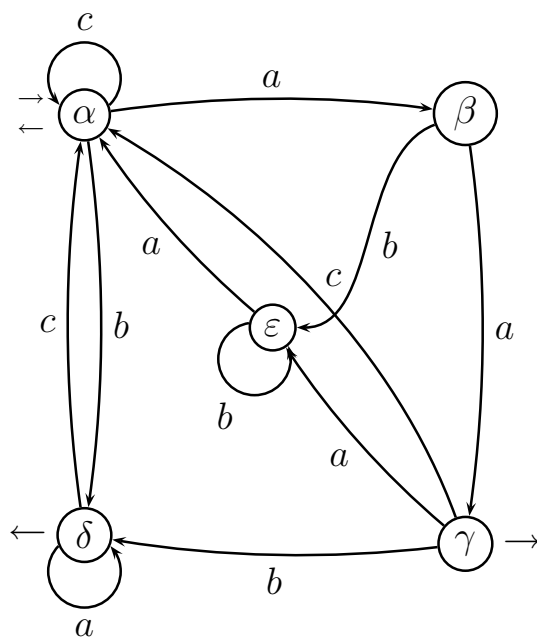
$$R = ((a \mid b^* \mid cc)^* bcb)^*$$

Tramite l'algoritmo di McNaughton-Yamada costruire l'automa deterministico che riconosce il linguaggio regolare  $L(R)$ .

**Soluzione**



4. Minimizzare l'automa seguente.



## 9.2 Grammatiche 20%

1. Si consideri il ling. di Dyck di alfabeto  $\{ '(', ')', '[, ']' \}$ .
  - (a) Trovare la grammatica BNF che genera tutte e sole le stringhe di Dyck in cui la parentesi tonda chiusa deve essere immediatamente seguita dalla parentesi quadra aperta.
  - (b) Generalizzando il caso precedente, trovare la grammatica BNF che genera tutte e sole le stringhe di Dyck in cui la parentesi tonda chiusa deve essere immediatamente seguita dalla parentesi quadra aperta oppure dalla parentesi tonda chiusa.

### Soluzione

(a)

$$S \rightarrow \{S\}S \mid [S]S \mid (S)Q \mid \varepsilon$$

$$Q \rightarrow [S]S$$

(b)

$$S \rightarrow \{S\}S \mid [S]S \mid (T)Q \mid \varepsilon$$

$$Q \rightarrow [S]S \quad T \rightarrow (T) \mid S \mid \varepsilon$$

2. Costruire la grammatica EBNF della dichiarazione della **struct** del ling. C, con le regole seguenti:

- (a) l'alfabeto comprende le lettere minuscole, le cifre, le due parentesi '{' e '}' e punto-e-virgola
- (b) i tipi semplici per i campi delle struct sono: **int**, **char**, **float**
- (c) le **struct** possono essere annidate
- (d) la grammatica non deve essere ambigua
- (e) esempio

```
struct {
    int campo1;
    char campo2;
    struct {
        float elema;
        char elemb;
    } alfa;
    float campo3;
} beta;
```

**Soluzione**

$$\begin{aligned}
 S &\rightarrow \text{struct } \{F\}I; \\
 F &\rightarrow ((\text{int} \mid \text{char} \mid \text{float})I; \mid S)^+ \\
 I' &\rightarrow [a \dots z, A \dots Z][a \dots z, A \dots Z, 0 \dots 9]^*
 \end{aligned}$$

### 9.3 Domanda relativa alle esercitazioni

Vedi fascicolo separato.

### 9.4 Grammatiche e analisi sintattica 20%

1. Calcolare gli insiemi guida e verificare se le regole di produzione soddisfano la condizione LL(1).

Regola	Insieme Guida
$S \rightarrow ASB$	
$S \rightarrow c$	
$A \rightarrow c$	
$A \rightarrow aA$	
$A \rightarrow B$	
$B \rightarrow bBc$	
$B \rightarrow \varepsilon$	

### Soluzione

Regola	Insieme Guida
$S \rightarrow ASB$	$\text{Ini}(ASB) = \{a, b, c\}$
$S \rightarrow c$	$c$
$A \rightarrow c$	$c$
$A \rightarrow aA$	$a$
$A \rightarrow B$	$\text{Ini}(B) \cup \text{Seg}(A) = \{b\} \cup \{a, b, c\} = \{a, b, c\}$
$B \rightarrow bBc$	$b$
$B \rightarrow \varepsilon$	$\text{Seg}(B) \cup \text{Seg}(A) \cup \text{Seg}(S) = \{c, \vdash\} \cup \{a, b, c\} \cup \{b, \vdash\} = \{a, b, c, \vdash\}$

Tutti i nonterminali violano la condizione LL(1).

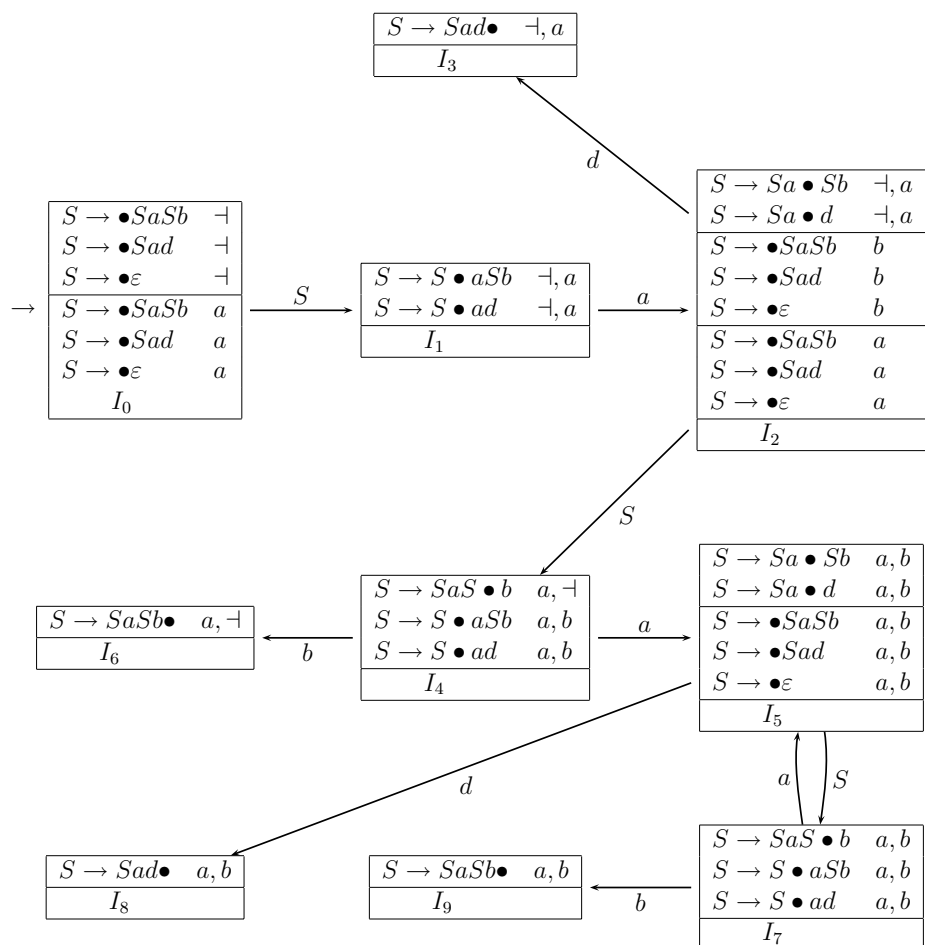
Nota: la grammatica è ambigua.



2. Costruire il riconoscitore deterministico dei prefissi ascendenti e indicare in quali stati sono violate le condizioni LR(1), LALR(1) e LR(0).

$$S \rightarrow SaSb \quad S \rightarrow Sad \quad S \rightarrow \varepsilon$$

## Soluzione



Tutti e soli gli stati con la candidata  $S \rightarrow \bullet \varepsilon$  non sono LR(0).

Tutti gli stati sono LR(1): ad es. in  $I_5$  il conflitto LR(0) tra la riduzione  $S \rightarrow \bullet \varepsilon$  e lo spostamento, scompare prospettando il successivo carattere

$d$ , l'unica etichetta terminale di un arco uscente.

La grammatica è anche LALR(1). Infatti fondendo insieme gli stati aventi lo stesso insieme di regole marcate:

$$I_2, I_5 \quad I_3, I_8 \quad I_4, I_7 \quad I_4, I_9$$

e unendo gli insiemi di prospezione a parità di regola marcata, si ottiene il riconoscitore LALR(1), in cui nessuno stato ha conflitto spostamento-riduzione. Si può anche osservare che il riconoscitore LR(1) è privo di stati contenenti due riduzioni, condizione che permette di affermare che la grammatica è LALR(1)

3. Domanda facoltativa. Dato il linguaggio delle espressioni in forma polacca postfissa con un operatore ‘+’ e una variabile  $v$ , trovare una grammatica LL(1) che lo generi.

Esempio di frase del linguaggio:  $vv + v +$

**Soluzione** La grammatica più immediata del linguaggio

$$E \rightarrow EE + \quad E \rightarrow v$$

è ricorsiva a sx, quindi inadatta all’analisi LL(1). Trasformando la trasformazione la ricorsione sx in dx, si ottiene la grammatica LL(1) seguente:

	Insieme Guida
$E \rightarrow vE'$	$v$
$E' \rightarrow E + E'$	$v$
$E' \rightarrow \varepsilon$	$\neg!, +$

## 9.5 Traduzione e semantica 20%

1. Dati gli alfabeti sorgente e pozzo

$$\Sigma = \{a, b, c\} \quad \Delta = \{a, b\}$$

considerare la traduzione

$$\tau(ucv) = vu^R \quad \text{dove } u, v \in \{a, b\}^+$$

- (a) Scrivere uno schema di traduzione (oppure una gramm. di traduzione) puramente sintattico per definire la traduzione  $\tau$  e disegnare l'albero sintattico della traduzione

$$\tau(abcba) = babaa$$

- (b) Verificare se è possibile calcolare la traduzione con un trasduttore a pila deterministico, ovvero con un parsificatore LL(1) o LR(1).

## Soluzione

- (a) La grammatica di traduzione (con i caratteri dell'alfabeto pozzo sottolineati) è:

	Insieme Guida LL(2)
$S \rightarrow aY\underline{a}$	$a$
$S \rightarrow bY\underline{b}$	$b$
$Y \rightarrow aY\underline{a}$	$a$
$Y \rightarrow bY\underline{b}$	$b$
$Y \rightarrow cX$	$c$
$X \rightarrow a\underline{a}X$	$\{aa, ab\}$
$X \rightarrow b\underline{b}X$	$\{ba, bb\}$
$X \rightarrow a\underline{a}$	$a \dashv$
$X \rightarrow b\underline{b}$	$b \dashv$

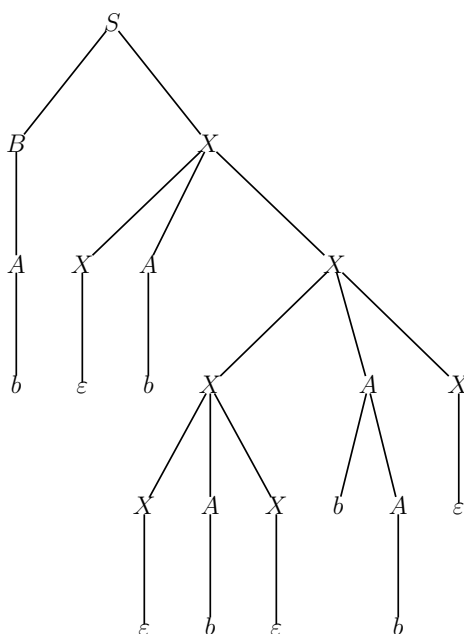
- (b) Essendo la sintassi deterministica LL(2), la traduzione può essere calcolata da un parsificatore a discesa ricorsiva, e quindi anche da un trasduttore a pila deterministico.

2. È data la Gramm. ad Attributi:

$S_0 \rightarrow B_1 X_2$	$\delta_0 \leftarrow \delta_2 \quad \gamma_2 \leftarrow \beta_1$
$X_0 \rightarrow X_1 A_2 X_3$	$\delta_0 \leftarrow \delta_1 + \delta_3 + (\text{if } (\gamma_0 = \alpha_2) \text{ then } 1 \text{ else } 0)$ $\gamma_1 \leftarrow \gamma_0 \quad \gamma_3 \leftarrow \gamma_0$
$X_0 \rightarrow \varepsilon$	$\delta_0 \leftarrow 0$
$B_0 \rightarrow A_1$	$\beta_0 \leftarrow \alpha_1$
$A_0 \rightarrow b A_1$	$\alpha_0 \leftarrow \alpha_1 + 1$
$A_0 \rightarrow b$	$\alpha_0 \leftarrow 1$

(a) Indicare gli attributi ereditati e sintetizzati

(b) Disegnare sull'albero gli attributi e le frecce delle dipendenze funzionali tra di essi.



(c) Quanto vale l'attributo  $\delta$  nella radice?

- (d) Verificare se la gramm. a attributi è del tipo L e scrivere almeno in parte le procedure del valutatore ricorsivo.
- (e) Domanda facoltativa. Spiegare che significato ha il calcolo sugli alberi astratti, definito dalla gramm. a attributi, e che cosa il valore di  $\delta$  rappresenta.
- Scrivere una gramm. ad attributi puramente sintetizzata che calcola lo stesso valore di  $\delta$ .

### Soluzione

- |  | sintetizzati   | ereditati        |
|--|--|------------------|
| (a) Indicare gli attributi ereditati e sintetizzati: | $\delta$ per $S, X$<br>$\beta$ per $B$<br>$\alpha$ per $A$ | $\gamma$ per $X$ |
- (b) Disegnare sull'albero gli attributi e le frecce delle dipendenze funzionali tra di essi.
- (c) Quanto vale l'attributo  $\delta$  nella radice? 2
- (d) • La gramm. a attributi è del tipo L:
- l'attrib. ered. dipende nella 2 dall'attrib ered. del padre,  
 nella 1 dall'attrib. sint. del fratello precedente;
- l'attrib. sint.  $\delta$  (nella 2) dipende dall'attributo ered. del padre e da attrib dei figli;
- gli altri attrib. sint. dipendono solo da attrib. sintet.
- scrivere almeno in parte le procedure del valutatore ricorsivo:

Poiché la sintassi è astratta e ambigua, non si può costruire un parsificatore integrato con il valutatore. Il valutatore opera dunque sull'albero sintattico già costruito. Il suo codice contiene una procedura per simbolo nonterminale, avente i seguenti argomenti:

un puntatore al nodo dell'albero;

un argomento d'ingresso per l'attributo ereditato  $\gamma$  dove è rilevante;

un argomento d'uscita per l'attributo sintetizzato.

Il corpo delle procedure  $A$  e  $X$  contiene un condizionale guidato dalla produzione sintattica applicata nel nodo.

(e) Domanda facoltativa.

- Spiegare che significato ha il calcolo sugli alberi astratti, definito dalla gramm. a attributi, e che cosa il valore di  $\delta$  rappresenta. Il sottoalbero  $B$  è il numero  $n$  (rappresentato dalla stringa  $a^n$ ) da cercare nel resto dell'albero, che non è altro che un albero binario con un numero  $m$  (la stringa  $a^m$ ) associato a ogni nodo interno.

Il valore di  $\delta$  è il numero di comparse di  $n$  nell'albero.

- Scrivere una gramm. ad attributi puramente sintetizzata che calcola lo stesso valore di  $\delta$ :



Gli attributi  $\alpha$  e  $\beta$  si conservano, e si eliminano gli altri. Si usa un attributo sintet.  $\mu$ , un multi-insieme di interi, che conterrà il numero di comparse delle stringhe  $a^m$  presenti. Nella produzione

$$X_0 \rightarrow X_1 A_2 X_3$$

l'attributo  $\mu_0$  è calcolato come unione dei multi-insiemi  $\mu_1, \mu_3$  e dell'insieme  $\{\alpha_2\}$

Nella radice, si assegna a  $\delta_0$  la molteplicità di  $\beta_1$  in  $\mu_2$ .

Nota: l'eliminazione degli attributi ereditati ha richiesto l'uso di un attributo sintetizzato di dominio complesso (multi-insieme).