Corso di Linguaggi di Programmazione — Parziale di fine modulo
Prova scritta **B** del 18 Dicembre 2017

Tempo a disposizione: 2 ore e 30 minuti.

1. Sia P^{L_1} un programma scritto in L_1. La seguente espressione

$$\mathcal{I}^{L_0}_{L_1}(\mathcal{C}^{L_1}_{L_1,L_2}, P^{L_1})$$

   cosa produce? Se non avessimo a disposizione un interprete per il linguaggio L_2, avremmo comunque la possibilità, con i programmi a disposizione nell'espressione sopra, di eseguire P^{L_1}?

2. Descrivere le regole di semantica operazionale strutturata per l'espressione aritmetica $e_0 * e_1$, secondo la disciplina di valutazione esterna-destra (ED). Mostrare un esempio di una espressione di quel tipo tale che la valutazione ED e quella ID (interna-destra) non sono uguali.

3. Costruire una grammatica G che generi il linguaggio $L = \{a^n b^{2m} c^{n+1} \mid n, m \geq 0\}$.

4. Classificare il linguaggio L del punto precedente, ovvero dire se L è regolare, oppure libero ma non regolare, oppure non libero, giustificando adeguatamente la risposta.

5. Si consideri l'espressione regolare $a^*(a|b)a$. Si costruisca l'automa NFA M associato, secondo la costruzione vista a lezione. Si trasformi l'NFA M nell'equivalente DFA M', secondo la costruzione per sottoinsiemi vista a lezione.

6. Preso il DFA M' calcolato al punto precedente, si ricavi da esso la grammatica regolare associata, seguendo la costruzione vista a lezione; quindi si semplifichi la grammatica ottenuta, eliminando i simboli inutili; infine, si ricavi da quella grammatica l'espressione regolare associata.

7. Se L ed R sono linguaggi regolari, il linguaggio $R \setminus L = \{w \in A^* \mid w \in R \land w \notin L\} = R \cap \overline{L}$ è regolare o libero, oppure non libero? Giustificare la risposta.

8. Mostrare che $L = \{a^n b^{2n+1} \mid n \geq 0\}$ è libero deterministico, costruendo un opportuno DPDA che riconosca $L\$$ per pila vuota.

9. Si consideri la seguente grammatica G con simbolo iniziale S:

$$\begin{array}{lcl} S & \to & AC \\ A & \to & \epsilon \mid \mathtt{a}SA \\ B & \to & \epsilon \mid \mathtt{b}B \\ C & \to & \mathtt{cc} \mid \mathtt{c}BC \end{array}$$

   (*i*) Si calcolino i First e i Follow per tutti i nonterminali. (*ii*) La grammatica G è di classe LL(1)? (*iii*) Si rimuovano le produzioni epsilon per ottenere una grammatica G' senza produzioni epsilon, che sia equivalente a G.

10. Si consideri la grammatica G con simbolo iniziale S:

$$\begin{array}{lcl} S & \to & S\mathtt{a} \mid \mathtt{b}B \\ B & \to & \epsilon \mid \mathtt{b}B \end{array}$$

   (*i*) Determinare il linguaggio generato $L(G)$. (*ii*) Verificare che G non è di classe LL(1). (*iii*) Manipolare la grammatica per ottenerne una equivalente G' di classe LL(1). (*iv*) Costruire il parser LL(1) per G'. (*v*) Mostrare il funzionamento del parser LL(1) su input baa.

11. Si consideri la grammatica G del punto precedente. (*i*) Costruire l'automa canonico LR(0). (*ii*) Costruire la tabella di parsing SLR(1) e verificare se ci sono conflitti. (*iii*) Mostrare il funzionamento del parser SLR(1) per l'input baa.

Corso di Linguaggi di Programmazione — Parziale di fine modulo
Prova scritta **B** del 18 Dicembre 2017

Tempo a disposizione: 2 ore e 30 minuti.

1. Sia P^{L_1} un programma scritto in L_1. La seguente espressione

$$\mathcal{I}_{L_1}^{L_0}(\mathcal{C}_{L_1,L_2}^{L_1}, P^{L_1})$$

   cosa produce? Se non avessimo a disposizione un interprete per il linguaggio L_2, avremmo comunque la possibilità, con i programmi a disposizione nell'espressione sopra, di eseguire P^{L_1}?

2. Descrivere le regole di semantica operazionale strutturata per l'espressione aritmetica $e_0 * e_1$, secondo la disciplina di valutazione esterna-destra (ED). Mostrare un esempio di una espressione di quel tipo tale che la valutazione ED e quella ID (interna-destra) non sono uguali.

3. Costruire una grammatica G che generi il linguaggio $L = \{a^n b^{2m} c^{n+1} \mid n, m \geq 0\}$.

4. Classificare il linguaggio L del punto precedente, ovvero dire se L è regolare, oppure libero ma non regolare, oppure non libero, giustificando adeguatamente la risposta.

5. Si consideri l'espressione regolare $a^*(a|b)a$. Si costruisca l'automa NFA M associato, secondo la costruzione vista a lezione. Si trasformi l'NFA M nell'equivalente DFA M', secondo la costruzione per sottoinsiemi vista a lezione.

6. Preso il DFA M' calcolato al punto precedente, si ricavi da esso la grammatica regolare associata, seguendo la costruzione vista a lezione; quindi si semplifichi la grammatica ottenuta, eliminando i simboli inutili; infine, si ricavi da quella grammatica l'espressione regolare associata.

7. Se L ed R sono linguaggi regolari, il linguaggio $R \setminus L = \{w \in A^* \mid w \in R \wedge w \notin L\} = R \cap \overline{L}$ è regolare o libero, oppure non libero? Giustificare la risposta.

8. Mostrare che $L = \{a^n b^{2n+1} \mid n \geq 0\}$ è libero deterministico, costruendo un opportuno DPDA che riconosca $L\$$ per pila vuota.

9. Si consideri la seguente grammatica G con simbolo iniziale S:

$$\begin{array}{lcl} S & \to & AC \\ A & \to & \epsilon \mid \mathtt{a}SA \\ B & \to & \epsilon \mid \mathtt{b}B \\ C & \to & \mathtt{cc} \mid \mathtt{c}BC \end{array}$$

   (*i*) Si calcolino i First e i Follow per tutti i nonterminali. (*ii*) La grammatica G è di classe LL(1)? (*iii*) Si rimuovano le produzioni epsilon per ottenere una grammatica G' senza produzioni epsilon, che sia equivalente a G.

10. Si consideri la grammatica G con simbolo iniziale S:

$$\begin{array}{lcl} S & \to & S\mathtt{a} \mid \mathtt{b}B \\ B & \to & \epsilon \mid \mathtt{b}B \end{array}$$

   (*i*) Determinare il linguaggio generato $L(G)$. (*ii*) Verificare che G non è di classe LL(1). (*iii*) Manipolare la grammatica per ottenerne una equivalente G' di classe LL(1). (*iv*) Costruire il parser LL(1) per G'. (*v*) Mostrare il funzionamento del parser LL(1) su input *baa*.

11. Si consideri la grammatica G del punto precedente. (*i*) Costruire l'automa canonico LR(0). (*ii*) Costruire la tabella di parsing SLR(1) e verificare se ci sono conflitti. (*iii*) Mostrare il funzionamento del parser SLR(1) per l'input *baa*.

1) $I^{L_0}_{L_1}\left(C^{L_1}_{L_1,L_2},\ P^{L_1}\right) = P^{L_2}$ dove P^{L_2} è un progr. equivalente a P^{L_1} ma scritto in L_2

Se non abbiamo a disposizione un interprete per L_2, possiamo comunque eseguire P^{L_1} perché già abbiamo $I^{L_0}_{L_1}$, cioè un interprete in grado di eseguire programmi scritti in L_1

2)
$$\frac{\langle e_1, \sigma\rangle \rightarrow \langle e_1', \sigma'\rangle}{\langle e_0 * e_1, \sigma\rangle \rightarrow \langle e_0 * e_1', \sigma'\rangle}$$

$$\frac{}{\langle e_0 * 0, \sigma\rangle \rightarrow \langle 0, \sigma\rangle} \qquad \frac{}{\langle e_0 * 1, \sigma\rangle \rightarrow \langle e_0, \sigma\rangle}$$

$$\frac{\langle e_0, \sigma\rangle \rightarrow \langle e_0', \sigma'\rangle}{\langle e_0 * n, \sigma\rangle \rightarrow \langle e_0' * n, \sigma'\rangle}\ n \neq 0,1 \qquad \frac{}{\langle m * n, \sigma\rangle \rightarrow \langle p, \sigma\rangle}\ \begin{array}{l} n \neq 0,1 \\ p = m * n \end{array}$$

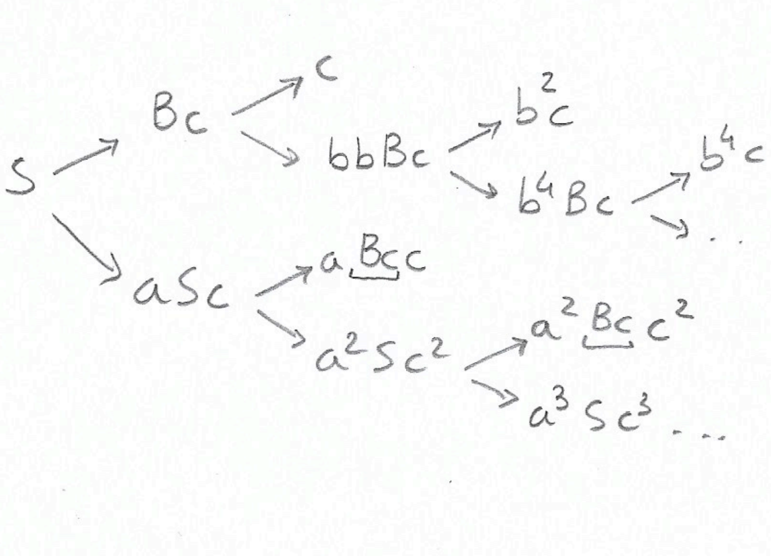
$$\langle (2-5) * 0, \sigma\rangle \xrightarrow[ED]{} \langle 0, \sigma\rangle$$

$$\langle (2-5) * 0, \sigma\rangle \not\xrightarrow[ID]{}$$

3) $L = \{ a^n b^{2m} c^{m+1} \mid n, m \geq 0 \}$

$S \rightarrow aSc \mid Bc$

$B \rightarrow \varepsilon \mid bbB$



$Bc \rightsquigarrow b^{2m}c \mid m \geq 0$

$a^n \underbrace{b^{2m}c}\, c^n = a^n b^{2m} c^{n+1}$

4) $L = \{ a^n b^{2m} c^{n+1} \mid n, m \geq 0 \}$ è libero, perché generato da una gr. libera.

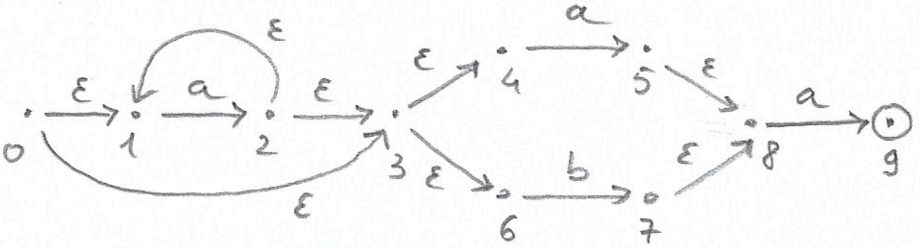
L è non regolare per il PL a rovescio.

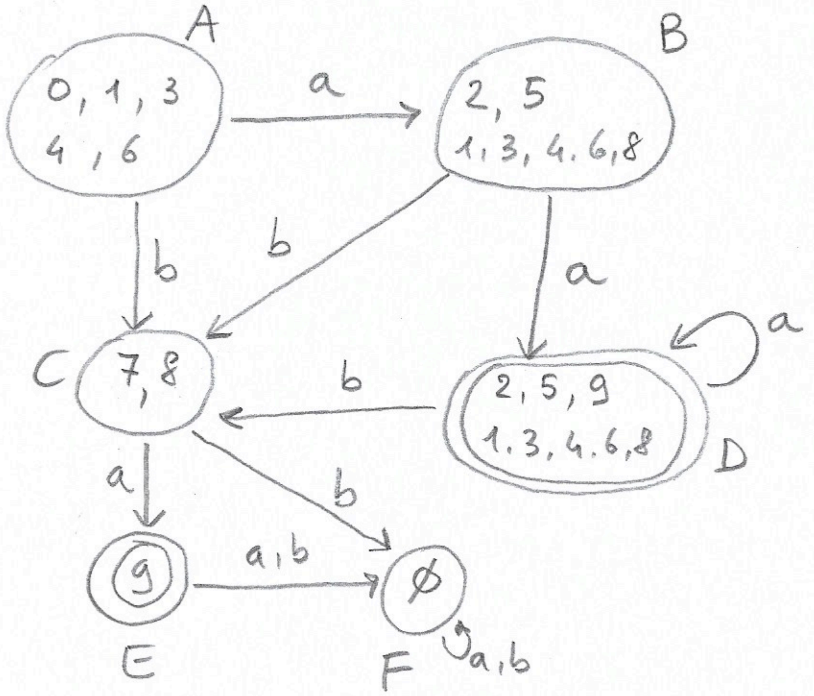
- Fissiamo $N > 0$ generico
- Scegliamo $z = a^N b^{2N} c^{N+1}$ $\quad z \in L$ e $|z| \geq N$
- Per ogni u, v, w tali che $z = uvw$, $|uv| \leq N$ e $|v| \geq 1$, deve essere $v \in a^*$. Sia $v = a^j$ con $j \geq 1$.
- Allora $\exists k = 2$. $uv^2w \notin L$. Infatti

$$uv^2w = a^{N+j} b^{2N} c^{N+1} \notin L$$

$\Rightarrow$ L non è regolare

5) $a^*(a|b)a$





) $A \to aB \mid bC$

$B \to aD \mid bC$

$C \to aE \mid \cancel{bF}$

$D \to aD \mid bC \mid \varepsilon$

$E \to \cancel{aF} \mid \cancel{bF} \mid \varepsilon$

$\cancel{F \to aF \mid bF}$

$A \to aB \mid bC$

$B \to aD \mid bC$

$C \to aE$

$D \to aD \mid bC \mid \varepsilon$

$E \to \varepsilon$

$C = a$

$E = \varepsilon$

Sostituendo

$A \to aB \mid ba$

$B \to aD \mid ba$

$C \to a$

$D \to aD \mid ba \mid \varepsilon$

$E \to \varepsilon$

$D = a^*(ba \mid \varepsilon)$

$\Rightarrow B = aa^*(ba \mid \varepsilon) \mid ba$

$\Rightarrow A = a\left(aa^*(ba \mid \varepsilon) \mid ba\right) \mid ba$

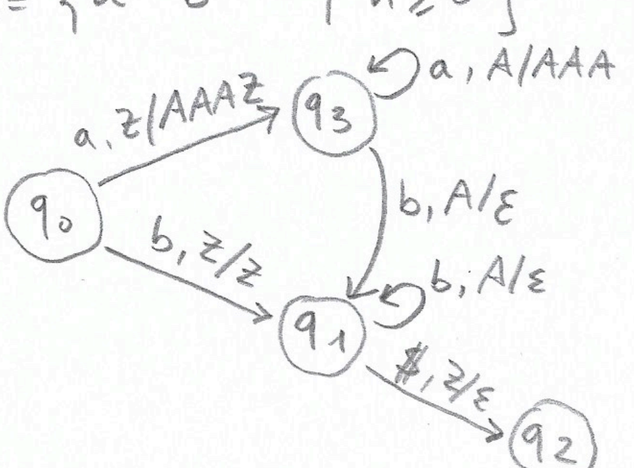
---

7) L regolare R regolare $R \wedge \bar{L}$ è regolare perché

- $\bar{L}$ è regolare, perché i ling. reg. sono chiusi per complementazione
- $R \wedge \bar{L}$ è regolare, perché i ling. reg. sono chiusi per intersezione

---

8) $L = \{a^n b^{2n+1} \mid n \geq 0\}$



9)

$$G \left[\begin{array}{l} S \to AC \\ A \to \varepsilon \mid aSA \\ B \to \varepsilon \mid bB \\ C \to cc \mid cBC \end{array} \right.$$

	First	Follow
S	a, c	\$, a, c
A	a, ε	c
B	b, ε	c
C	c	\$, a, c

G non è di classe LL(1), perché, ad es.

$C \to cc \mid cBC$ crea conflitti.

$N(G) = \{A, B\}$

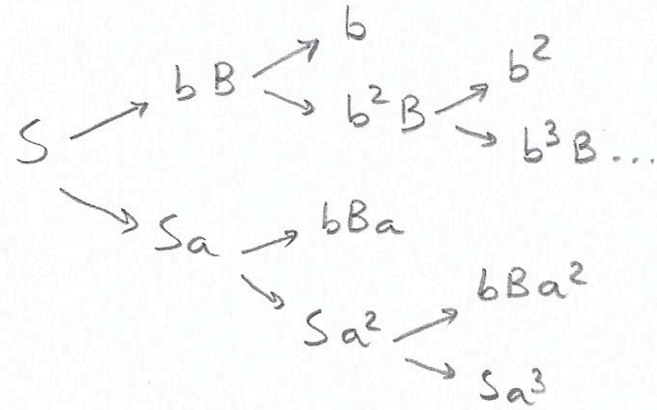
$$G' \left[\begin{array}{l} S \to AC \mid C \\ A \to aSA \mid aS \\ B \to bB \mid b \\ C \to cc \mid cBC \mid cC \end{array} \right.$$

$L(G) = L(G')$

per un teorema visto a lezione

10) $S \to Sa \mid bB$

$B \to \varepsilon \mid bB$



$\underbrace{bB} = b^+$

$b^+ a^+$

$b^+ \mid b^+a^+$

$= b^+(\varepsilon \mid a^+)$

$= b^+a^*$

$L(G) = \{ b^n a^m \mid n \geq 1, m \geq 0 \}$

G non è LL(1) perché è ricorsiva sx

$\underbrace{S \to Sa}$

$$G' \begin{cases} S \to bBS' \\ S' \to aS' \mid \varepsilon \\ B \to \varepsilon \mid bB \end{cases}$$

	First	Follow
S	b	\$
S'	a, ε	\$
B	b, ε	a, \$

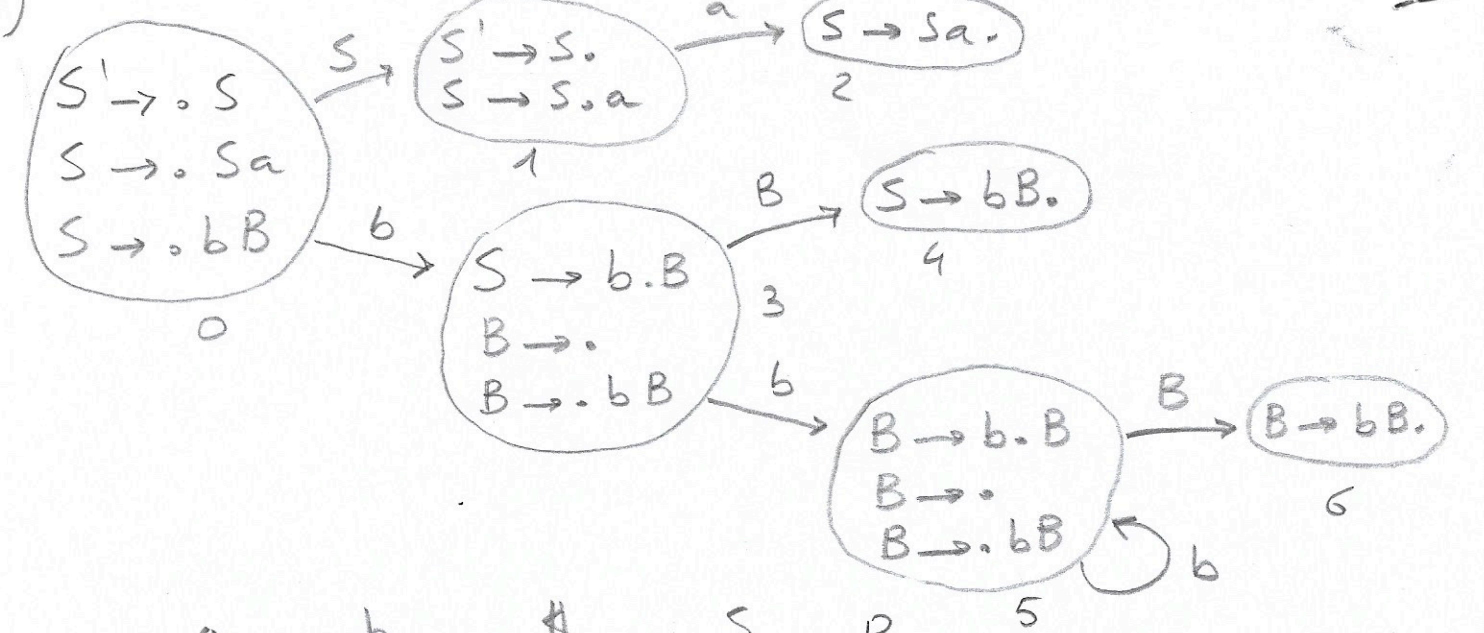
$$G' \text{ è } LL(1) \Leftarrow \begin{cases} First(aS') \cap First(\varepsilon) = \emptyset \\ First(aS') \cap Follow(S') = \emptyset \\ First(\varepsilon) \cap First(bB) = \emptyset \\ Follow(B) \cap First(bB) = \emptyset \end{cases}$$

	a	b	\$
S		$S \to bBS'$	
S'	$S' \to aS'$		$S' \to \varepsilon$
B	$B \to \varepsilon$	$B \to bB$	$B \to \varepsilon$

<u>b</u>aa\$	S
	<u>b</u>BS'
<u>a</u>a\$	BS'
	S'
	<u>a</u>S'
<u>a</u>\$	S'
	<u>a</u>S'
\$	S'
	ε

accettata

11)



	a	b	$	S	B
0		S3		g1	
1	S2		acc		
2	R1		R1		
3	R3	S5	R3		g4
4	R2		R2		
5	R3	S5	R3		g6
6	R4		R4		

$S \to \overset{(1)}{Sa} \mid \overset{(2)}{bB}$

$B \to \underset{(3)}{\varepsilon} \mid \underset{(4)}{bB}$

	First	Follow
S	b	a, $
B	b, ε	a, $

(0, ε, baa$)

(03, b, aa$)

→ B

(034, bB, aa$)

→ S

(01, S, aa$)

(012, Sa, a$)

→ S

(01, S, a$)

(012, Sa, $)

→ S

(01, S, $)

ACCEPT!