CORSO DI LINGUAGGI DI PROGRAMMAZIONE — PARZIALE A-L DI FINE MODULO
PROVA SCRITTA DEL 20 DICEMBRE 2022

Tempo a disposizione: 2 ore e 30 minuti.

1. Per quali valori delle variabili X e Y la seguente espressione
$$\mathcal{I}^{L\delta}_{L_1}(\mathcal{C}^{X}_{L_1,L_2}, \mathcal{C}^{Y}_{X,Y})$$
non produce errore? È utile il programma che viene calcolato?

2. Descrivere le regole di semantica operazionale strutturata per l'espressione booleana b_0 or b_1, secondo la disciplina di valutazione interna-parallela (IP). Mostrare un esempio di una espressione di quel tipo tale che la valutazione IP e quella ES (esterna-sinistra, vista a lezione) non sono uguali.

3. Classificare $L = \{a^{n+1}b^m c^n \mid n \geq 0, m \geq 1\}$, ovvero dire se L è regolare, oppure libero ma non regolare, oppure non libero, giustificando adeguatamente la risposta.

4. È vero che, per ogni linguaggio regolare L, esiste una grammatica G di classe LR(1) tale che $L = L(G)$? Motivare la risposta.

5. Si consideri il seguente DFA $M = (\Sigma, Q, \delta, q_0, F)$, dove $\Sigma = \{a, b\}$, $Q = \{q_0, q_1, q_2, q_3, q_4, q_5\}$, $F = \{q_2, q_3\}$ e la funzione di transizione $\delta : Q \times \Sigma \to Q$ è cosí definita: $\delta(q_0, a) = q_1$, $\delta(q_0, b) = q_2$, $\delta(q_1, a) = q_0$, $\delta(q_1, b) = q_3$, $\delta(q_2, a) = q_4$, $\delta(q_2, b) = q_5$, $\delta(q_3, a) = q_4$, $\delta(q_3, b) = q_5$, $\delta(q_4, a) = q_4$, $\delta(q_4, b) = q_5$, $\delta(q_5, a) = q_4$, $\delta(q_5, b) = q_5$.

   Si fornisca una rappresentazione grafica di M. Si verifichi se M sia minimo, utilizzando l'algoritmo con tabella a scala; nel caso esistano stati equivalenti, produrre l'automa minimo M'. Si ricavi da M' la grammatica regolare associata secondo la costruzione vista a lezione. Qual è il linguaggio riconosciuto da M'?

6. Considerando il DFA M' del punto 5 ed il suo linguaggio riconosciuto L, costruire l'automa M'' che riconosce il linguaggio complementare $\overline{L} = \{w \in \{a, b\}^* \mid w \notin L\}$. Quale espressione regolare denota il linguaggio $\overline{L}$?

7. Mostrare che $L_1 = \{a^n b^m \mid 0 \leq n \leq m\}$ è libero deterministico, costruendo un opportuno DPDA. Sapendo che anche $L_2 = \{a^n b^m \mid 0 \leq m \leq n\}$ è libero deterministico, è vero che $L_1 \cup L_2$ è un linguaggio libero deterministico?

8. Si consideri la seguente grammatica G con simbolo iniziale S:
$$\begin{array}{lcllcl} S & \to & BCA \mid ABE & \quad A & \to & \mathtt{a} \mid \mathtt{a}D\mathtt{b} \mid \mathtt{b}S\mathtt{c} \\ B & \to & C \mid B\mathtt{b} & \quad C & \to & \epsilon \mid \mathtt{d}C \\ D & \to & \mathtt{d}D & \quad E & \to & D \mid \mathtt{d}E \end{array}$$
(i) Si calcolino i First e i Follow per tutti i nonterminali. (ii) Si rimuovano i simboli inutili, ottenendo una grammatica semplificata G'. (iii) Si rimuova la produzione epsilon per ottenere una grammatica G'' senza produzioni epsilon, che sia equivalente a G'. (iv) Si rimuovano le produzioni unitarie da G'' per ottenere una grammatica G''' senza produzioni unitarie equivalente a G''. (v) Si rimuova la ricorsione sinistra immediata per B per ottenere una $\overline{G}$ equivalente a G'''.

9. Data la seguente grammatica G con simbolo iniziale S:
$$\begin{array}{lcl} S & \to & \epsilon \mid A \\ A & \to & \mathtt{a}A\mathtt{c} \mid \mathtt{ac} \mid B \\ B & \to & \mathtt{b} \mid \mathtt{b}B \end{array}$$
si determini $L(G)$. Si verifichi se G è di classe LL(1). Se non lo è, si manipoli opportunamente G per trasformarla in una grammatica equivalente G' di classe LL(1). Si costruisca quindi la tabella di parsing LL(1). Si nostri il funzionamento del parser LL(1) sull'input $abbc$.

10. Data la grammatica G del punto 9, si verifichi che non è di classe LR(0), ma è di classe SLR(1). Si mostri il funzionamento del parser SLR(1) sull'input $abbc$.

Parziale A del 20/12/22

1) $I^{L_0}_{L_1}\left(C^{X}_{L_1 L_2}, C^{Y}_{XY}\right)$

Per non produrre errore, deve essere $X = L_1$ e $Y = L_1$.

Ma quello che viene calcolato è $C^{L_2}_{L_1 L_1}$ che non ha senso; sicuramente non è utile un compilatore da L_1 a L_1...

2) b_0 or b_1 con valutazione IP (interna - parallela)

$$\frac{\langle b_0, \sigma\rangle \longrightarrow_b \langle b_0', \sigma'\rangle}{\langle b_0 \text{ or } b_1, \sigma\rangle \longrightarrow_b \langle b_0' \text{ or } b_1, \sigma'\rangle}$$

$$\frac{\langle b_1, \sigma\rangle \longrightarrow_b \langle b_1', \sigma'\rangle}{\langle b_0 \text{ or } b_1, \sigma\rangle \longrightarrow_b \langle b_0 \text{ or } b_1', \sigma'\rangle}$$

$$\frac{}{\langle t_1 \text{ or } t_2, \sigma\rangle \longrightarrow_b \langle t, \sigma\rangle}$$

t	t_1	t_2
V	V	V
V	V	F
V	F	V
F	F	F

$\langle tt \text{ or } (3-5)=0, \sigma\rangle \not\longrightarrow_{IP}$

$\xrightarrow{ES} \langle tt, \sigma\rangle$

IP e ES, se terminano entrambe, danno lo stesso risultato. Tuttavia ES è più "definita" di IP, cioè può fornire un risultato quando IP non ci riesce.

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

$$G \begin{cases} S \to aA \\ A \to aAc \mid bB \\ B \to \varepsilon \mid bB \end{cases} \qquad \begin{aligned} L(S) &= a \cdot L(A) = L \\ L(A) &= \{a^n b b^* c^n \mid n \geq 0\} \\ L(B) &= b^* \end{aligned}$$

G è libera, quindi $L = L(G)$ è libero.

L non è regolare e lo dimostro usando il pumping lemma a rovescio

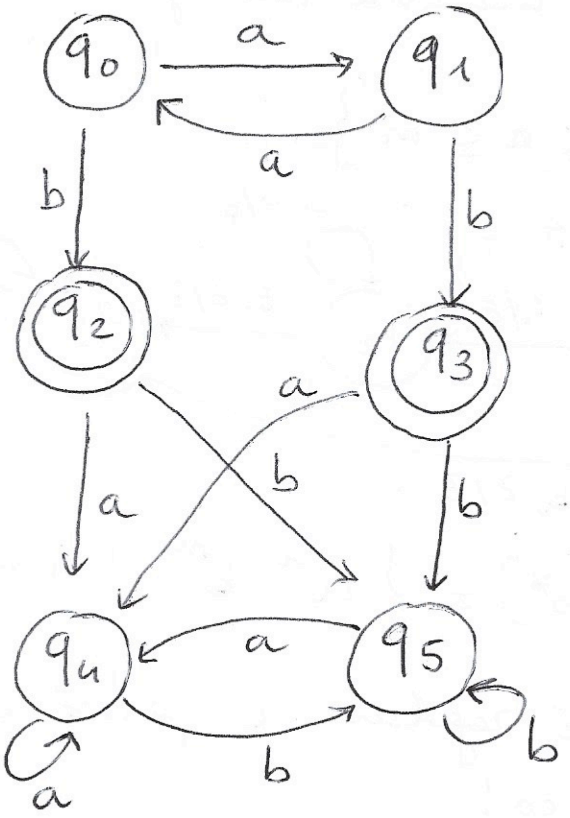
- Fissiamo $N > 0$ generico $\quad (\forall N > 0)$
- Scegliamo $z = a^{N+1} b c^N \quad (\exists z \in L, |z| \geq N)$
- Per ogni u, v, w tali che $z = uvw$, $|uv| \leq N$ e $|v| \geq 1$, deve essere che $v = a^J$ con $J \geq 1$.
- Allora $\exists k = 2$ tale che $uv^2w \notin L$

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

$\Rightarrow L$ non è regolare!

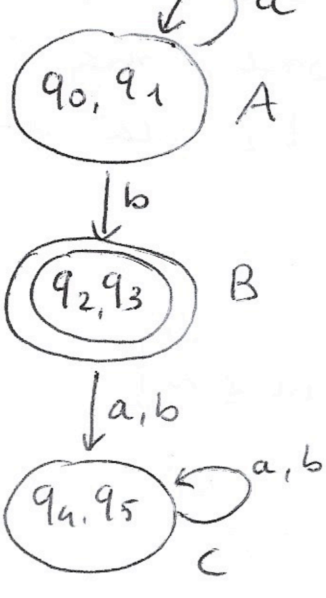
4) Se L è regolare, allora ∃ G di classe LL(1) tale che L = L(G) per un teorema visto a lezione. Poiché le grammatiche LL(1) sono incluse nelle grammatiche LR(1), ne segue che l'affermazione è vera.

5)



	0	1	2	3	4
1					
2	X_0	X_0			
3	X_0	X_0			
4	X_1	X_1	X_0	X_0	
5	X_1	X_1	X_0	X_0	

$0 \sim 1 \quad 2 \sim 3 \quad 4 \sim 5$



$A \to aA \mid bB \mid b$

$B \to aC \mid bC$

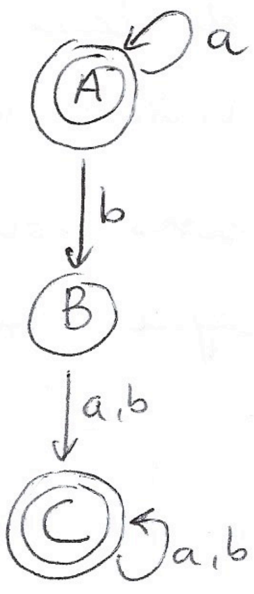
$C \to aC \mid bC$

– B e C sono inutili

$A \to aA \mid b$

$L(A) = a^* b$

6)



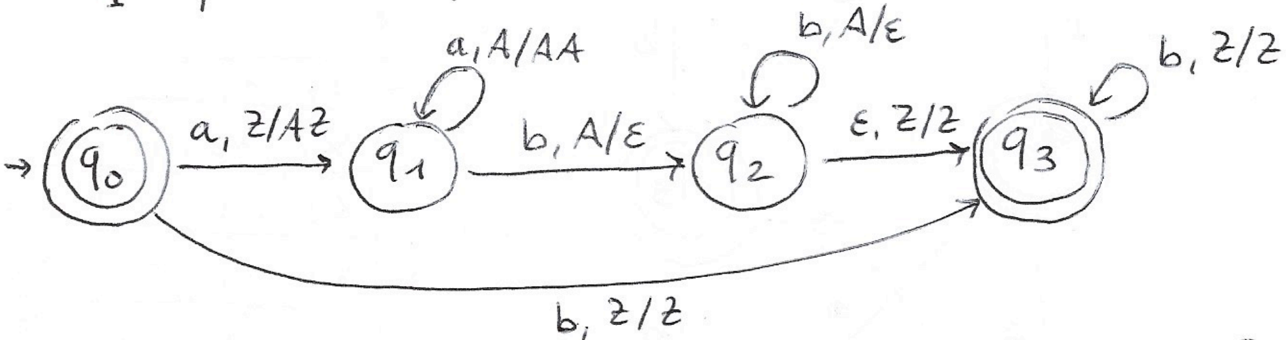
Automa $\overline{M}$ che riconosce il linguaggio $A^* \setminus a^*b = \overline{L}$ dove $A = \{a, b\}$

Una espressione regolare per $\overline{L}$ potrebbe essere

$$a^*(\varepsilon \mid b(a|b)^+)$$

ricavabile dalla grammatica per $\overline{M}$

7) $L_1 = \{ a^n b^m \mid 0 \leq n \leq m \}$



$L_1 \cup L_2 = \mathcal{L}[a^* b^*] = \{ a^n b^m \mid n, m \geq 0 \}$

Questo linguaggio è regolare e perciò è pure libero deterministico!

(Notare che i ling. liberi deterministici <u>non</u> sono chiusi per unione, ma, in questo caso, l'unione di L_1 e L_2 dà un ling. libero deterministico.)

$L_2 = \{ a^n b^m \mid 0 \leq m \leq n \}$

8) $S \to BCA \mid ABE$

$$G \begin{cases} A \to a \mid aDb \mid bSc \\ B \to C \mid Bb \\ C \to \varepsilon \mid dC \\ D \to dD \\ E \to D \mid dE \end{cases}$$

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

D e E non sono generatori ⇒ inutili

$$\left.\begin{array}{l} S \to BCA \\ A \to a \mid bSc \\ B \to C \mid Bb \\ C \to \varepsilon \mid dC \end{array}\right] G'$$

elimino le prod. epsilon

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

$$G'' \begin{cases} S \to BCA \mid CA \mid BA \mid A \\ A \to a \mid bSc \\ B \to C \mid Bb \mid b \\ C \to dC \mid d \end{cases}$$

rimuovere le prod. unitarie

$$G''' \begin{cases} S \to BCA \mid CA \mid BA \mid a \mid bSc \\ A \to a \mid bSc \\ B \to dC \mid d \mid Bb \mid b \\ C \to dC \mid d \end{cases}$$

$\{(S,A), (B,C)\} \cup Id$

coppie unitarie

rimuovere la ricorsione sx $B \to Bb$

$S \to \dots$

$A \to \dots$

$B \to dCA' \mid dA' \mid bA'$

$A' \to bA' \mid \varepsilon$

$C \to \dots$

10)

$$G: \begin{cases} S \to \varepsilon \mid A \\ A \to aAc \mid ac \mid B \\ B \to b \mid bB \end{cases}$$

$L(S) = \{\varepsilon\} \cup L(A)$

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

$L(B) = b^+$

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

G non è di classe LL(1) < $B \to b \mid bB$; $A \to aAc \mid ac$

Fattorizziamo G:

$$G': \begin{cases} S \to \varepsilon \mid A \\ A \to aA' \mid B \\ A' \to Ac \mid c \\ B \to bB' \\ B' \to \varepsilon \mid B \end{cases}$$

G' è LL(1) perché:

- $First(\varepsilon) \cap \underbrace{First(A)}_{\{a,b\}} = \emptyset$
- $\underbrace{Follow(S)}_{\{\$\}} \cap First(A) = \emptyset$
- $\underbrace{First(aA')}_{\{a\}} \cap \underbrace{First(B)}_{\{b\}} = \emptyset$
- $\underbrace{First(Ac)}_{\{a,b\}} \cap \underbrace{First(c)}_{\{c\}} = \emptyset$
- $First(\varepsilon) \cap First(B) = \emptyset$
- $\underbrace{Follow(B')}_{\{\$,c\}} \cap \underbrace{First(B)}_{\{b\}} = \emptyset$

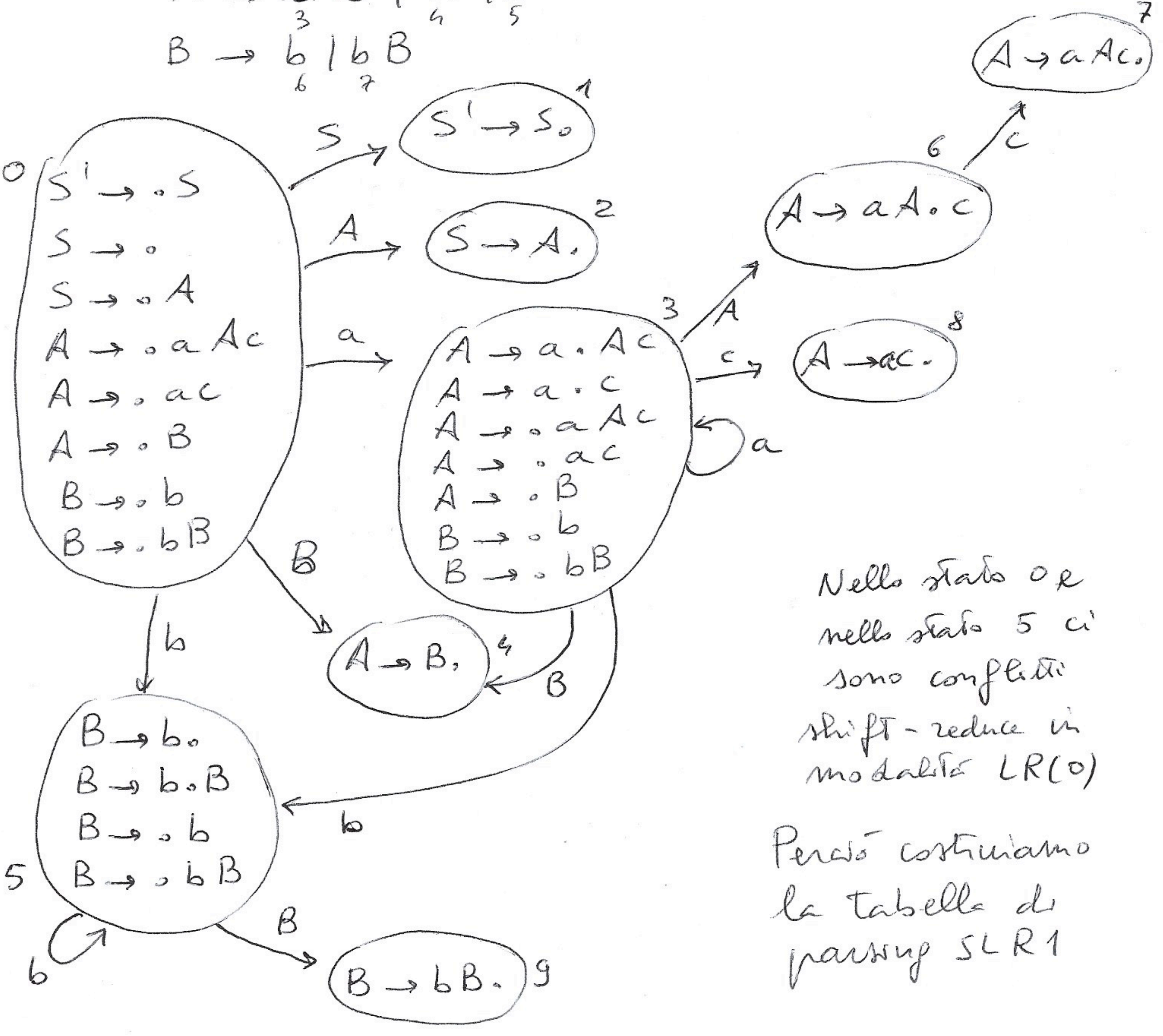
	a	b	c	\$
S	$S \to A$	$S \to A$		$S \to \varepsilon$
A	$A \to aA'$	$A \to B$		
A'	$A' \to Ac$	$A' \to Ac$	$A' \to c$	
B		$B \to bB'$		
B'		$B' \to B$	$B' \to \varepsilon$	$B' \to \varepsilon$

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

abbc\$	S
	A
	aA'
bbc\$	A'
	Ac
	Bc
	bB'c
bc\$	B'c
	Bc
	bB'c
c\$	B'c
	c
\$	ε
	ok

10)

$$S \to \overset{1}{\varepsilon} \mid \overset{2}{A}$$
$$A \to \underset{3}{aAc} \mid \underset{4}{ac} \mid \underset{5}{B}$$
$$B \to \underset{6}{b} \mid \underset{7}{bB}$$



Nello stato 0 e nello stato 5 ci sono conflitti shift-reduce in modalità LR(0)

Perciò costruiamo la tabella di parsing SLR1

	Follow
S	\$
A	\$, c
B	\$, c

	a	b	c	$	S	A	B
0	S3	S5		R1	G1	G2	G4
1				ACC			
2				R2			
3	S3	S5	S8			G6	G4
4			R5	R5			
5		S5	R6	R6			G9
6			S7				
7			R3	R3			
8			R4	R4			
9			R7	R7			

$(0,\ \varepsilon,\ abbc\$)$

$(03,\ a,\ bbc\$)$

$(035,\ ab,\ bc\$)$

$(0355,\ ab\underbrace{b}_{B},\ c\$)$

$(0359,\ a\underbrace{bB}_{B},\ c\$)$

$(034,\ a\underbrace{B}_{A},\ c\$)$

$(036,\ aA,\ c\$)$

$(0367,\ \underbrace{aAc}_{A},\ \$)$

$(02,\ \underbrace{A}_{S},\ \$)$

$(01,\ S,\ \$)$

ACCEPT!