Corso di Linguaggi di Programmazione — Parziale M-Z di fine modulo
Prova scritta del 20 Dicembre 2021

Tempo a disposizione: 2 ore e 30 minuti.

1. Per quali valori delle variabili X e Y la seguente espressione

$$\mathcal{I}_X^{L_0}(\mathcal{C}_{Y,L_1}^{L_1}, \mathcal{I}_{L_1}^{L_0})$$

   ha senso? Se si, calcola qualcosa di utile?

2. Descrivere le regole di semantica operazionale strutturata per il comando **for** i:=1 **to** n **do** c. (Si suppone che la variabile di controllo i ed anche la variabile n non vengano mai modificate durante l'esecuzione del corpo c. *Suggerimento*: Ricondursi al caso del **while**.)

3. Fornire una definizione regolare per *password*, che deve essere una qualunque sequenza di lettere e/o cifre che deve terminare con una cifra e deve contenere almeno una lettera minuscola ed almeno una lettera maiuscola (in qualsiasi ordine).

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

5. Si consideri l'espressione regolare $a^*(a|\emptyset)^*b$. 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 verifichi se è minimo; se non lo fosse, lo si minimizzi per ottenere un DFA M''; poi si ricavi da M'' la grammatica regolare associata, seguendo la costruzione vista a lezione; quindi si semplifichi la grammatica ottenuta, eliminando i simboli inutili; infine, si ricavi dalla grammatica l'espressione regolare associata.

7. Dati i linguaggi L_1 ed L_2, il primo libero, ma non regolare, e il secondo regolare, a quale classe appartiene il linguaggio $L_1 \cup L_2 = \{w \mid w \in L_1 \vee w \in L_2\}$? Può $L_1 \cup L_2$ essere finito?

8. È vero che, per ogni linguaggio finito L, esiste un DPDA N tale che $L = L[N]$? Motivare la risposta.

9. Mostrare che $L = \{a^nb^n \mid n \geq 2\}$ è libero, costruendo un semplice parser Top-down ndeterministico (come PDA con un solo stato che riconosca L per pila vuota).

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

$$\begin{array}{lll} S & \to & AB \mid C \mid \mathtt{a}E \\ B & \to & \mathtt{a} \mid \mathtt{b}BC \\ D & \to & \mathtt{c} \mid \mathtt{d}S \end{array} \qquad \begin{array}{lll} A & \to & \epsilon \mid \mathtt{a}SB \\ C & \to & A \mid C\mathtt{d} \\ E & \to & \mathtt{a}DE \end{array}$$

    (i) Si calcolino i First e i Follow per tutti i nonterminali. (ii) Si rimuovano i simboli inutili per ottenere una grammatica G' senza simboli inutili, che sia equivalente a G. (iii) Si rimuova la produzione epsilon per ottenere una grammatica G'' senza produzioni epsilon, che sia equivalente a G' (a meno di ϵ). (iv) Si rimuovano le produzioni unitarie da G'' per ottenere una grammatica G''' senza produzioni unitarie equivalente a G''.

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

$$\begin{array}{lll} S & \to & AB \\ A & \to & \epsilon \mid \mathtt{a}A\mathtt{b} \\ B & \to & \epsilon \mid \mathtt{c}B\mathtt{d} \end{array}$$

    (i) Determinare il linguaggio generato $L(G)$. (ii) Verificare che G è di classe LL(1). (iii) Costruire la tabella di parsing LL(1) per G. (iv) Mostrare il funzionamento del parser LL(1) su input $abcc$.

12. 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 che non ci sono conflitti. (iii) Mostrare il funzionamento del parser SLR(1) per l'input $abcc$.

$$I_X^{L_0}\left(C^{L_1}_{Y,L_1}, I^{L_0}_{L_1}\right)$$

È necessario che:
- $X = L_1$
- $Y = L_0$

Viene calcolato $I^{L_1}_{L_1}$ che non ha senso, perché un interprete per L_1 scritto in L_1 non serve a niente.

---

2)

$\langle \text{for } i := 1 \text{ to } n \text{ do } c, \sigma\rangle \longrightarrow \langle i := 1; \text{ while } i \le n \text{ do } (c; i := i+1), \sigma\rangle$

---

3)

password := tutto* Minus tutto* Maius tutto* cifra
| tutto* Maius tutto* Minus tutto* cifra

tutto := Minus | Maius | Cifra

Minus := [a-z]
Maius := [A-Z]
Cifra := [0-9]

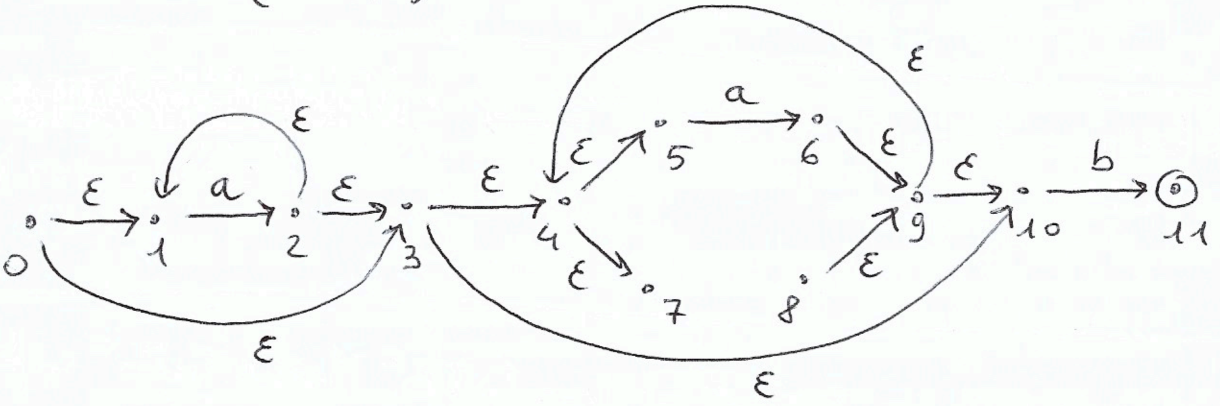
4) $L = \{ b^{m+k} a^n \mid n, k \geq 0, m \geq 1 \}$

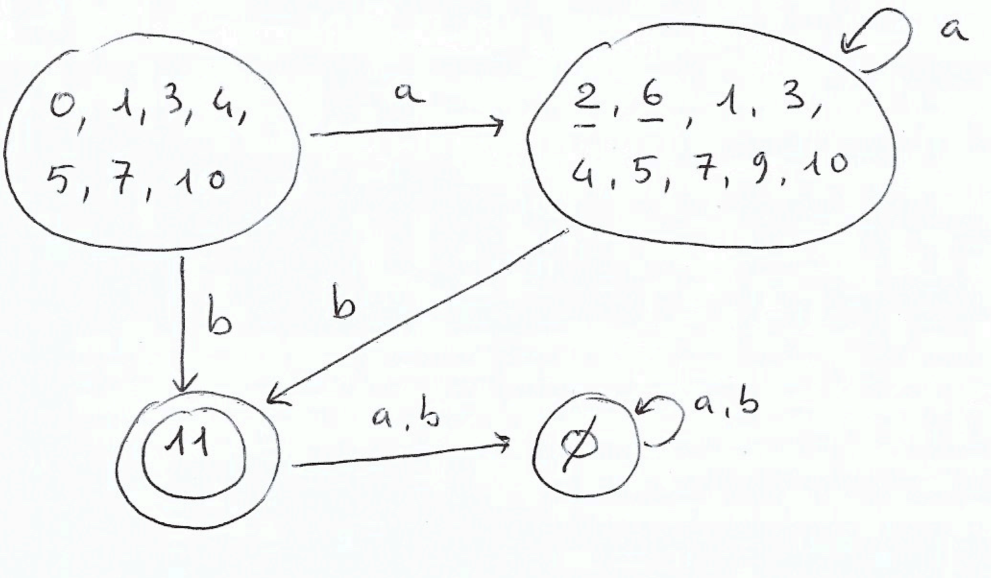
L è regolare perché è descrivibile dalla espressione regolare $b^+ a^*$

$$\left.\begin{array}{l} S \to BA \\ B \to b \mid bB \\ A \to \varepsilon \mid aA \end{array}\right] \text{gr. libera} \qquad \text{gr. regolare} \left[\begin{array}{l} S \to bB \\ B \to \varepsilon \mid bB \mid aA \\ A \to \varepsilon \mid aA \end{array}\right.$$

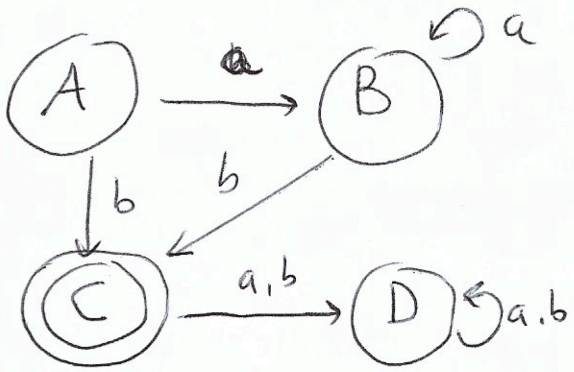
---

5) $a^* (a \mid \emptyset)^* b$



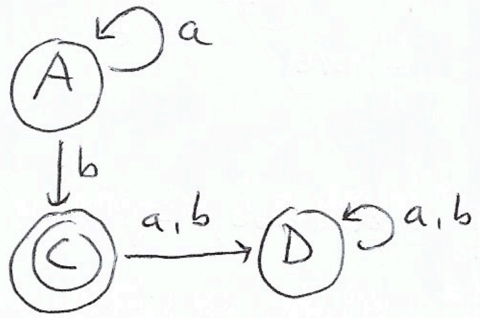


6)



	A	B	C
B			
C	X_0	X_0	
D	X_1	X_1	X_0

$A \sim B$



$A \to aA \mid \cancel{bC} \mid b$

$\cancel{C \to aD \mid bD}$

$\cancel{D \to aD \mid bD}$

$A \to aA \mid b \rightsquigarrow a^* b$

---

7) L_1 libero ma non regolare

L_2 regolare

$L_1 \cup L_2$ appartiene alla classe dei liberi, perché L_2 è pure libero, e i ling. liberi sono chiusi per unione.

$L_1 \cup L_2$ non può essere finito, perché L_1 non è regolare e quindi, poiché tutti i ling. finiti sono regolari, L_1 non è finito.

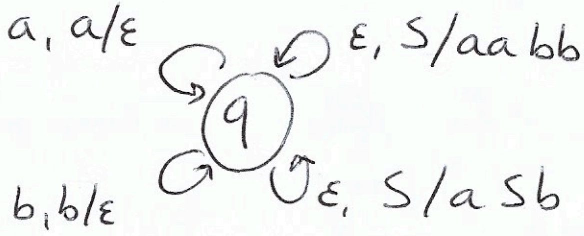
---

8) Se L è finito, allora L è regolare, allora $\exists$ DFA M tale che $L = L[M]$.

Il DFA M può essere usato per costruire un DPDA N che si comporta esattamente come M, senza mai modificare la pila. Allora N è tale che $L = L[N]$.

9) $L = \{ a^n b^n \mid n \geq 2 \}$

$$S \to aabb \mid aSb$$



$(q, aaabbb, S) \vdash (q, aaabbb, aSb) \vdash (q, aabbb, Sb)$
$\vdash (q, aabbb, aabbb) \vdash (q, abbb, abbb) \vdash (q, bbb, bbb) \vdash \dots$
$\dots \; (q, b, b) \vdash (q, \varepsilon, \varepsilon)$

---

10)

$S \to AB \mid C \mid aE$

$A \to \varepsilon \mid aSB$

$B \to a \mid bBC$

$C \to A \mid Cd$

$D \to c \mid dS$

$E \to aDE$

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

E non è un generatore e, se tolgo E, D non è più raggiungibile

$$\left.\begin{aligned} S &\to AB \mid C \\ A &\to \varepsilon \mid aSB \\ B &\to a \mid bBC \\ C &\to A \mid Cd \end{aligned}\right] \quad G' \text{ senza simboli inutili}$$

$N(G) = \{A, C, S\}$ simboli annullabili

$$\left.\begin{array}{l} S \to AB \mid B \mid C \\ A \to aSB \mid aB \\ B \to a \mid bBC \mid bB \\ C \to A \mid Cd \mid d \end{array}\right] G''$$

G'' senza prod. ε

N.B. $\varepsilon \notin L(G'')$
mentre $\varepsilon \in L(G')$

Le coppie unitarie sono (S,B), (S,C), (C,A), (S,A)

$$G''' \left[\begin{array}{l} S \to AB \mid \underbrace{a \mid bBC \mid bB}_{B} \mid \underbrace{Cd \mid d}_{C} \mid \underbrace{aSB \mid aB}_{A} \\ A \to aSB \mid aB \\ B \to a \mid bBC \mid bB \\ C \to \underbrace{aSB \mid aB}_{A} \mid Cd \mid d \end{array}\right.$$

G''' è senza prod. unitarie ed è equivalente a G''.

:)

$$G \begin{cases} S \to AB \\ A \to \varepsilon \mid aAb \\ B \to \varepsilon \mid cBd \end{cases}$$

$L(A) = \{ a^n b^n \mid n \geq 0 \}$
$L(B) = \{ c^m d^m \mid m \geq 0 \}$

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

G è di classe LL(1) perché

- $\text{First}(\varepsilon) \cap \text{First}(aAb) = \emptyset$
  - $\text{Follow}(A) \cap \text{First}(aAb) = \emptyset$
    $\{b, c, \$\} \cap \{a\} = \emptyset$
- $\text{First}(\varepsilon) \cap \text{First}(cBd) = \emptyset$
  - $\text{Follow}(B) \cap \text{First}(cBd) = \emptyset$
    $\{d, \$\} \cap \{c\} = \emptyset$

	a	b	c	d	\$
S	$S \to AB$		$S \to AB$		$S \to AB$
A	$A \to aAb$	$A \to \varepsilon$	$A \to \varepsilon$		$A \to \varepsilon$
B			$B \to cBd$	$B \to \varepsilon$	$B \to \varepsilon$

$\text{First}(AB) = \{a, c, \varepsilon\}$
$\text{Follow}(S) = \{\$\}$

input	stack
<u>a</u>bcc\$	S\$
	AB\$
	<u>a</u>AbB\$
<u>b</u>cc\$	AbB\$
	<u>b</u>B\$
<u>c</u>c\$	B\$
	<u>c</u>Bd\$
<u>c</u>\$	Bd\$
	<u>c</u>Bdd\$
\$	Bdd\$
	dd\$ errore

ho finito l'input ma lo stack non è vuoto
"no-match" fra "d" e "\$"

12) $S \to AB_{(1)}$

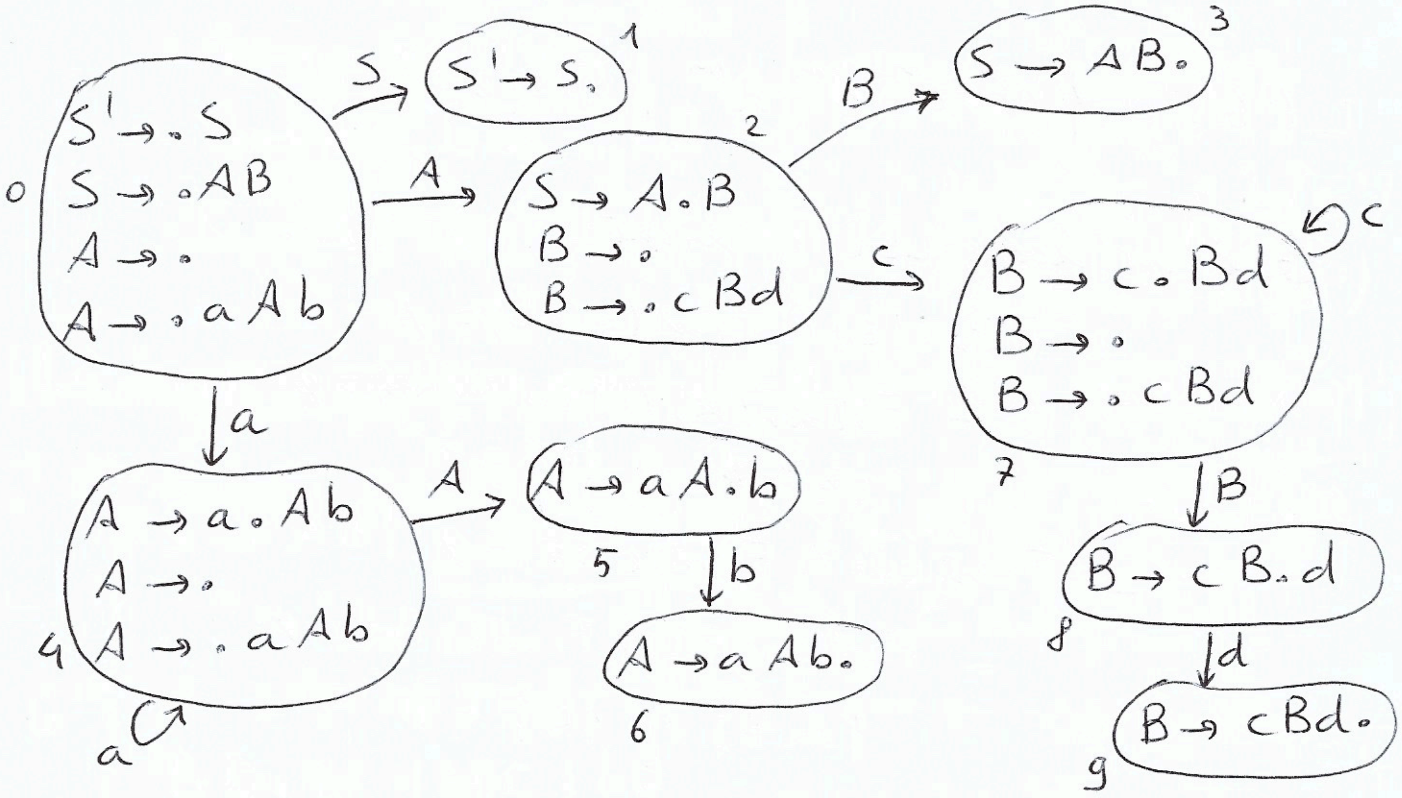
$A \to \underset{(2)}{\varepsilon} \mid aAb_{(3)}$

$B \to \underset{(4)}{\varepsilon} \mid \underset{(5)}{cBd}$

Follow(A) = {b, c, $}

Follow(B) = {d, $}

Follow(S) = {$}



	a	b	c	d	$	S	A	B
0	S4	R2	R2		R2	G1	G2	
1					ACC			
2			S7	R4	R4			G3
3					R1			
4	S4	R2	R2		R2		G5	
5		S6						
6		R3	R3		R3			
7			S7	R4	R4			G8
8				S9				
9				R5	R5			

(0, ε, abcc$)

(04, $\underbrace{a}_{\to A}$, bcc$)

(045, aA, bcc$)

(0456, $\underbrace{aAb}_{A}$, cc$)

(02, A, cc$)

(027, Ac, c$)

(0277, $Acc_{\sqcup \to B}$, $)

(02778 AccB, $)

M[8, $] = "bianco"

errore