

**Fondamenti di Programmazione - 08/02/2013**  
**2° Parziale e Esame Finale (1° appello - 1° semestre A.A. 12/13)**

Cognome: .....	Nome: .....	Matricola: .....
<input type="checkbox"/> TOTALE:	<b>Svolgere Es.1, Es.2, Es.3</b>	Tempo: 2 ore
<input type="checkbox"/> II PARZIALE:	<b>Es.2, Es.3</b>	Tempo: 1 ora e 30 min.
Anno di immatricolazione:		

**Es. 1)** Sia  $\Lambda = \{\text{Questo, È, Un, Esercizio, Già, Fatto, In, Classe, Ma, Prestate, Attenzione}\}$  e siano

$\mathcal{L}_1 = \{\text{Questo}^n (\text{È Un})^k \text{Esercizio}^n \mid k \geq 3, n \geq 2\}$ ,  $\mathcal{L}_2 = \{\text{Già}^n \text{Fatto}^{n+m} (\text{In Classe})^m \mid n \geq 2, m \geq 1\}$  e

$\mathcal{L}_3 = \{\text{Ma}^n \text{Prestate Attenzione}^m \mid n \geq 0, m \geq 0\}$  linguaggi su  $\Lambda$ .

**a)** Definire una grammatica che genera il linguaggio:

$$\mathcal{L}_{123} = \{(sss)^n \mid n \geq 0, s \in \mathcal{L}_1 \text{ oppure } s \in \mathcal{L}_2 \text{ oppure } s \in \mathcal{L}_3\}.$$

Per esempio, la stringa `QuestoQuestoÈUnÈUnÈUnEsercizioEsercizioGiàGiàGiàFattoFattoFattoFattoFattoInClasseInClasseMaMaPrestateAttenzione` appartiene a  $\mathcal{L}_{123}$ .

**b)** La stringa `QuestoQuestoÈUnEsercizioEsercizioGiàFattoInClasseInClasseInClasseMa` appartiene a  $\mathcal{L}_{123}$ ? Se no, motivare la risposta. Se sì, mostrare l'albero di derivazione.

**c) Facoltativo** - Se la grammatica definita è ambigua, dimostrarlo.

**SOLUZIONE:**

**a)**

$S ::= S' S' S' S' \mid \varepsilon$   
 $S' ::= S_1 \mid S_2 \mid S_3$   
 $S_1 ::= \text{Questo } S_1 \text{ Esercizio} \mid \text{QuestoQuesto } S'_1 \text{ EsercizioEsercizio}$   
 $S'_1 ::= S'_1 \text{ È Un} \mid \text{È Un È Un È Un}$   
 $S_2 ::= S'_2 S''_2$   
 $S'_2 ::= \text{Già } S'_2 \text{ Fatto} \mid \text{GiàGiàFattoFatto}$   
 $S''_2 ::= \text{Fatto } S''_2 \text{ InClasse} \mid \text{FattoInClasse}$   
 $S_3 ::= S'_3 \text{ Prestate } S''_3$   
 $S'_3 ::= \text{Ma } S'_3 \mid \varepsilon$   
 $S''_3 ::= \text{Attenzione } S''_3 \mid \varepsilon$

**b)** La stringa `QuestoQuestoÈUnEsercizioEsercizioGiàFattoInClasseInClasseInClasse` NON appartiene a  $\mathcal{L}_{123}$  in quanto devono esserci almeno 2 “Già” e almeno 2 “Fatto”. Il numero di “Fatto” non è corretto rispetto al numero di “InClasse”. Inoltre, ...

**c)** La grammatica definita nel punto **a)** non è ambigua. Infatti, ... (ragionare sulla struttura degli alberi di derivazione in funzione delle produzioni definite).

**Es. 2)** Dire se i seguenti comandi COM1 e COM2 sono equivalenti.

COM1: if E then C<sub>1</sub>;C<sub>3</sub>;C<sub>4</sub> else C<sub>2</sub>;C<sub>3</sub> fi

COM2: if E then C<sub>1</sub> else C<sub>2</sub> fi; C<sub>3</sub>;C<sub>4</sub>

### SOLUZIONE:

**N.B.:** NON si fanno assunzioni sullo stato iniziale. Si consideri quindi un generico stato  $\sigma$ . La dimostrazione avviene poi ragionando per casi: il caso in cui  $\varepsilon[[E]]\sigma = ff$  e il caso in cui  $\varepsilon[[E]]\sigma = tt$ .

**CASO I:** Consideriamo ora il caso in cui  $\varepsilon[[E]]\sigma = ff$

Per il comando COM1 abbiamo:

$$\frac{\varepsilon[[E]]\sigma = ff \quad \frac{\langle C_2, \sigma \rangle \longrightarrow_{com} \sigma' \quad \langle C_3, \sigma' \rangle \longrightarrow_{com} \sigma''}{\langle C_2; C_3, \sigma \rangle \longrightarrow_{com} \sigma''} \quad (com_1;)}{\langle \text{if E then } C_1; C_3; C_4 \text{ else } C_2; C_3 \text{ fi}, \sigma \rangle \longrightarrow_{com} \sigma''} \quad (com_{if-ff})$$

Per il comando COM2 abbiamo:

$$\frac{\frac{\varepsilon[[E]]\sigma = ff \quad \langle C_2, \sigma \rangle \longrightarrow_{com} \sigma'}{\text{if E then } C_1 \text{ else } C_2 \text{ fi}, \sigma \rangle \longrightarrow_{com} \sigma'} \quad (com_{if-ff}) \quad \langle C_3, \sigma' \rangle \longrightarrow_{com} \sigma''}{\text{if E then } C_1 \text{ else } C_2 \text{ fi}; C_3, \sigma \rangle \longrightarrow_{com} \sigma''} \quad (com_1;)}{\langle \text{if E then } C_1 \text{ else } C_2 \text{ fi}; C_3; C_4, \sigma \rangle \longrightarrow_{com} \sigma'''} \quad (com_1;)$$

Non si è fatta alcuna ipotesi sullo stato iniziale  $\sigma$  ed è bastato verificare che, nel **CASO I**, si sono ottenuti due stati finali differenti. I comandi NON sono pertanto equivalenti.

**NOTA:** Si osservi che si è scelta un'associatività sinistra e quindi, e.g., per il comando COM2, si è scelto di valutare prima `if E then C1 else C2 fi; C3`; e poi C<sub>4</sub>. Sebbene non comunemente usata dai linguaggi di programmazione, ai fini dello svolgimento dell'esercizio, poteva scegliersi un'associatività destra.

**Es. 3)** Dato il seguente programma +/-Java, si calcolino:

- lo stato risultante dopo l'esecuzione delle righe 1-33;
- lo stato risultante dopo l'esecuzione delle righe 1-40;
- lo stato risultante dopo l'esecuzione di tutto il programma;
- fare la derivazione di valutazione dello statement di riga 42 nello stato calcolato al punto b).

```

1  class Producer {
2      private int resource;
3
4      public int getRes() {
5          return this.resource;
6      }
7
8      public void setRes(int r) {
9          this.resource = r;
10     }
11 }
12
13 class Consumer {
14     private int money;
15     private int consRes;
16     private Producer p;

```

```

17
18 public Consumer(Producer prod) {
19     this.p = prod;
20     this.consRes = prod.getRes();
21     this.money = 10;
22 }
23
24 public int getMoney() {
25     return this.money;
26 }
27
28 public void halveRes() {
29     int rate = 2;
30     this.consRes = this.consRes / rate;
31     this.money = this.money - 5;
32 }
33 }
34
35 public class Program {
36
37     public static void main(String[] cLine) {
38         int j = 0;
39         Consumer cc;
40         Producer pp = new Producer();
41
42         pp.setRes(16);
43         cc = new Consumer(pp);
44
45
46         while (cc.getMoney() > 0) {
47             cc.halveRes();
48         }
49     }
50 }

```

Listing 1: Programma +/- Java

### SOLUZIONE:

- Applicazione della regola ...
- Applicazione della regola ...

a) Lo stato alla linea 33 è  $\sigma_{33} = \langle \rho_0, \mu_0, \delta_1 \rangle$

$$\delta_1 :$$

Producer	$\langle \varphi_v(Producer), \varphi_m(Producer) \rangle$
Consumer	$\langle \varphi_v(Consumer), \varphi_m(Consumer) \rangle$

$$\varphi_v(Producer) :$$

resource	0
----------	---

$$\varphi_m(Producer) :$$

getRes	$\langle \text{int}, \perp, \{\text{return this.resource;}\} \rangle$
setRes	$\langle \text{void}, r, \{\text{this.resource} = r;\} \rangle$

$\varphi_v(Consumer) :$	money	0
	consRes	0
	p	<i>null</i>

$\varphi_m(Consumer) :$	<init>	$\langle$ void, prod, $\begin{array}{l} \{ \text{this.p} = \text{prod}; \\ \text{this.consRes} = \text{prod.getRes()}; \\ \text{this.money} = 10; \} \end{array}$ $\rangle$	
	getMoney	$\langle$ int, $\perp$ ,	$\{ \text{return this.money}; \}$ $\rangle$
	halveRes	$\langle$ void, $\perp$ , $\begin{array}{l} \{ \text{int rate} = 2; \\ \text{this.consRes} = \text{this.consRes}/\text{rate}; \\ \text{this.money} = \text{this.money} - 5; \} \end{array}$ $\rangle$	

- Binding parametro main alla linea 37.
- Applicazione della regola ...

b) Lo stato alla linea 40 è  $\sigma_{40} = \langle \rho_1, \mu_1, \delta_1 \rangle$

$\rho_1 :$	cLine	<i>null</i>
	j	<b>0</b>
	cc	<i>null</i>
	pp	$l_1$

$\mu_1 :$	$l_1$	$\langle$	$\varphi_{v(l_1)} :$	<table><tr><td>resource</td><td>0</td></tr></table>	resource	0	$, \varphi_{m(Producer)} \rangle$
resource	0						

- Applicazione della regola ...
- Applicazione della regola ...

c) Lo stato alla fine del programma è  $\sigma_{end} = \langle \rho_2, \mu_2, \delta_1 \rangle$

$\rho_2 :$	<table><tr><td>cLine</td><td><i>null</i></td></tr><tr><td>j</td><td><b>0</b></td></tr><tr><td>cc</td><td><math>l_2</math></td></tr><tr><td>pp</td><td><math>l_1</math></td></tr></table>	cLine	<i>null</i>	j	<b>0</b>	cc	$l_2$	pp	$l_1$
cLine	<i>null</i>								
j	<b>0</b>								
cc	$l_2$								
pp	$l_1$								

$\mu_2 :$	<table><tr><td><math>l_1</math></td><td><math>\langle \varphi_{v(l_1)} : </math><table><tr><td>resource</td><td>16</td></tr></table><math>, \varphi_m(Producer) \rangle</math></td></tr><tr><td><math>l_2</math></td><td><math>\langle \varphi_{v(l_2)} : </math><table><tr><td>money</td><td>0</td></tr><tr><td>consRes</td><td>4</td></tr><tr><td>p</td><td><math>l_1</math></td></tr></table><math>, \varphi_m(Consumer) \rangle</math></td></tr></table>	$l_1$	$\langle \varphi_{v(l_1)} : $ <table><tr><td>resource</td><td>16</td></tr></table> $, \varphi_m(Producer) \rangle$	resource	16	$l_2$	$\langle \varphi_{v(l_2)} : $ <table><tr><td>money</td><td>0</td></tr><tr><td>consRes</td><td>4</td></tr><tr><td>p</td><td><math>l_1</math></td></tr></table> $, \varphi_m(Consumer) \rangle$	money	0	consRes	4	p	$l_1$
$l_1$	$\langle \varphi_{v(l_1)} : $ <table><tr><td>resource</td><td>16</td></tr></table> $, \varphi_m(Producer) \rangle$	resource	16										
resource	16												
$l_2$	$\langle \varphi_{v(l_2)} : $ <table><tr><td>money</td><td>0</td></tr><tr><td>consRes</td><td>4</td></tr><tr><td>p</td><td><math>l_1</math></td></tr></table> $, \varphi_m(Consumer) \rangle$	money	0	consRes	4	p	$l_1$						
money	0												
consRes	4												
p	$l_1$												

c) Derivazione di valutazione della regola  $com_{mcallpar}$  applicata alla linea 42:

$$\begin{array}{c}
\sigma_{40} = \langle \rho_1, \mu_1, \delta_1 \rangle \quad \rho_1(pp) = l_1 \quad \mu_1(l_1) = \langle \varphi_{v(l_1)}, \varphi_m(Producer) \rangle \\
\varphi_m(Producer)(\text{setRes}) = \langle \text{void}, r, \text{this.resource} = r; \rangle \\
\langle 16, \sigma_{40} \rangle \longrightarrow_{exp} \langle \underline{16}, \sigma_{40} \rangle \\
\sigma_{local} = \langle \omega[{}^{l_1}/\text{this}], \mu_1, \delta_1 \rangle \quad \rho_{local} = \omega[{}^{l_1}/\text{this}] \\
\langle \text{this.resource} = r; , \sigma_{local} \rangle \longrightarrow_{com} \sigma'_{local} \quad \sigma'_{local} = \langle \rho_{local}, \mu'_1, \delta_1 \rangle \\
\sigma'_{42} = \langle \rho_1, \mu'_1, \delta_1 \rangle \\
\hline
\langle pp.\text{setRes}(16), \sigma_{40} \rangle \longrightarrow_{com} \sigma'_{42} \quad (com_{mcallpar})
\end{array}$$

Quindi, dopo l'esecuzione dello statement alla linea 42 lo stato è  $\sigma'_{42} = \langle \rho_1, \mu'_1, \delta_1 \rangle$

$\rho_1 :$	<table><tr><td>cLine</td><td><i>null</i></td></tr><tr><td>j</td><td><b>0</b></td></tr><tr><td>cc</td><td><i>null</i></td></tr><tr><td>pp</td><td><math>l_1</math></td></tr></table>	cLine	<i>null</i>	j	<b>0</b>	cc	<i>null</i>	pp	$l_1$
cLine	<i>null</i>								
j	<b>0</b>								
cc	<i>null</i>								
pp	$l_1$								

$\mu'_1 :$	<table><tr><td><math>l_1</math></td><td><math>\langle \varphi_{v(l_1)} : </math><table><tr><td>resource</td><td>16</td></tr></table><math>, \varphi_m(Producer) \rangle</math></td></tr></table>	$l_1$	$\langle \varphi_{v(l_1)} : $ <table><tr><td>resource</td><td>16</td></tr></table> $, \varphi_m(Producer) \rangle$	resource	16
$l_1$	$\langle \varphi_{v(l_1)} : $ <table><tr><td>resource</td><td>16</td></tr></table> $, \varphi_m(Producer) \rangle$	resource	16		
resource	16				