

# Linguaggi per il Web – appello del 30/6/2010

COGNOME: .....

NOME: .....

MATRICOLA: .....

Autorizzo la pubblicazione del mio voto di questo esame sul sito web <http://www.dis.uniroma1.it/~rosati/lw>, secondo quanto prevede il decreto legislativo 196/2003 (codice in materia di protezione dei dati personali) che dichiaro di conoscere. In fede,

## Esercizio 1

(a) Data la seguente grammatica  $G$ :

$$\begin{aligned} S &\rightarrow xA \mid yAB \mid zB \\ A &\rightarrow yz \mid zA \\ B &\rightarrow xAB \mid yB \mid zx \end{aligned}$$

è possibile stabilire se  $G$  è una grammatica LL(1) senza costruire esplicitamente gli insiemi FIRST e FOLLOW e la tabella di parsing di  $G$ ? motivare la risposta;

(b) Data la seguente grammatica  $G$ :

$$\begin{aligned} S &\rightarrow ASA \mid SSA \mid SbS \\ A &\rightarrow Acc \mid AbA \mid aAB \mid a \\ B &\rightarrow ABc \mid Bx \mid Bxb \mid b \end{aligned}$$

scrivere una grammatica  $G'$  tale che  $G'$  non presenti né ricorsione sinistra diretta e  $\mathcal{L}(G') = \mathcal{L}(G)$ .

**Esercizio 2** Si consideri il frammento del linguaggio Java costituito dalle stringhe che corrispondono alla definizione di un metodo Java di tipo `int` con 0, 1 o 2 argomenti di tipo `int`. Il corpo del metodo contiene una sequenza qualsiasi di istruzioni di assegnazione, il cui lato destro è o una costante intera oppure una variabile oppure una chiamata ad un metodo a 0, 1 o 2 argomenti, in cui ogni argomento è o una costante intera o un nome di variabile. Come ultima istruzione, il corpo del metodo presenta una istruzione `return` il cui argomento è o una variabile o una costante intera.

Esempi di stringhe appartenenti a questo linguaggio sono i seguenti:

```
int pluto() {                               int paperino3(int a, int b) {
    a=0;                                       nuovaVariabile=1000;
    b=54;                                       variabile45=xyz;
    variabile=pippo(34);                       variabileOld=pluto(x,42);
    return 300;                                x2=paperino(abc,defg);
}                                               return x2;
}
```

Scrivere una grammatica non contestuale per tale linguaggio, dividendo la specifica del lessico del linguaggio (che va definita mediante espressioni regolari) dalla specifica della sintassi vera e propria. Scrivere preferibilmente tale specifica come specifica JavaCC.

**Esercizio 3** Data la seguente DTD:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE a [
  <!ELEMENT a ((b,c)+ | (d,e)? )>
  <!ELEMENT b (#PCDATA | b | c )*>
  <!ELEMENT c (#PCDATA | d )*>
  <!ELEMENT d ((e,f)*|a)>
  <!ELEMENT e ANY>
  <!ELEMENT f EMPTY>
  <!ATTLIST a param (A | B | C | D) "A">
  <!ATTLIST b attr CDATA #REQUIRED>
  <!ATTLIST c attr CDATA #IMPLIED>
]>
```

1. dire se la DTD è corretta ed in caso negativo evidenziare gli errori presenti;
2. scrivere un documento XML che sia valido rispetto alla DTD e che contenga tutti gli elementi dichiarati nella DTD.

**Esercizio 4** Con riferimento al documento XML dell'esercizio precedente:

1. scrivere l'albero DOM corrispondente a tale documento;
2. scrivere una sequenza di chiamate a metodi della API DOM che genera l'albero DOM corrispondente al documento.

**Esercizio 5** Scrivere un foglio di stile XSL che, dato un documento XML, restituisce il documento tale che: 1) l'elemento radice è uguale all'elemento radice del documento di input (il nome dell'elemento radice non è noto a priori); 2) tutti gli elementi figli dell'elemento radice vengono copiati, e il loro contenuto viene ricorsivamente trasformato; 3) tutti gli elementi *x* che non sono figli dell'elemento radice vengono copiati, ma il loro contenuto viene eliminato; 4) tutti gli elementi *y* vengono restituiti in uscita e il loro contenuto viene ricorsivamente trasformato; 5) tutti gli altri elementi vengono tramutati in elementi *z* e il loro contenuto viene eliminato. Ad esempio, se il documento XML di input è il seguente:

```
<r>
  <x>testo 0</x>
  <y>
    <x>testo 1</x>
    <w>testo 2</w>
  </y>
  <x>
    <x/>
    <y>testo 3
      <x>testo 4</x>
    </y>
    <x>testo 5
      <x/>
    </x>
  </x>
  <t>
    <y>testo 6</y>
    <t>testo 7
      <x/>
    </t>
  </t>
</x>
</r>
```

il foglio di stile applicato al documento deve restituire il documento seguente:

```
<r>
  <x>testo 0</x>
  <y>
    <x/>
    <z/>
  </y>
  <x>
    <x/>
    <y>testo 3
      <x/>
    </y>
    <x/>
  </x>
  <t>
    <y>testo 6</y>
    <z/>
  </t>
</x>
</r>
```

**Esercizio 6** Siano URI1, URI2, URI3, URI4 delle URI. Scrivere un modello RDF che rappresenta le seguenti informazioni: "URI1 è stato creato il 30 giugno 2010, URI4 e URI1 hanno lo stesso autore (che però non è conosciuto), URI1 crede che URI2 e URI3 sono stati creati nella stessa data (che però non è conosciuta)".