

Complementi di Basi di dati – Basi di dati II (secondo modulo)
Esame del 1 luglio 2009 – Compito A – Tempo a disposizione: 1 ora e 20 minuti (libri chiusi)

Domanda 1 (20%)

Scrivere (a) un esempio di documento XML valido per il seguente DTD e (b) lo schema di una base di dati relazionale (inclusi i vincoli di integrità) nella quale sia possibile memorizzare i dati in esso contenuti.

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT officina (riparazione+)>
<!ELEMENT riparazione (auto,cliente,data-in,data-out,costo)>
<!ELEMENT auto (marca,modello,cc)>
<!ELEMENT data-in (gg,mm,aa)>
<!ELEMENT data-out (gg,mm,aa)>
<!ELEMENT costo (#PCDATA)>
<!ELEMENT marca (#PCDATA)> <!ELEMENT modello (#PCDATA)> <!ELEMENT cc (#PCDATA)>
<!ELEMENT gg (#PCDATA)> <!ELEMENT mm (#PCDATA)> <!ELEMENT aa (#PCDATA)>
<!ELEMENT cliente EMPTY>
<!ATTLIST cliente cf ID #REQUIRED>
<!ATTLIST cliente nome CDATA #IMPLIED>
<!ATTLIST auto targa ID #REQUIRED>
```

Domanda 2 (30%)

Si vogliono gestire documenti XML in grado di rappresentare articoli in vendita in un sito di commercio elettronico. Ogni articolo ha un codice identificativo, una categoria di articolo (abbigliamento, libro, telefono, ecc.), una sottocategoria (per l'abbigliamento: uomo, donna, sport, scarpe, ecc.), una lista di modelli (taglie per un capo di abbigliamento) con le relative disponibilità, una descrizione dell'articolo in vendita, una lista di file contenenti foto dell'articolo, il prezzo, un eventuale sconto. Scrivere l'XML Schema in grado di validare documenti di questo tipo tenendo conto che si vogliono raggruppare gli articoli nel documento per categoria e, in ogni categoria, per sottocategoria.

Domanda 3 (35%)

Considerare il seguente documento XML:

```
<students>
  <student id="100026">
    <name>Joe Average</name>
    <age>21</age>
    <major>Biology</major>
    <results>
      <result course="Math 101" grade="3"/>
      <result course="Biology 101" grade="4"/>
      <result course="Statistics 101" grade="1"/>
    </results>
  </student>
  <student id="100078">
    <name>Jack Doe</name>
    <age>18</age>
    <major>Physics</major>
    <major>XML Science</major>
    <results>
      <result course="Math 101" grade="9"/>
      <result course="XML 101" grade="8"/>
      <result course="Physics 101" grade="7"/>
      <result course="Statistics 101" grade="9"/>
    </results>
  </student>
</students>
```

(continua sul retro del foglio)

Con riferimento a documenti XML di questi genere:

- (a) scrivere l'espressione XPATH che restituisce i nomi dei corsi sostenuti, con voto maggiore di 8, da studenti con età inferiore a 20 anni
- (b) scrivere il foglio di stile XSLT che restituisce il seguente file XML:

```
<exams>
  <exam name="Math 101" count="2"/>
  <exam name="Biology 101" count="1"/>
  <exam name="Statistics 101" count="2"/>
  <exam name="XML 101" count="1"/>
  <exam name="Physics 101" count="1"/>
</exams>
```

- (c) scrivere l'interrogazione XQuery che, per ogni corso, restituisce un elemento `results` contenente il nome del corso come attributo e, come sottoelementi, tutti gli studenti che hanno sostenuto l'esame con il relativo voto.

Domanda 4 (15%)

Descrivere in maniera sintetica (mezza pagina) le principali differenze tra l'OLAP (On Line Analytical Processing) e il Data Mining, facendo degli esempi pratici.