Two hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

M.Sc. in Advanced Computer Science

Semi-Structured Data and the Web

Date: Tuesday 20\textsuperscript{th} May 2008

Time: 09:45 – 11:45

Please answer BOTH Questions

This is a CLOSED book examination

The use of electronic calculators is NOT permitted
1. **Compulsory**

   a) Describe each of the following terms in one or two sentences:

   i) well-formedness of an XML document
   ii) validity of an XML document
   iii) an XML parser
   iv) a validating XML parser
   v) the DOM tree or infoset of an XML document (5 marks)

   b) In a few sentences, explain what XML namespaces have been designed for, and provide a simple example of their usage. Please try to find an example different from the “table” example in the lecture. (2 marks)

   c) In a few sentences, explain the general purpose of a schema language in general and name three schema languages for XML. (3 marks)

   d) In one sentence, explain what XPath is and where it is used. (1 mark)

   e) Consider the following XML fragment. Is it well-formed? In case it is not, explain why not. In case it is, give, for the elements in bold, their universal names and explain your choice:

   ```xml
   <name xmlns:mns=http://www.bear.org/
       xmlns:yns="http://www.bear.org/
       xmlns="http://www.wolf.org/">
     <lastname xmlns="http://www.cabbage.org/">Smith</lastname>
     <yns:firstname xmlns="http://www.lamb.org/">Smith</yns:firstname>
     <fullname xmlns=http://www.lamb.org/">John</fullname>
   </name>
   ```

   (3 marks)
f) For the following stylesheet and the XML document nodes.xml below, describe what an XSLT processor returns when it runs the stylesheet on the document. Also, explain whether and what changes if the line marked <!-- important? --> is removed. (6 marks)

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:output method="html"/>

  <xsl:template match="deeper">
    <xsl:text> Deeper Node found with: </xsl:text>
    <xsl:value-of select="@type"/><br/>
    <xsl:apply-templates/>            <!-- important? -->
  </xsl:template>

  <xsl:template match="T">
    <xsl:text> T Node found with: </xsl:text>
    <xsl:value-of select="."/><br/>
  </xsl:template>
</xsl:stylesheet>
```
g) For the following XQuery, describe the result when answered over the two XML documents nodes.xml and articles.xml given in the box below and explain your answer. (6 marks)

```xml
<mylist>
{for $dnode in doc("nodes.xml")/root/middle/deeper,
$art in doc("articles.xml")/articlelist/article
where $dnode/@type = $art/@cat
return <product id="{$art/@cat}">
{ $art/description,
<morestuff> { $dnode/T } </morestuff>
}
</product>
}
</mylist>
```

**nodes.xml:**
```xml
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <middle>
    <deeper type="A">
      <T>t1</T>
      <T>t2</T>
    </deeper>
    <deeper type="A">
      <T>t1</T>
      <T>t3</T>
    </deeper>
    <deeper type="F">
      <T>t2</T>
      <T>t5</T>
    </deeper>
  </middle>
</root>
```

**articles.xml:**
```xml
<?xml version="1.0" encoding="UTF-8"?>
<articlelist>
  <article cat="1" price="3.00">
    <description>super wheel</description>
    <supplier>t2</supplier>
  </article>
  <article cat="2" price="3.50">
    <description>great bike</description>
    <supplier>t1</supplier>
  </article>
  <article cat="3" price="3.60">
    <description>nice scooter</description>
    <supplier>t3</supplier>
  </article>
</articlelist>
```
(Question 1 continues from the previous page)

h) Consider the following XML schema declarations. Provide a well-formed element UKShoesize that validates against this declaration and one that does not validate. Provide a well-formed element Company that validates against this declaration and one that does not validate. (4 marks)

i)

```xml
<xs:complexType name="shoesizeType">
    <xs:simpleContent>
        <xs:extension base="xs:integer">
            <xs:attribute name="lang" type="xs:string"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>

<xs:element name="UKShoesize">
    <xs:complexType>
        <xs:simpleContent>
            <xs:restriction base="shoesizeType">
                <xs:minExclusive value="3"/>
                <xs:maxExclusive value="14"/>
            </xs:restriction>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
```

ii)

```xml
<xs:element name="Company">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="name" type="xs:string"/>
            <xs:element name="holding" type="xs:string">
                <xs:attribute name="maxOccurs" type="xs:string" value="3"/>
                <xs:attribute name="minOccurs" type="xs:string" value="1"/>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
```
2. **Compulsory**

b) For each element in lines 10 to 19, give the style content which applies to that element, and explain why that content so applies. (For each element, list the line number of its start tag, then the property/value pairs that apply, then any inheritance, cascading, or overriding. For elements with only the default style, it suffices to write "default").:

1. `<style type="text/css">`
2. `.title {font-weight: bold}`
3. `div.title {text-align:center; font-size: large; }`
4. `div.entry div.title {text-align: left; font-weight: normal}`
5. `span.date {font-style: italic}`
6. `span.date:after{content:" by"}`
7. `div.content {font-style: italic}`
8. `div.content i {font-style: normal; font-weight: bold}`
9. `#one {color: red}</style>`
10. `<div class=title>My Weblog</div>`
11. `<div class="entry">
12. `<div class=title>What I Did Today</div>
13. `<p class=byline>
14. `<span class=date>Feb. 11, 2008</span>`
15. `<span class=author>Bijan Parsia</span>`
16. `</p>`
17. `<p style="color:blue" id="one">Taught a class and
18. it went <i>very</i> well.</p>
19. `</div>`

b) In a few sentences, explain the motivations for separating structure from style. Given an example of structural and of presentational markup in HTML and justify your describing them as such. (4 marks)

c) State Postel's law. (1 mark)

d) In a sentence or two, give a Metcalfe's law based explanation of the success of the Web. (2 marks)

e) List the major parts of a (hierarchical) URI and who determines the meaning of each part. (3 marks)

f) Explain the difference between an architecture and an architectural style. (1 mark)

g) Explain the "uniform interface" style (esp. in the context of REST; i.e., include the 4 constraints). Contrast with the SOA approach. (5 marks)

(Question 2 continues on the following page)
h) Discuss whether XML is self-describing (consider the well-formed and the valid cases)? (4 marks)

i) Compare Schematron and Relax-NG in their ability to support human readable error messages. (2 marks)

j) Compare the error handling strategy of CSS (according to the spec) and HTML (according to HTML5 or as done by browsers). (3 marks)