Two hours - on line

The exam will be taken on line.
This is the paper format, which will be available as a backup
and to be handed out to students for reference immediately AFTER the examination starts

Please do NOT use the exam paper to write your answers

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

Semi-structured Data and the Web

Date: Thursday 20th January 2011
Time: 14:00 - 16:00

Please answer ALL questions provided.
The exam contains MULTIPLE CHOICE QUESTIONS: each of these questions may
have one or more correct answers, and you have to select all correct answers for full
marks.

This is a CLOSED book examination

The use of electronic calculators is NOT permitted
Answer both questions

1. (35 marks)

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

   i) a validating XML parser (2 marks)

   ii) the DOM tree or infoset of an XML document (2 marks)

   iii) the PSVI of an XML document (2 marks)

b) Describe 2 important properties of XML documents that XML schema languages are designed to describe or constrain, and use an example to illustrate these properties. (4 marks)

c) Consider the following XML fragment.

   `<name xmlns:mns="white"
        xmlns:yins="pink"
        xmlns="blue">
      <lastname xmlns="green">Smith</lastname>
      <yins:firstname xmlns="white">
        <fullname>John</fullname>
        <mns:nickname xmlns:mns="pink">Jo</mns:nickname>
        <mns:nickname>Yo</mns:nickname>
      </yins:firstname>
    </name>`

   i) Is it well-formed? In case it is not, explain why not. In case it is, give, for each element, its universal (or expanded) name. (3 marks)

   ii) Comment, in two sentences, on the way namespaces are used in this document: does this seem to be a useful or beneficial way? (2 marks)
d) For the following XQuery, describe the result when answered over the two XML documents nodes.xml and articles.xml given below and explain your answer.

(8 marks)

XQuery:

```xml
<report>
  {for $sale in doc("sales.xml")/sale,
   $art in doc("articles.xml")/articlelist/article
   where $sale/@on = $art/@id
   return <product id="{$art/@id}" price="{$art/@price}" sold="{$sale/@number}">
     { $art/description }
   </product>
  }
</report>
```

articles.xml:

```xml
<articlelist>
  <article id="1" price="3.00">
    <description>super wheel</description>
    <supplier>t2</supplier>
  </article>
  <article id="2" price="3.50">
    <description>great bike</description>
    <supplier>t1</supplier>
  </article>
  <article id="3" price="3.60">
    <description>nice scooter</description>
    <supplier>t3</supplier>
  </article>
</articlelist>

sales.xml:

```xml
<!DOCTYPE sales [ 
<!ELEMENT sales (sale+)>
<!ELEMENT sale EMPTY>
<!ATTLIST sale
    on CDATA "1"
    number CDATA #IMPLIED> ]
<sales>
  <sale number="300"/>
  <sale number="2" on="3"/>
  <sale number="22"/>
  <sale on="7" number="300"/>
</sales>
```
e) i) How many trees are accepted by the grammar \( G = (\{N,M\}, \{a,b\}, \{N\}, P) \), where \( P \) is given below?

\[
P = \begin{cases} 
N & \rightarrow a \ (M, M, M), \\
M & \rightarrow b \ \varepsilon 
\end{cases}
\]

(2 marks)

- not enough information given to be able to say
- no tree at all
- exactly one tree
- at least two trees, but finitely many
- infinitely many trees

ii) How many XML documents with root element \( N \) are valid with respect to the following DTD?

\[
<!ELEMENT N (B+)>
<!ELEMENT B EMPTY>
\]

(2 marks)

- not enough information given to be able to say
- no document at all
- exactly one document
- at least two documents, but finitely many
- infinitely many documents
iii) Which of the following statements are true for the given DTD d.dtd and the XML schema s.xsd? (3 marks)

d.dtd:
<!ELEMENT a (b, c) >
<!ELEMENT b (#PCDATA)>
<!ELEMENT c (#PCDATA) >
<!ATTLIST a IsNice CDATA #IMPLIED "f">

s.xsd:
  <xs:element name="a">
    <xs:complexType>
      <xs:all>
        <xs:element name="b" type="xs:anySimpleType"/>
        <xs:element name="c" type="xs:anySimpleType"/>
      </xs:all>
      <xs:attribute name="IsNice" use="optional"/>
    </xs:complexType>
  </xs:element>
</xs:schema>

- every document that validates against d.dtd also validates against s.xsd.
- every document that validates against s.xsd also validates against d.dtd.
- there are documents that validate against d.dtd but do not validate against s.xsd.
- there are documents that validate against s.xsd but do not validate against d.dtd.
- none of the above

f) Explain when and why it can be useful to use schemas that correspond to single-type grammars. (5 marks)
2. (35 marks)

a) True or False: An XML Document can be well-formed but not valid with respect to a DTD. (1 mark)

b) Consider the following XML Document:

```xml
<!DOCTYPE books [  
<!ELEMENT books (book)+>  
<!ELEMENT book (title, author)>  
<!ELEMENT title (#PCDATA)>  
<!ELEMENT author (#PCDATA)> ]>
<books>
  <book>
    <title>Sister Outsider: Essays & Speeches</title>
    <author>Audre Lorde</author>
  </book>
  <book>
    <title>Pride and Prejudice</title>
    <author>Jane Austen</author>
  </book>
</books>
```

- The document is valid but not well-formed.
- The document is neither well-formed nor valid.
- The document is well-formed but not valid.
- The document is both well-formed and valid. (2 marks)

c) XML’s draconian error handling (i.e., treating well-formedness errors as fatal)

- conforms with Postel’s law.
- violates Postel’s law.
- conforms to the spirit (i.e., its intention), but violates the letter (i.e., the literal, direct interpretation) of Postel’s law.
- conforms to the letter (i.e., the literal, direct interpretation), but violates the spirit (i.e., its intention) of Postel’s law. (2 marks)

d) An XML Document with a DOCTYPE declaration

- can be internally valid.
- must be internally valid.
- can be externally validated against another DTD.
- must be externally validated against another DTD.
- can never be externally validated against another DTD.
e) Briefly (in 4 - 5 sentences) compare the advantages and disadvantages of internal vs. external validation.  

f) True or False: DTD limitations tend to encourage modelling with attributes.

g) Which of the following WXS features alter the PSVI?
   - Groups.
   - Abstract types.
   - Substitution groups.
   - Default elements.
   - Keys.

h) Briefly (in 4 - 5 sentences) explain how XPath can be used as a schema language and the advantages or disadvantages of using it as such.

i) Which of the following are well-formed XML documents?
   - <html><TITEL/></html>
   - <foo bar="a" baz='a'/>
   - <foo bar="a" baz='a'/><foo bar="a" baz='a'/>
   - <html><head><title></head><title><body></body></html>
   - <foo />

j) Write a DTD that describes the following XML document. Strive for a fairly tight, but sensible DTD. (It should be about 6 declarations long).

   <discussion>
     <topic id="htk" title="Hume’s theory of knowledge">
       <discussants>
         <person nick="bigan"/>
         <person nick="ruth"/>
       </discussants>
     </topic>
   </discussion>

k) True or false: Every XML Document can be (externally) validated against an infinite number of DTDs.
1) Consider the following XQuery (recall that declaring a global variable external means that it will be set by the calling environment):

```xquery
declare variable $aBool as xs:boolean external;
declare variable $anInt as xs:integer external;
declare variable $str1 as xs:string external;
declare variable $str2 as xs:string external;
    if ($aBool) then ($anInt + 1) else concat($str1, $str2)
```

i) This query has type

- `xs:integer`
- `xs:positiveInteger`
- `xs:string`
- `union(xs:string, xs:integer)`
- `union(xs:string, xs:positiveInteger)`

(2 marks)

ii) True or false: Can it be cast to `xs:string`? (1 mark)

iii) XQuery's type system is primarily

- strong, manifest, and nominal
- strong, latent, and structural
- strong, latent, and nominal
- weak, latent, and nominal
- weak, manifest, and structural

(2 marks)

iv) Briefly (in 2-3 sentences) explain why one might create a DTD instead of a WXS for a new project. (2 marks)