On-Line Examination  
COMP60372

Two hours

UNIVERSITY OF MANCHESTER  
SCHOOL OF COMPUTER SCIENCE  

M.Sc. in Advanced Computer Science

Semi-Structured Data and the Web

Date:  Thursday 20\textsuperscript{th} May 2010

Time:  14.00 – 16.00

Please answer BOTH Questions

This is a CLOSED book examination

The exam will be taken on line.
This is the paper format, which will be available as a backup

The use of electronic calculators is NOT permitted

[PTO]
1. You must answer ALL subquestions. (30 marks total)

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

      i) well-formedness of an XML document (1 mark)
      ii) validity of an XML document (1 mark)
      iii) a validating XML parser (1 mark)
      iv) the DOM tree or infoset of an XML document (1 mark)
      v) the PSVI of an XML document (1 mark)

   b) Describe two aspects of an XML document that most XML schema languages are designed to describe and constrain. For each aspect, give an example of a constraint with respect to this aspect. (4 marks)

   c) We have discussed the concepts of tree grammars, single-typed and local tree grammars. Answer each of the following questions in one or two sentences.

      i) What are tree grammars used for in the context of XML? (1 mark)
      ii) What is a "local" tree grammar? (1 mark)
      iii) When is being "single-type" relevant for tree grammars in the context of XML? (2 marks)
      iv) Briefly explain how we can build an efficient validator to validate a document against a Relax NG schema, and how much memory will it need (in the depth or size of the document)? (2 marks)
d) Consider the following XML schema declarations. Provide an element `UKWomensClothesSize` and an element `Company` that validates against these declarations.

```
<xs:complexType name="clothesSizeType">
  <xs:simpleContent>
    <xs:extension base="xs:integer">
      <xs:attribute name="gender" type="xs:string"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="UKWomensClothesSize">
  <xs:complexType>
    <xs:simpleContent>
      <xs:restriction base="clothesSizeType">
        <xs:minExclusive value="6"/>
        <xs:maxExclusive value="26"/>
        <xs:attribute name="gender" type="xs:string" fixed="woman"/>
      </xs:restriction>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>

<xs:element name="Company">
  <xs:complexType>
    <xs:choice>
      <xs:element name="name" type="xs:string"/>
      <xs:element name="holding" type="xs:string">
        maxOccurs="3" minOccurs="1"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
```

(2 marks)

(Question 1 continues on the following page)
(Question 1 continues from the previous page)

e) For the following XQuery, describe the result when answered over the two XML documents myDocument.xml and myDocument2.xml given in the box below and explain your answer.  

```xquery
<colatedList>
{ for $d in doc("myDocument.xml")//deeper,
  $art in doc("myDocument2.xml")//article[@price > "2"]
where $d/@type = $art/@cat
return
  <prod id="{$art/@cat}'">
  { $art/name,
    <more> { $d/T } </more>
  }
  </prod>
}
</colatedList>
```

myDocument.xml:

```xml
<!DOCTYPE root [ 
  <!ELEMENT root (deeper+)>
  <!ELEMENT deeper (T|(T,T))>
  <!ATTLIST deeper type CDATA 'B'>
  <!ELEMENT T (#PCDATA)> ]>

<root>
  <deeper type="A">
    <T>t1</T>
  </deeper>
  <deeper type="C">
    <T>t3</T>
  </deeper>
  <deeper>
    <T>t5</T>
    <T>t6</T>
  </deeper>
</root>
```

(Question 1 continues on the following page)
(Question 1 continues from the previous page)

myDocument2.xml:

```xml
<list>
  <article cat="A" price="2">
    <name>Thing A</name>
  </article>
  <article cat="B" price="3">
    <name>Thing B</name>
  </article>
  <article cat="C" price="3.5">
    <name>Thing C</name>
  </article>
</list>
```

f) In schema languages for XML, various mechanisms have been introduced to express uniqueness of elements: sketch DTD's ID and IDREF mechanism and how XML Schema extends this, and give an example of where and how they could suitably be used. (4 marks)

```xml
<list>
  <article cat="A" price="2">
    <name>Thing A</name>
  </article>
  <article cat="B" price="3">
    <name>Thing B</name>
  </article>
  <article cat="C" price="3.5">
    <name>Thing C</name>
  </article>
</list>
```

g) Explain why one would want to know whether a schema is empty, and why one would want to know whether it is contained in another schema. (3 marks)
2. You must answer all subquestions. (30 marks total)

a) Give two reasons for having a schema for XML documents. (4 marks)

b) State Postel's law. (2 marks)

c) What is the “contained” namespace pattern? If one were to use this pattern, why would it be a bad idea to require that your documents be in namespace normal form? (4 marks)

d) What is a datatype (in general)? (1 mark)

e) Explain (with an example) the difference between a nominal and structural type system? (3 marks)

f) Explain the difference, in the XQuery formal semantics, between:

   a value \( V \) matching a type \( T \)
   and

   a value \( V \) validating as a type \( T \)? (2 marks)

e) List which of schema languages we studied in the course (DTD, WXS, RELAX NG, and Schematron) can constrain the content of the element <age> to be positive integers, that is, can rule out: <age>twenty-nine</age> or <age>-900</age> while accepting: <age>16</age>. (2 marks)

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

```
<argument>
  <topic id="htk" title="Hume's theory of knowledge"></topic>
  <arguers>
    <person><nick>bijan"</nick></person>
    <person><nick>ruth</nick></person>
  </arguers>
</argument>
```

g) What are the benefits of an XSugar grammar (.xsg file)? (2 marks)
h) Contrast the error handling philosophy of Schematron and CSS. (2 marks)

i) Consider the following XQuery (recall that declaring a global variable “external” means that it will be set by the calling environment):

```xml
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)
```

What is the type of this query? Can it be cast to `xs:integer` (and why)? (2 marks)