Final Exam for XML (Fall of 2007)
Instructor: Kyuseok Shim

1. Consider the following XML document, PersonList.xml:

```
<?xml version="1.0" ?>
<PersonList Type="Student" Date="20061221">
   <Title Value="Student List"/>
   <Contents>
      <Person><Name>Joe Doe</Name>
        <Id>s11111111</Id>
        <Addr><Num>123</Num><Str>Elm St</Str></Addr>
      </Person>
      <Person><Name>John Ho</Name>
        <Id>s66666666</Id>
        <Addr><Num>321</Num><Str>Mall Rd</Str></Addr>
      </Person>
   </Contents>
</PersonList>
```

   (a) Write a DTD for the above XML document. [10 points]
   (b) Write an XML Schema for the above XML document. [10 points]

2. Consider the following XML document, StudentList.xml:

```
<?xml version="1.0" ?>
<!-- Some comment -->
<Students>
   <Student StudId="111111111" >
      <Name><First>John</First><Last>Doe</Last></Name>
      <Status>U2</Status>
      <CrsTaken CrsCode="CS308" Semester="F1997" />
      <CrsTaken CrsCode="MAT123" Semester="F1997" />
   </Student>
   <Student StudId="987654321" >
      <Name><First>Bart</First><Last>Simpson</Last></Name>
      <Status>U4</Status>
      <CrsTaken CrsCode="CS308" Semester="F1994" />
   </Student>
</Students>
<!-- Some other comment -->
```

   Give the XPath expressions for the following queries.

   (a) Return the second CrsTaken child of first Student child of Students. [5 points]

```
//Students/Student[1]/CrsTaken[2]
```

(b) Return all the last CrsTaken elements within each Student element. [5 points]

```
//Students/Student/CrsTaken[last( )]
```

(c) Return the students who have taken CSC343. [5 points]

```
//Student[CrsTaken/@CrsCode="CSC343"]
```

3. Consider the following XML document, TranscriptList.xml:

```
<Transcripts>
   <Transcript>
      <Student StudId="111111111" Name="John Doe" />
      <CrsTaken CrsCode="CS308" Sem="F97" Gr="B" />
      <CrsTaken CrsCode="MAT123" Sem="F97" Gr="B" />
      <CrsTaken CrsCode="EE101" Sem="F1997" Gr="A" />
      <CrsTaken CrsCode="CS305" Sem="F1995" Gr="A" />
   </Transcript>
   <Transcript>
      <Student StudId="987654321" Name="Bart Simpson" />
      <CrsTaken CrsCode="CS305" Sem="F1995" Gr="C" />
      <CrsTaken CrsCode="CS308" Sem="F1994" Gr="B" />
   </Transcript>
   <Transcript>
      <Student StudId="123454321" Name="Joe Blow" />
      <CrsTaken CrsCode="CS315" Sem="S97" Gr="A" />
      <CrsTaken CrsCode="CS305" Sem="S96" Gr="A" />
      <CrsTaken CrsCode="MAT123" Sem="S96" Gr="C" />
   </Transcript>
   <Transcript>
      <Student StudId="023456789" Name="Homer Simpson" />
      <CrsTaken CrsCode="EE101" Sem="F1995" Gr="B" />
      <CrsTaken CrsCode="CS305" Sem="S1996" Gr="A" />
   </Transcript>
</Transcripts>
```

Give the XQuery expressions for the following queries.

(a) Return all students who took MAT123. [5 points]

```
for $t in doc("http://uoft.edu/transcript.xml")//Transcript
where $t/CrsTaken/@CrsCode = "MAT123"
return $t/Student
```

(b) Produce a list of students along with the number of courses each student took. [10 points]

```
for $t in fn:doc("transcripts.xml")//Transcript, $s in $t/Student
    let $c := $t/CrsTaken
    let $d := count(distinct-values($c/@CrsCode))
order by $d
```

```
        return
            <StudSummary StudId="{$s/@StudId}"
                         Name="{$s/@Name}"
                         TotalCourses = "{$d}" />
```

4. Consider the queries, $Q_1 = /a/b//c$, $Q_2 = //b/*/c/d$, $Q_3 = /*/a/c//d$ and $Q_4 = /a/*/*/c//e$. Show path nodes for each query and Query index in the X-Filter system. [10 points]

5. Consider an XML document that has the distribution of the elements as follows. We will represent as (element, frequency) for each element and its frequency.

   (book, 0.1), (author, 0.1), (title, 0.1), (section, 0.3), (subsection 0.3), (subtitle, 0.1)

   Encode the simple path of book.section.subsection as an interval. [10 points]

6. Describe the definition of PageRank and how it is computed given Web documents. [10 points]

7. Consider the following 4 HTML documents. The position numbers were shown for your convenience. Please ignore the numbers when you look at the HTML documents, but use them when you compute their common template by Arasu and Garcia-Molina's SIGMOD'03 paper. Explain basic idea of the paper and show how template is computed and how the values are extracted from HTML documents using computed template. [20 points]

```
<html>1<body>2
    <b>3Book4Name5</b>6Databases
    <b>7Reviews8</b>9
    <ol>10
        <li>11
            <b>12Reviewer13Name14</b>15John
            <b>16Rating17</b>18 7
            <b>19Text20</b>21
        </li>22
    </ol>23
</body>24</html>25

<html>1<body>2
    <b>3Book4Name5</b>6Query Opt.
    <b>7Reviews8</b>9
    <ol>10
        <li>11
            <b>12Reviewer13Name14</b>15John
            <b>16Rating17</b>18 8
            <b>19Text20</b>21
        </li>22
    </ol>23
</body>24</html>25

<html>1<body>2
    <b>3Book4Name5</b>6Data Mining
    <b>7Reviews8</b>9
    <ol>10
        <li>11
```

```
            <b>12Reviewer13Name14</b>15Jeff
            <b>16Rating17</b>18 2
            <b>19Text20</b>21
        </li>22
        <li>11
            <b>12Reviewer13Name14</b>15Jane
            <b>16Rating17</b>18 6
            <b>19Text20</b>21
        </li>22
    </ol>23
</body>24</html>25

<html>1<body>2
    <b>3Book4Name5</b>6Transactions
    <b>7Reviews8</b>9
    <ol>10
</ol>23 </body>24</html>25
```