UCT CSC303 2005 :: XML/IR + ToA :: Test [35 marks]

Answer Questions 1, 2 AND 3.

Question 1: XML / XSLT [10]

Answer the following questions based on this piece of XML:

```
<mdata>
   <name>CSC303 Test</name>
   <author>
      <first_name>hussein</first_name>
      <last_name>suleman</last_name>
   </author>
</mdata>
```

Assume that the **name** and **author** elements are both infinitely repeatable and optional and that **first_name** and **last_name** must both appear exactly once each.

1. Write an XML Schema complexType type definition **mdataType** corresponding to the content of the **mdata** element and its descendents. [5]

```
<complexType name="mdataType">
  <sequence>
    <element name="name" minOccurs="0" maxOccurs="unbounded" type="string">
    <element name="author" minOccurs="0" maxOccurs="unbounded">
      <complexType>
        <sequence>
          <element name="first_name" type="string"/>
          <element name="last_name" type="string"/>
        </sequence>
      </complexType>
    </element>
  </sequence>
</complexType>
```

Minus one mark for each major error.

2. Write an XPath expression that locates the **first_name** node corresponding to the first **author** node (assuming there could be multiple authors). The current context node is the root node **mdata**. [1]

author[1]/first_name

3. The given XML document is well-formed - what 2 properties make it well-formed? [2]

1 – single root 2 – properly nested matching start and end tags

4. If exactly one bit of a well-formed XML document is corrupted, how does the self-segregrating nature of UTF-8 prevent cascading errors. [2]

each UTF-8 code has a start byte that is unique from follow-on bytes. thus, even if there is corruption, the XML stream can be synchronised from the next start byte.

Question 2: Information Retrieval [10]

1. Inverted files can be used to optimise filtering. What are inverted files and what is filtering? [2]

an inverted file is a list of all documents in which a term occurs [1]

filtering is the process of removing documents that are not relevant from the result set [1]

2. In a typical inverted file, how can we reduce the number of bytes required without reducing the information content? [1]

use differential encoding for sorted sequences of numbers

3. Discuss 2 different storage approaches for inverted files. [2]

use a single file with all terms in it [1]

use a database table [1]

use one file for each term [1]

4. Inverted files only produce a list of exact matches, which is not always enough – discuss one technique to increase recall. [2]

LSI returns documents that are inherently similar even if they don't contain the specified terms [2]

stemming increases the range of terms covered to all prefix/suffix forms of the query [2]

a thesaurus will provide synonyms that can be used to match additional documents [2]

etc.

5. After filtering documents, we can rank them on the basis of links. Describe the simple PageRank algorithm used for this purpose. [3]

step1: Assign arbitrary rank values to all nodes such that sum=1 [1]

step2: calculate new rank for each node as the sum of all weights of incoming links (where the rank of each predecessor node is distributed evenly to all outgoing links) [1]

step3: iterate step2 until a steady state is reached [1]