Universität Freiburg
Institut für Informatik
Prof. Dr. P. Fischer
Anas Alzobgi

Georges-Köhler Allee, Geb. 51
D-79110 Freiburg

Data Models and Query Languages
Summer 12
Due/Discussion by 12.07.2012

8. Sheet: XQuery (2), XQuery Update, XQuery Fulltext

Exercise 1 (Properties of XQuery Comparison Operators – 15 points)

a) Find a variable binding for $x so that $x=1 and $x=2. Can one infer that, in XQuery, 1=2?

b) Find variable bindings for $x, $y and $z so that $x > $y and $y > $z, but $x > $z is not true.

c) Find a variable binding for $x so that neither $x eq $x nor $x = $x is true. Explain why.

Exercise 2 (Element constructors and node identities – 10 points)

a) Given is the following XQuery expression:

```
let $a := <a/> let $b := <b>{$a}</b> return $b/a is $a
```

Is the result true, false, or an error? Explain the result.

b) As a comparison, what is the result of the following expression?

```
let $a := <a/> let $b := $a return $b is $a
```

Exercise 3 (Halloween Problem and Snapshot Semantics – 10 points)

The following XML document is given:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
    <node>text<node>text</node>text</node>
    <node>text</node>
    <node>text<node><node/>text</node>text</node>
</root>
```

What is the result of the following XQuery updating expression? Why?

```
for $n in /root//node()
where count($n/element()) eq 0
return delete node $n
```

Exercise 4 (Simple exrpessions and updating expressions - 20 points)

Which of the following expressions/functions are allowed, according to the composition restrictions of the XQuery Update Facility. Which ones are allowed with the XQuery Scripting Extension?

a) `(delete node $x, insert node $y into $z)`

b) `(insert node $x as first into $y, delete node $z, <a/>)`

c) `(insert node $x before $y, (), fn:error())`

d) `insert node <a/> into insert node <b/> into doc("file.xml")`

e)
```
declare function local:fun($i as item()*) {()};
        local:fun(insert node $x into $y)
```

f)
```
declare updating function local:upd($i as node())
       {
           copy $x := $i modify delete node $x/a return $x
        };
```

Exercise 5 (Recursive XQuery - 30 points)
Consider the XML document flights.xml validated against exercise6-1.xsd) which describes an airline flight scenario. Write the XQuery/XQuery update expressions for solving the following problems. If a result is returned, it should be well-formed XML.

a) Give the list of the direct flights on the date of 2009-12-24 which have N̈orth Pole(̈airport name) as the source airport.

b) Consider the case of combined flights (two or more). As an example, flying from London to Zurich on the date of 2008-12-24 might mean taking two separate flights: London-Amsterdam and Amsterdam Zurich, both on the same date. Retrieve all flight possibilities from North poleto SSouth poleön the date of 2009-12-24 with one or two intermediate stops. Note that the schema was enhanced with departure and arrival times.

Exercise 6 (Updates - 15 points)
Use the flight reservation XML file from the previous to perform the following operations:

a) Return the list of flights, but without the number of seats.

b) Update the date of a given reservation. Make sure (programmatically) that the flight mentioned in the reservation data exists at the new date.

c) Delete an airport. Ensure that all depending data objects (flights, reservations) are also deleted.

You do not need to use schema validation when performing these operations.