Prof. Dr. Stefan Brass
Institut für Informatik
MLU Halle-Wittenberg

24. Januar 2019

XML und Datenbanken
— 12. Übungsblatt: XQuery —

Bei Teil a) und b) handelt es sich um Präsenzübungen, die nicht abgegeben werden müssen. Sie sollten über die Wiederholungsfragen a) allerdings vor der nächsten Übung selbst nachdenken. Teil c) bis e) sind Hausaufgaben. Bitte schicken Sie eine Lösung per EMail an den Dozenten (mit "`xml18`" in der Betreff-Zeile, bis zum 30.01.2019). Schicken Sie bitte keine leeren EMails nur mit Anhang, diese landen im Spam-Ordner.

a) Wie würden Sie in einer mündlichen Prüfung auf folgende Fragen zu XQuery antworten?

- Was ist die wichtigste Einschränkung von XPath gegenüber XQuery (d.h. was kann man in XQuery machen, was in XPath nicht geht)?
- Was sind FLWOR-Ausdrücke? Nennen Sie die Schlüsselworte und erklären Sie den Zweck jeder Klausel.
- Vergleichen Sie FLWOR-Ausdrücke mit SQL. Was sind die Entsprechungen von SELECT und FROM (WHERE ist ja offensichtlich)?
- Wie sehen direkte Konstruktoren aus? An welchen Stellen darf man auszuwertende Ausdrücke einfügen? Wie werden diese markiert?
- Sei der direkte Konstruktor `<E>{Q}</E>` gegeben. Die Auswertung der Anfrage `Q` liefere eine Sequenz `S`. Wie sieht das Ergebnis aus, wenn `S` eine Sequenz von atomaren Werten ist? Was passiert, wenn `S` am Anfang Attribut-Knoten enthält?
- Was ist "Boundary Whitespace" in einem direkten Konstruktor? Warum ist dieser Begriff wichtig?
- Nennen Sie ein System, mit dem Sie XQuery-Anfragen auswerten können (also eine XML-Datenbank).
- Was ist die aktuelle Version von XQuery?
- Es sei die folgende einfache SQL-Anfrage gegeben:

```
SELECT E.ENAME, E.SAL
FROM   EMP E, DEPT D
WHERE  E.DEPTNO = D.DEPTNO
AND    D.DNAME = 'RESEARCH'
```

  Nennen Sie mindestens zwei verschiedene Arten, wie man diese Anfrage nach XQuery übersetzen kann, wenn die Relationen auf eine übliche Art (z.B. mit den Daten in Attributen) in XML repräsentiert sind.

- Es sei der folgende FLWOR-Ausdruck gegeben:

```
for    $x in //a
where  exists($x/b) and $x/@c > 0
return $x/d
```

  Geben Sie einige äquivalente Version der Anfrage an. Wo kann man Bedingungen schreiben außer unter `where`?
- Was ist Ihre Meinung zur Behauptung "XQuery Anfragen sind oft länger und komplexer als die entsprechenden SQL Anfrage"?
- Angenommen, das Ergebnis der Anfrage "`//a/b[c and d]`" ist leer, aber Sie wissen, dass das nicht die erwartete Antwort ist. Was würden Sie tun, um den Fehler zu finden?
- Wie würde sich eine typische XQuery-Implementierung verhalten, wenn man Elementnamen oder Attributnamen falsch schreibt? Wie verhalten sich typische SQL-Implementierungen, wenn man Tabellen- oder Spaltennamen falsch schreibt?
- Wie kann man `LIKE`-Bedingungen in XQuery ausdrücken?
- Wie kann man in XQuery angeben, dass man die Ergebnisse "absteigend" sortiert haben möchte, also mit dem größten Wert zuerst? Übertragen Sie dieses Beispiel nach XQuery:

```
SELECT E.ENAME, E.SAL
FROM   EMP E
ORDER  BY E.SAL DESC
```

  Was ist der Unterschied von XQuery zu SQL bezüglich der absteigenden Sortierung?
- Angenommen, der Sortierwert `$x/@A` existiert nicht für einige Variablenbelegungen (d.h. dieser "`order by`"-Ausdruck liefert die leere Sequenz). Die Antworten für diese Variablenbelegungen sollen zuletzt in der Ergebnisfolge erscheinen. Wie kann man das spezifizieren (bei der normalen, "`ascending`" Ordnung)?

Präsenzaufgaben

b) Wir werden die Klausur aus dem Wintersemester 2016/17 durchrechnen:

[http://users.informatik.uni-halle.de/~brass/xml18/exams/exam16.pdf]

Hausaufgabe

Verwenden Sie nochmals die XML-Datensammlung für klassische Musik-CDs (diesmal aber die deutsche Version):

- Daten:
  [http://www.informatik.uni-halle.de/~brass/xml18/examples/cd/cd.xml]
- XML Schema:
  [http://www.informatik.uni-halle.de/~brass/xml18/examples/cd/cd.xsd]
- DTD:
  [http://www.informatik.uni-halle.de/~brass/xml18/examples/cd/cd.dtd]

Die englische Version ist jetzt verschoben in `cd_en`.

Das Dokument enthält die folgenden Elemente:

- `CDDB`: `(komponisten, cds?, solisten?)`
- `komponisten`: `(komponist*)`
- `komponist`: `(stueck*)`
  Attribute: `knr`, `vorname`, `name`, `geboren`, `gestorben`.
- `stueck`: `(aufnahme*)`
  Attribute: `snr`, `titel`, `tonart`, `opus`.
- `aufnahme`: Leerer Inhalt.
  Attribute: `aufnnr`, `orchester`, `leitung`.
- `cds`: `(cd*)`.
- `cd`: `(CDAufnahme*)`
  Attribute: `cdnr`, `name`, `hersteller`, `anz-cds`, `gesamtspielzeit`.
- `CDAufnahme`: Leerer Inhalt.
  Attribute: `aufnnr`.
- `solisten`: `(solist*)`.
- `solist`: `(solist_aufnahme*)`
  Attribute: `name`.
- `solist_aufnahme`: Leerer Inhalt.
  Attribute: `aufnnr`, `instrument`.

Schreiben Sie die folgenden Anfragen in XQuery und testen Sie Ihre Lösung mit einer XQuery-Implementierung.

c) Erzeugen Sie ein XML-Dokument, das alle Komponisten enthält, aber nur mit den Attributen `vorname`, `name`, `geboren`. Die `stueck`-Elemente, die in den `komponist`-Elementen geschachtelt sind, sollen nicht in der Ausgabe erscheinen. Die Ausgabe soll also so aussehen:

```
<HA12>
    <komponist vorname='Georg Friedrich' name='Händel'
               geboren='1685' />
    <komponist vorname='Serge' name='Prokofiev'
               geboren='1891' />
    ...
</HA12>
```

Die Zeilenumbrüche und die Einrückung werden von der Implementierung gewählt und können anders aussehen, als hier gezeigt. Die Reihenfolge der “`komponist`”-Elemente soll die gleiche Reihenfolge wie in der Eingabedatei sein.

d) Erzeugen Sie eine Liste aller Komponisten, die vor 1700 geboren wurden, geordnet nach dem Geburtsjahr (absteigend). Die Ausgabe soll eine ungeordnete Liste in HTML sein, und wie folgt strukturiert:

```
<ul>
    <li>Lean-Marie Leclair: 1697</li>
    <li>Pietro Locatelli: 1695</li>
    <li>Benedetto Marcello: 1686</li>
    <li>Georg Friedrich Händel: 1685</li>
    <li>Johann Sebastian Bach: 1685</li>
    <li>Domenico Scarlatti: 1685</li>
    <li>Georg Philipp Telemann: 1681</li>
    ...
</ul>
```

e) Gesucht sind alle Komponisten zusammen mit der Anzahl Stücke des jeweiligen Komponisten. Es sollen nur Komponisten mit mindestens 5 Stücken berücksichtigt werden. Die Ausgabe soll nach der Anzahl Stücke sortiert werden (größte Anzahl zuerst). Die Ausgabe soll als HTML Tabelle geschehen, also folgendermaßen aussehen:

```
<table>
    <tr><th>Name</th><th>Vorname</th><th>Anzahl</th></tr>
    <tr><td>Händel</td><td>Georg Friedrich</td><td>25</td></tr>
    ...
</table>
```