

**Web-Informationssysteme, WS 2009/10**  
**Übungsblatt 6**

Besprechung am Di 08.12.2009

**Aufgabe 6-1 XML, Datenrepräsentation**

Beschreiben Sie den folgenden arithmetischen Ausdruck durch ein XML-Fragment:

$$5 \cdot x^3 - 2 \cdot x + 7$$

Geben Sie mindestens zwei verschiedene Lösungen an und diskutieren Sie, welche davon in welchen Fällen sinnvoll sind.

Finden Sie wenigstens eine verbreitete XML-Anwendungen zur Repräsentation von mathematischen Formeln und stellen Sie den Ausdruck in dieser dar.

**Aufgabe 6-2 XML, Datenrepräsentation II**

Die folgenden Daten zum Vorlesungsbesuch sollen in XML repräsentiert werden:

Vier Studenten *A, B, C, D* und drei Vorlesungen *I, II, III*.

*I* wird von *A, B, C* besucht, *II* von *B, C* und *III* von *C, D*.

Überlegen Sie sich wie Sie diese Daten in XML darstellen können, wenn

- es möglichst einfach sein soll, die Studenten einer Vorlesung zu finden.
- es möglichst einfach sein soll, die Vorlesungen eines Studenten zu finden.
- beide Fälle gleich wichtig sind.
- Änderungen an Daten über Studenten (Name, Matrikelnummer, etc.) und Vorlesungen häufig auftreten.
- die Größe des XML-Dokuments möglichst gering sein soll.

**Aufgabe 6-3 XML, Strukturbeschreibung von Textdokumenten**

*Training*

Ergänzen Sie das folgende Element, so dass es eine wohlgeformte Beschreibung der Struktur von Goethes „Faust“ (erster Teil) ergibt. Eine Online-Version des Texts mit Versnummern finden Sie unter <http://sciencesoft.at/book/book?book=Faust1>.

```
<LiterarischesWerk>
  <Autor>Johann Wolfgang Goethe</Autor>
  <Titel>Faust</Titel>
  <Untertitel>Der Tragödie erster Teil</Untertitel>
  ...
</LiterarischesWerk>
```

Hinweise: eine Zeile wie

*Ihr naht euch wieder, schwankende Gestalten,*

nennt man einen *Vers*. Es ist sinnvoll, sich einen Überblick bis etwa Vers 1350 zu verschaffen, um die Struktur des Werks zu erfassen. Außerdem empfiehlt es sich, zunächst die Struktur hinter Vers 353 zu klären und erst danach die Struktur des Anfangs. Berücksichtigen Sie auch Fälle wie Vers 299, Vers 485.

Selbstverständlich können Sie geeignete Teile des Elements mit drei Punkten abkürzen.

#### Aufgabe 6-4 XML-Parser

*Vertiefung*

Ein XML-Parser liest ein XML-Dokument ein und baut dabei entweder eine interne Darstellung des XML-Baums auf oder generiert Ereignisse, die von einem konsumierenden Prozess interpretiert werden können.

- a) Skizzieren Sie die Operationen eines Kellerautomaten, der ein XML-Dokument bestehend aus Elementen, Text und Attributen einlesen kann, und erläutern Sie die Arbeitsweise Ihres Kellerautomaten an dem folgenden Dokument:

```
<?xml version='1.0'?>
2 <article>
  <para id='intro'>Hello, world!</para>
4 </article>
```

Um diesen Kellerautomaten zu definieren, ist es einfacher auf einem Strom von Ereignissen (*events*) statt direkt auf dem Zeichenstrom zu operieren. Dies ist ähnlich zu der Verwendung von Symbol-Strömen (*token streams*) in Parsern. Das obige Dokument kann in dieser Darstellung wie folgt repräsentiert werden:

```
start_doc
2 start_elem:  article
  start_elem:  para @id='intro'
4 characters:  Hello, world!
  end_elem:    para
6 end_elem:    article
end_doc
```

- b) Welche Ereignistypen mit welchen Informationen benötigt ein solcher Kellerautomat, damit keine relevante Information verloren geht? (Dokument kann 1:1 reproduziert werden).

#### Aufgabe 6-5 Wohlgeformtes XML

*Vertiefung*

Mit einem Ereignis-basierten XML-Parser wie in der vorigen Aufgaben können Sie in einem Durchlauf über ein XML-Dokument dessen Wohlgeformtheit (hier: passende *start/end tags*, korrekte Schachtelung, und eindeutige Attributnamen) überprüfen.

- a) Welche Datenstrukturen benötigen Sie, um die Wohlgeformtheit zu garantieren?
- b) Skizzieren Sie für jeden Ereignistyp aus der vorigen Aufgabe die Aktionen, um die Wohlgeformtheit zu überprüfen.