

Aufgabe 2:

(6)

Betrachten Sie die folgende XML- Datei **ns.xml**:

```
<?xml version="1.0" encoding="UTF-8"?>
<A xmlns="uri1">
  <ns1:B xmlns="uri2" xmlns:ns1="uri1">
    <D attr1="X"/>
  </ns1:B>
  <C xmlns="uri2">
    <E attr2="Y"/>
  </C>
</A>
```

Kreuzen Sie an, ob die folgenden Aussagen für die Datei **ns.xml** wahr oder falsch sind.

- | | |
|--|--|
| 1. Element E und Attribut attr2 liegen im selben Namespace. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 2. Element A und Element B liegen im selben Namespace. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 3. Elemente A und E liegen im selben Namespace. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 4. Element D liegt im leeren Namespace. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrekter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 3:

(12)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | |
|---|--|
| 1. Um die Wohlgeformtheit eines XML-Dokuments zu überprüfen wird eine DTD oder ein XML-Schema benötigt. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 2. In einem wohlgeformten XML-Dokument dürfen zwei Attribute mit demselben qualifizierten Namen in einem Elementstarttag auftreten. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 3. Die DTD Elementdeklaration (A* B* C*) ist gleichwertig mit (A B C)*. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 4. Rekursive Definitionen wie z.B. <!ELEMENT B (B?,C)> sind in DTDs verboten. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 5. Im DOM-Modell ist die Wurzel gleich dem Wuzelement des Dokuments. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 6. SAX: Textevents können auch direkt hintereinander auftreten. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 7. Die Reihenfolge der Templates in einem XSLT 1.0 Stylesheet hat standardmäßig keine Auswirkung auf den Ablauf des XSLT-Stylesheets. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 8. In XQuery sind selbst definierte rekursive Funktionen erlaubt. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrekter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(6)

Betrachten Sie die folgende XML- Datei **ns.xml**:

```
<A xmlns:ns="URI_A">
  <ns:B xmlns="URI_B">
    <D>abc</D>
  </ns:B>
  <C xmlns="URI_A">
    <ns:E xmlns:ns="URI_B">
      <C>
    </C>
  </C>
</A>
```

Kreuzen Sie an, ob die folgenden Aussagen für die Datei **ns.xml** wahr oder falsch sind.

- | | |
|---|--|
| 1. Elemente A und B liegen im selben Namespace. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 2. Elemente A und C liegen im selben Namespace. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 3. Elemente B und C liegen im selben Namespace. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 4. Elemente D und E liegen im selben Namespace. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrekter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 3:

(12)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | |
|--|--|
| 1. CDATA-Sections werden von XML-Parsern ignoriert. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 2. Die DTD Elementdeklaration (A+ B+ C+) ist gleichwertig mit (A B C)+. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 3. Mit XSLT ist es möglich, XML Dokumente in beliebige andere Dokumente (zB Text, HTML etc) zu transformieren. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 4. Ein XSLT-Stylesheet muss ein wohlgeformtes XML-Dokument sein. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 5. SAX verwendet die DOM API um ein XML Dokument zu lesen. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 6. Der Speicherbedarf eines DOM-Parsers ist unabhängig von der Größe des geparsten XML-Files. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 7. DOM erlaubt wahlfreien Zugriff auf das gesamte XML Dokument. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 8. Bei DOM sind die Attributwerte direkt im Elementknoten gespeichert. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrekter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(6)

Betrachten Sie die folgende XML- Datei **ns.xml**:

```
<A xmlns="namespace1">
  <ns:B xmlns:ns="namespace2">
    <D>abc</D>
  </ns:B>
  <C xmlns="namespace2">
    <E/>
  </C>
</A>
```

Kreuzen Sie an, ob die folgenden Aussagen für die Datei **ns.xml** wahr oder falsch sind.

- | | |
|---|--|
| 1. Element B liegt im Namespace "namespace2". | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 2. Element D liegt im Namespace "namespace2". | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 3. Element C liegt im Namespace "namespace1". | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 4. Element E liegt im Namespace "namespace1". | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrekter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 3:

(12)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | |
|--|--|
| 1. Bei SAX-Parsern muss sich im Allgemeinen der Programmierer / die Programmiererin um die Überprüfung auf Wohlgeformtheit des eingelesenen XML-Dokuments kümmern. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 2. In einem wohlgeformten XML-Dokument müssen die lokalen Namen der Attribute innerhalb eines Elementstarttags verschieden sein. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 3. Die DTD Elementdeklaration (B+,C,C*) ist gleichwertig mit (B,B*,C+). | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 4. Rekursive Definitionen wie z.B. <!ELEMENT B (B?,C)> sind in DTDs erlaubt. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 5. DOM erlaubt wahlfreien Zugriff auf das gesamte XML Dokument. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 6. SAX: Textevents können auch direkt hintereinander auftreten. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 7. Die Ordnung der Knoten im Resultat eines XPath-Ausdrucks ist immer in Document Order. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 8. Der XPath-Ausdruck //status ist die Kurzschreibweise des XPath-Ausdrucks /descendant::status | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrekter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(6)

Betrachten Sie die folgende XML- Datei **ns.xml**:

```
<A xmlns:ns="www.NS1.at">
  <ns:B xmlns="www.NS2.at">
    <D>abc</D>
  </ns:B>
  <ns:C xmlns:ns2="www.NS2.at">
    <ns2:E><F/></ns2:E>
  </ns:C>
</A>
```

Kreuzen Sie an, ob die folgenden Aussagen für die Datei **ns.xml** wahr oder falsch sind.

- | | | |
|---|---------------------------------------|---|
| 1. Elemente A und B liegen im selben Namespace. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 2. Elemente B und C liegen im selben Namespace. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 3. Elemente D und E liegen im selben Namespace. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 4. Elemente E und F liegen im selben Namespace. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrekter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 3:

(12)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | | |
|--|---------------------------------------|---|
| 1. Eine XML Schema Definition muss ein wohlgeformtes XML-Dokument sein. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 2. Der Speicherbedarf eines DOM-Parsers ist unabhängig von der Größe des geparsten XML-Files. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 3. Die Ordnung der Knoten im Resultat eines XPath-Ausdrucks ist immer in Document-Order. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 4. Um die Wohlgeformtheit eines XML-Dokuments zu überprüfen wird eine DTD oder ein XML Schema benötigt. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 5. Die DTD Elementdeklaration (A? B? C?) ist gleichwertig mit (A B C)?. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 6. SAX Textevents treten niemals direkt hintereinander auf. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 7. Wenn in XSLT für einen Knoten kein Template angegeben wird, dann wird die Verarbeitung abgebrochen, falls dieser in der Eingabe auftritt. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 8. Das Ergebnis eines XSLT-Stylesheets ist immer ein wohlgeformtes XML Dokument. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrekter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(9)

Betrachten Sie die folgende Schema-Datei **ns.xsd**:

```
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.dbai.tuwien.ac.at"
  xmlns:dbai="http://www.dbai.tuwien.ac.at">

  <xsd:element name="root" type="dbai:rootType">
    <xsd:key name="k1">
      <xsd:selector xpath="/node"/>
      <xsd:field xpath="@name"/>
    </xsd:key>
  </xsd:element>

  <xsd:complexType name="rootType">
    <xsd:sequence>
      <xsd:element name="node" maxOccurs="unbounded">
        <xsd:complexType>
          <xsd:attribute name="name" type="xsd:string"/>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Kreuzen Sie an, welche Aussagen bzgl. **ns.xsd** (bzw. für ein gültiges Instanzdokument desselben) wahr bzw. falsch sind.

1. Der Präfix **xsd** ist für XML Schema fix vorgegeben und darf nicht anders heißen. wahr ☐ falsch ☒
2. Im Instanzdokument liegen Elemente **root** und **node** im selben Namespace. wahr ☐ falsch ☒
3. Im Instanzdokument darf man statt **dbai** auch einen anderen Präfix verwenden, sofern dieser auf den richtigen Namespace zeigt. wahr ☒ falsch ☐
4. In der XML Schema Datei könnte man die Zeile `targetNamespace="http://www.dbai.tuwien.ac.at"` auch weglassen. wahr ☐ falsch ☒
5. Die Schlüsseldefinition ist ungültig, da kein Fremdschlüssel definiert wird. wahr ☐ falsch ☒
6. Der Pfad für den Schlüssel ist nicht korrekt, da der Präfix fehlt. wahr ☐ falsch ☒

Aufgabe 3:

(9)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

1. Um die Wohlgeformtheit eines XML-Dokuments zu überprüfen wird eine DTD oder ein XML-Schema benötigt. wahr ☐ falsch ☒
2. Die DTD Elementdeklaration $(A+|B+|C+)$ ist gleichwertig mit $(A|B|C)^+$. wahr ☐ falsch ☒
3. Die Ordnung der Knoten im Resultat eines XPath-Ausdrucks ist immer in Document Order. wahr ☐ falsch ☒
4. Ein XSLT-Stylesheet muss ein wohlgeformtes XML-Dokument sein. wahr ☒ falsch ☐
5. SAX verwendet die DOM API um ein XML Dokument zu lesen. wahr ☐ falsch ☒
6. SAX: Textevents können auch direkt hintereinander auftreten. wahr ☒ falsch ☐

Aufgabe 2:

(9)

Betrachten Sie die folgende Schema-Datei **ns.xsd**:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.dbai.tuwien.ac.at/education/ssd/SS11/pruefung/NS"
  xmlns:el="http://www.dbai.tuwien.ac.at/education/ssd/SS11/pruefung/NS"
  elementFormDefault="qualified">

  <xs:element name="edgelist">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="edge" type="el:edgeType" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:complexType name="edgeType">
    <xs:sequence>
      <xs:element name="point" type="el:pointtype" minOccurs="2" maxOccurs="2"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="pointtype">
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>

</xs:schema>
```

Kreuzen Sie an, welche Aussagen bzgl. **ns.xsd** (bzw. für ein gültiges Instanzdokument desselben) wahr bzw. falsch sind.

1. Würde man im XML Schema bei der Attributdefinition von **name** das Attribut **form** auf den Wert **qualified** setzen, dann muss im Instanzdokument bei diesem Attribut der **targetNamespace** referenziert werden. wahr ☒ falsch ☐
2. In dem XML Schema ist es über das Attribut **targetNamespace** möglich mehrere Target-NS für Instanzdokumente zu deklarieren. wahr ☐ falsch ☒
3. Im Instanzdokument wäre es möglich sowohl einen Präfix **el1** als auch einen weiteren Präfix **el2** zu deklarieren, welche beide auf den im XML Schema deklarierten **targetNamespace** zeigen. wahr ☒ falsch ☐
4. Im Instanzdokument zu dem XML Schema müssen Attribute immer im leeren NS liegen. wahr ☒ falsch ☐
5. Das Attribut **targetNamespace** dient nur der Dokumentation und kann in der XML Schema Datei auch weggelassen werden. wahr ☐ falsch ☒
6. Im Instanzdokument zu dem XML Schema ist es möglich, dass ein Element keinem Namespace zugeordnet ist. wahr ☐ falsch ☒

Aufgabe 3:

(9)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

1. Gültige XML-Dokumente sind immer auch wohlgeformte XML-Dokumente. wahr ☒ falsch ☐
2. Bei XML handelt es sich um eine Teilmenge der Meta-Markup-Sprache SGML. wahr ☒ falsch ☐
3. Bei einer Ableitung neuer Typen in einem XML Schema mittels 'Restriction', müssen alle Komponenten, die enthalten sein sollen, noch einmal explizit angegeben werden. wahr ☒ falsch ☐
4. DOM sieht Attribute als Kinder von Elementen an. wahr ☐ falsch ☒
5. In XPath beginnt ein relativer Pfad immer vom aktuellen Context Node aus wahr ☒ falsch ☐
6. XSLT ist auch ohne Kontrollstrukturen noch Turing-vollständig wahr ☒ falsch ☐

Aufgabe 2:

(12)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | | |
|--|---------------------------------------|---|
| 1. Wohlgeformte XML-Dokumente sind immer auch gültig. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 2. DOM sieht Attribute als Kinder von Elementen an. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 3. In XPath beginnt ein absoluter Pfad immer vom aktuellen Context Node aus | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 4. SAX erlaubt es, ein XML Dokument in beliebiger Reihenfolge zu durchlaufen. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 5. Die DTD Elementdeklaration $(A^* B^* C^*)$ ist gleichwertig mit $(A B C)^*$. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 6. Bei XSLT wird pro Knoten maximal ein Template angewendet. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 7. Attribute werden bei einem SAX Parser als eigene Events getriggert. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 8. Der "parent" eines SAX-Filter kann selbst auch ein SAX-Filter sein. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(12)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

1. XPath ist keine W3C-Recommendation, aber ein de-facto Standard. wahr ☐ falsch ☒
2. Die Ordnung der Knoten im Resultat eines XPath-Ausdrucks ist immer in Document Order. wahr ☐ falsch ☒
3. Der Speicherbedarf eines DOM-Parsers ist unabhängig von der Größe des geparsten XML-Files. wahr ☐ falsch ☒
4. Attributewerte werden von SAX Parsern ignoriert. wahr ☐ falsch ☒
5. Das Typsystem von XQuery basiert auf XML Schema. wahr ☒ falsch ☐
6. Die DTD Elementdeklaration (A?**|**B?**|**C?) ist gleichwertig mit (A**|**B**|**C)?. wahr ☒ falsch ☐
7. Rekursive Definitionen wie z.B. <!ELEMENT B (A,B?,C)> sind in DTDs erlaubt. wahr ☒ falsch ☐
8. SGML ist eine aktuelle Weiterentwicklung von XML. wahr ☐ falsch ☒

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(12)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

1. Der XPath-Ausdruck `//station` ist die Kurzschreibweise des XPath-Ausdrucks `/descendant::station` wahr ☐ falsch ☒
2. XQuery ist keine W3C-Recommendation, aber ein de-facto Standard. wahr ☐ falsch ☒
3. Eine XML Schema Definition muss ein wohlgeformtes XML-Dokument sein. wahr ☒ falsch ☐
4. Um die Wohlgeformtheit eines XML-Dokuments zu überprüfen wird eine DTD oder ein XML Schema benötigt. wahr ☐ falsch ☒
5. Bei SAX kann maximal ein Filter zwischen Reader und Applikation geschaltet werden. wahr ☐ falsch ☒
6. SAX Textevents treten niemals direkt hintereinander auf. wahr ☐ falsch ☒
7. Rekursive Definitionen wie z.B. `<!ELEMENT B (A,B?,C)>` sind in DTDs verboten. wahr ☐ falsch ☒
8. HTML ist eine Weiterentwicklung von XML. wahr ☐ falsch ☒

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(12)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | | |
|---|---------------------------------------|---|
| 1. Wird ein leeres XSLT-Stylsheet auf ein XML-Dokument angewandt, wird standard-mäßig ein Fehler geliefert. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 2. Das Resultat eines XPath Ausdrucks ist immer in Dokument-Order. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 3. XQuery und XPath 2.0 bauen auf demselben Datenmodell auf. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 4. SAX ist keine W3C-Recommendation, aber ein de-facto Standard. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 5. Ein XSLT Stylesheet muss ein wohlgeformtes XML-Dokument sein. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 6. SAX Filter dienen zur Umleitung der Ausgabe in eine Datei. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 7. HTML ist eine Teilmenge der Sprache XML. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 8. XML ist eine Teilmenge der Sprache HTML. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(15)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | | |
|---|---------------------------------------|---|
| 1. XML gibt es schon wesentlich länger als HTML. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 2. Um die Wohlgeformtheit eines XML-Dokuments zu überprüfen wird eine DTD oder ein XML Schema benötigt. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 3. Eine XML Schema Definition muss ein wohlgeformtes XML-Dokument sein. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 4. Pro XML-Schema Datei ist maximal ein Target-Namespace erlaubt. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 5. SAX ist keine W3C-Recommendation, aber ein de-facto Standard. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 6. SAX Filter dürfen nicht verschachtelt verwendet werden. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 7. Der Speicherbedarf eines DOM-Parsers ist unabhängig von der Größe des geparsen XML-Files. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 8. In XPath gilt $(1,2) = (2,3)$ | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 9. In XPath gilt $(1,2) \neq (2,3)$ | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 10. XSLT Stylesheets sind wohlgeformte XML Dokumente. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(15)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | | |
|---|---------------------------------------|---|
| 1. XML gibt es schon wesentlich länger als das WWW. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 2. Wohlgeformte XML-Dokumente sind immer auch gültig. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 3. Eine XML Schema Definition spezifiziert das Wurzelement des XML-Instanz Dokuments. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 4. Pro XML-Schema Datei ist maximal ein Target-Namespace erlaubt. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 5. DOM ist eine W3C-Recommendation. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 6. SAX Filter dürfen verschachtelt verwendet werden. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 7. Der Speicherbedarf eines SAX-Parsers ist im Allgemeinen geringer als eines DOM-Parsers | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 8. Bei XSLT wird pro Knoten maximal ein Template angewendet. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 9. Das Resultat eines XPATH Ausdrucks ist immer in Dokument-Order. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 10. XQUERY baut auf einem relationalem Datenmodell auf. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(15)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | |
|--|--|
| 1. Pro XML-Dokument ist maximal eine CDATA-Section erlaubt. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 2. Der Inhalt einer CDATA-Section wird als normaler Text an den Parser weitergeleitet, auch wenn darin Tags vorkommen. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 3. Um die Wohlgeformtheit eines XML-Dokuments zu überprüfen, muss eine DTD angegeben werden. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 4. Um die Wohlgeformtheit eines XML-Dokuments zu überprüfen, muss ein XML Schema angegeben werden. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 5. XSLT ist eine W3C-Recommendation. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 6. SAX Parser ignorieren Attributwerte. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 7. Der Speicherbedarf eines DOM-Parsers ist im Allgemeinen geringer als eines SAX-Parsers | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 8. In XPath gilt $(1,2) \neq (2,3)$ | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 9. Rekursive Definitionen wie z.B. <code><!ELEMENT B (A,B?,C)></code> sind in DTDs erlaubt. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 10. Der XPath-Ausdruck <code>//subpage</code> ist die Kurzschreibweise des XPath-Ausdrucks <code>/descendant::subpage</code> | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(15)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | |
|---|--|
| 1. XML gibt es schon wesentlich länger als das WWW. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 2. Die XML-Recommendation lässt offen, wie sich XML-Parser bei Wohlgeformtheitsfehler verhalten sollen. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 3. Wohlgeformte XML-Dokumente dürfen im Allgemeinen mehr als ein Wurzelement haben. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 4. Der Speicherbedarf eines SAX-Parsers ist im Allgemeinen geringer als eines DOM-Parsers. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 5. Das Resultat eines XPATH Ausdrucks ist immer in Dokument-Order. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 6. Bei SAX können auch mehrere characters events hintereinander auftreten. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 7. Rekursive Definitionen wie z.B. <code><!ELEMENT B (A,B?,C)></code> sind in DTDs verboten. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 8. Der XPath-Ausdruck <code>//name</code> ist die Kurzschreibweise des XPath-Ausdrucks <code>/descendant::name</code> | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 9. Syntaktisch ist JSON ein Teil von JavaScript. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 10. Das Konzept der “CDATA Sections” spielt in JSON eine wichtige Rolle. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(15)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | | |
|--|---------------------------------------|---|
| 1. XML steht für eXtended Modeling Language | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 2. Eine Eigenschaft von XML ist die Trennung von Struktur und Präsentation | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 3. Die Reihenfolge von Attributen ist entscheidend | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 4. Elemente ohne Namespace Präfix sind immer im leeren Namespace | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 5. DTD bieten keine Unterstützung von Namespaces | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 6. In einem XML-Schema spezifiziert xsd:sequence die Reihenfolge des Auftretens im Instanzdokument | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 7. Die Verwendung von complexType mit simpleContent macht keinen Sinn | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 8. Ein XML Parser prüft immer die Gültigkeit eines XML Dokuments | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 9. SAX Filter können nicht verschachtelt verwendet werden | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 10. Ein Absoluter XPath Pfad beginnt beim aktuellen Context Node | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(15)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | | |
|--|---------------------------------------|---|
| 1. HTML ist eine Weiterentwicklung von XML. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 2. Ein (bezüglich eines Schemas) gültiges XML-Dokument muss auch wohlgeformt sein. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 3. DTDs erlauben typischerweise eine exaktere Spezifikation als XML Schema. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 4. Pro XML-Schema Datei ist maximal ein Target-Namespace erlaubt. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 5. In XPath gilt $(a, b) \neq (b, c)$. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 6. SAX Filter dürfen verschachtelt verwendet werden. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 7. SAX-Parser benutzen intern einen DOM-Parser. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 8. Bei XSLT wird pro Knoten maximal ein Template angewendet. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 9. XQUERY ist eine W3C Recommendation. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 10. JSON ist eine XML-Applikation. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(15)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | | |
|---|---------------------------------------|---|
| 1. DTD steht für Data Type Definition. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 2. Für jedes XML Dokument gilt: pro Element sind maximal 2 Kind-Elemente erlaubt. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 3. XML Schemas erlauben typischerweise eine exaktere Spezifikation als DTDs. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 4. In einem XML-Schema kann <code>xsd:attribute</code> ein Kind-Element von <code>xsd:simpleType</code> sein. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 5. Der XPath Ausdruck “ <code>../../* eq .</code> ” liefert in jedem Knoten (außer der Wurzel) <i>wahr</i> . | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 6. Das DOM ist eine W3C Recommendation. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 7. Bei einem Push Parser wird das XML-Dokument mehrmals durchlaufen. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 8. Bei XSLT werden (mittels XPath) Knoten ausgewählt, auf die ein Template angewendet wird. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 9. XSLT ist hauptsächlich eine imperative Programmiersprache. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 10. XQuery erlaubt keine rekursiven Funktionsdeklarationen. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Aufgabe 2:

(15)

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

- | | | |
|--|---------------------------------------|---|
| 1. Das “X” in XML steht für eXtensible. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 2. Bei XML befindet sich immer Struktur und Präsentation in einem Dokument. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 3. Das Wurzelement eines XML Dokuments darf keine Attribute haben. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 4. Jedes wohlgeformte XML-Dokument ist auch gültig. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 5. In einem XML-Schema haben Attribute einen beliebigen <code>xsd:simpleType</code> als Datentyp. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 6. Der XPath Ausdruck “ <code>./../* eq .</code> ” liefert in jedem Knoten (außer der Wurzel) <i>falsch</i> . | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 7. XSLT ist eine W3C Recommendation. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 8. Der Speicherbedarf eines SAX-Parsers ist im Allgemeinen höher als eines DOM-Parsers. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 9. Im <code>match</code> -Attribut eines <code><xsl:template></code> -Elements eines XSLT-Stylesheets dürfen beliebige XPath-Ausdrücke verwendet werden. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 10. Ein absoluter XPath-Pfad beginnt beim Wurzelement. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |

(Pro korrekter Antwort 1.5 Punkte, **pro inkorrektter Antwort -1.5 Punkte**, pro nicht beantworteter Frage 0 Punkte, für die gesamte Aufgabe mindestens 0 Punkte)

Exercise 2:

(15)

Decide which of the following statements is true or false.

- | | | |
|---|---------------------------------------|--|
| 1. Semi-structured data is a flexible format for exchanging data. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 2. The “M” in XML stands for model. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 3. XML can be used for storing digitized data such as photos. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 4. An element in an XML document cannot be empty. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 5. Namespaces can be used for disambiguating elements and attributes. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 6. Validating errors cannot be ignored. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 7. DTDs are more powerful than XML schemas. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 8. Event-based parsers use a constant amount of memory. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 9. XPath is a schema language. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 10. XSLT documents are XML documents. | true <input checked="" type="radio"/> | false <input type="radio"/> |

(For every correct answer 1.5 points, **for every incorrect answer -1.5 points**, for every unanswered question 0 points, you can have at least 0 points on this exercise)

Exercise 2:

(15)

Decide which of the following statements is true or false.

- | | | |
|---|---------------------------------------|--|
| 1. Semi-structured data is a special case of structured data. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 2. XML is a markup language. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 3. XML documents are plain text. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 4. XML documents can be executable files. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 5. Validating errors can be ignored. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 6. The “T” in DTD stands for transformation. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 7. DTDs are XML documents. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 8. XML Schemas are more powerful than DTDs. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 9. XPath is a query language. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 10. XPath is more powerful than XSLT. | true <input type="radio"/> | false <input checked="" type="radio"/> |

(For every correct answer 1.5 points, **for every incorrect answer -1.5 points**, for every unanswered question 0 points, you can have at least 0 points on this exercise)

Exercise 2:

(15)

Decide which of the following statements is true or false.

- | | |
|--|---|
| 1. A relational table can be represented as an XML document. | true <input checked="" type="radio"/> false <input type="radio"/> |
| 2. XML is a presentation language like HTML. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 3. XML documents are stored as plain text. | true <input checked="" type="radio"/> false <input type="radio"/> |
| 4. Every element in an XML document, including the root element, has exactly one parent. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 5. DTDs are written in XML syntax. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 6. In XML schema a complex element cannot have attributes. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 7. Tree-based parsers are faster than event-based parsers. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 8. In XPath “@” is the abbreviation for “ <code>attribute::</code> ” | true <input checked="" type="radio"/> false <input type="radio"/> |
| 9. Every XQuery expression can be written as an XPath expression. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 10. The “T” in XSLT stands for Template. | true <input type="radio"/> false <input checked="" type="radio"/> |

(For every correct answer 1.5 points, **for every incorrect answer -1.5 points**, for every unanswered question 0 points, you can have at least 0 points on this exercise)

Exercise 2:

(15)

Decide which of the following statements is true or false.

- | | | |
|---|---------------------------------------|--|
| 1. In semi-structured data entities in the same class have the same attributes. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 2. An XML document can be seen as a graph. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 3. XML is a W3C standard. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 4. The “X” in XML stands for Extensible. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 5. DTDs can be used for querying XML documents. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 6. An XML Schema is not an XML document. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 7. DOM is a tree-based API for manipulating XML documents. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 8. XPath is a language for extracting parts of an XML document. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 9. Every XPath expression is an XQuery expression. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 10. XSLT is more powerful than XQuery. | true <input type="radio"/> | false <input checked="" type="radio"/> |

(For every correct answer 1.5 points, **for every incorrect answer -1.5 points**, for every unanswered question 0 points, you can have at least 0 points on this exercise)

Exercise 2:

(15)

Decide which of the following statements are true or false.

- | | | |
|--|---------------------------------------|--|
| 1. Structured data can be represented as a graph. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 2. The “X” in XML stands for eXchangeable. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 3. An XML document is not a database. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 4. An XML document must be well-formed. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 5. DTDs are not XML documents. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 6. Validating errors cannot be ignored. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 7. DTDs are more expressive than XML schemas. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 8. Tree-based parsers use a constant amount of memory. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 9. XPath is a query language. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 10. XPath is more powerful than XSLT. | true <input type="radio"/> | false <input checked="" type="radio"/> |

(For every correct answer 1.5 points, **for every incorrect answer -1.5 points**, for every unanswered question 0 points, you can have at least 0 points on this exercise)

Exercise 2:

(15)

Decide which of the following statements are true or false.

- | | | |
|---|---------------------------------------|--|
| 1. To check whether an XML document is well-formed, a DTD or an XML-Schema is required. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 2. The “X” in XML stands for eXecutable. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 3. In XML-Schema attributes have an arbitrary <code>xsd:simpleType</code> as data type. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 4. DTD declaration $(A+ B+ C+)$ is not equivalent to $(A B C)+$. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 5. XML-Schema documents are not XML documents. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 6. XSLT is a W3C recommendation. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 7. XPath is used to modify XML documents. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 8. Event-based parsers use a constant amount of memory. | true <input checked="" type="radio"/> | false <input type="radio"/> |
| 9. XQuery is a schema language. | true <input type="radio"/> | false <input checked="" type="radio"/> |
| 10. The XPath test “ <code>../.* eq .</code> ” returns in every element (except the root element) <i>true</i> . | true <input type="radio"/> | false <input checked="" type="radio"/> |

(For every correct answer 1.5 points, **for every incorrect answer -1.5 points**, for every unanswered question 0 points, you can have at least 0 points on this exercise)

Exercise 2:

(15)

Decide which of the following statements are true or false.

- | | |
|--|---|
| 1. Semi-structured data can be represented as labelled trees. | true <input checked="" type="radio"/> false <input type="radio"/> |
| 2. XML can be used as programming language and as network protocol. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 3. DTDs are written in XML syntax. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 4. A relational table can be represented as an XML document. | true <input checked="" type="radio"/> false <input type="radio"/> |
| 5. Every XQuery expression can be written as an XPath expression. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 6. DTDs are more powerful than XML schemas. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 7. XPath is more powerful than XSLT. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 8. SAX is a tree-based API for manipulating XML documents. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 9. Tree-based parsers are faster than event-based parsers. | true <input type="radio"/> false <input checked="" type="radio"/> |
| 10. Namespaces can be used for disambiguating elements and attributes. | true <input checked="" type="radio"/> false <input type="radio"/> |

(For every correct answer 1.5 points, **for every incorrect answer -1.5 points**, for every unanswered question 0 points, you can have at least 0 points on this exercise)

Aufgabe 2:

(15)

Entscheiden Sie, ob die folgenden Aussagen wahr oder falsch sind.

- | | | |
|---|---------------------------------------|---|
| 1. Das übliche Datenmodell für semi-strukturierte Daten sind vollständige Binärbäume. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 2. XML ist eine Einschränkung von HTML. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 3. Sowohl XML Schemas als auch DTDs sind in XML Syntax geschrieben. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 4. XPath ist Teil des XQuery W3C Standards. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 5. XPath ist Teil des XSLT W3C Standards. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 6. XML wird häufig als Netzwerk Protokol verwendet. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 7. Nicht jedes wohlgeformte XML Dokument ist auch valide. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 8. DOM ist eine Baum-basierte API um XML Dokumente zu manipulieren. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 9. Namespace URIs sind immer gültige XML Namen. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 10. Attribute ohne Prefix sind im Default Namespace. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |

(Für jede korrekte Antwort 1.5 Punkte, **für jede falsche Antwort -1.5 Punkte**, unbeantwortete Fragen 0 Punkt, Insgesamt nicht weniger als 0 Punkte)

Aufgabe 2:

(15)

Entscheiden Sie, ob die folgenden Aussagen wahr oder falsch sind.

- | | | |
|--|---------------------------------------|---|
| 1. DOM und SAX sind Standards für Event-Based XML-Parser. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 2. DTDs sind immer wohlgeformte XML Dokumente. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 3. DTDs und XML Schema sind gleichmächtig. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 4. Die DOM API kommt nur bei XHTML Dokumenten zum Einsatz. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 5. Strukturierte Daten sind ein Spezialfall von semi-strukturierten Daten. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 6. XML Dokumente sind das einzige Format um semi-strukturierten Daten zu speichern. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 7. Event-based Parser benötigen nur konstant viel Speicher,
d.h. der Speicherbedarf ist unabhängig von der Größe des geparsten XML Dokuments. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 8. Das übliche Datenmodell für semi-strukturierte Daten sind Bäume. | wahr <input checked="" type="radio"/> | falsch <input type="radio"/> |
| 9. Namespace URIs sind immer gültige XML Namen. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |
| 10. Nur Elemente ohne Prefix sind im Default Namespace. | wahr <input type="radio"/> | falsch <input checked="" type="radio"/> |

(Für jede korrekte Antwort 1.5 Punkte, **für jede falsche Antwort -1.5 Punkte**, unbeantwortete Fragen 0 Punkt, Insgesamt nicht weniger als 0 Punkte)

Aufgabe 2:

(15)

Entscheiden Sie, ob die folgenden Aussagen wahr oder falsch sind.

- | | |
|--|--|
| 1. Event-based Parser benötigen nur konstant viel Speicher,
d.h. der Speicherbedarf ist unabhängig von der Größe des geparsten XML Dokuments. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 2. SAX ist ein Standard für Tree-Based XML-Parser. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 3. XPath ist Teil der XQuery Sprache. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 4. XSLT ist Teil der XPath Sprache. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 5. Das Wurzelement eines XML Dokuments darf keine Attribute haben. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 6. Namespace URIs sind immer gültige XML Namen. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 7. Attribute ohne Prefix sind im Default Namespace. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 8. Im <code>match</code> -Attribut eines <code><xsl:template></code> -Elements eines XSLT-Stylesheets dürfen beliebige XPath-Ausdrücke verwendet werden. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 9. DTD steht für Data Type Definition. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 10. XML Schemas erlauben typischerweise eine exaktere Spezifikation als DTDs. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |

(Für jede korrekte Antwort 1.5 Punkte, **für jede falsche Antwort -1.5 Punkte**, unbeantwortete Fragen 0 Punkt, Insgesamt nicht weniger als 0 Punkte)

Aufgabe 2:

(15)

Entscheiden Sie, ob die folgenden Aussagen wahr oder falsch sind.

- | | |
|--|--|
| 1. In einem wohlgeformten XML-Dokument dürfen zwei Attribute mit demselben qualifizierten Namen in einem Elementstarttag auftreten | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 2. Genauso wie bei Elementen ist die Reihenfolge von Attributen innerhalb eines Elements in XML-Dokumenten signifikant. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 3. DOM erlaubt wahlfreien Zugriff auf das gesamte XML Dokument. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 4. Frage entfernt (da nicht eindeutig) | |
| 5. In XPath beginnt ein relativer Pfad immer vom aktuellen Context Node aus. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 6. Wenn in XSLT für einen Knoten kein Template angegeben wird, dann wird die Verarbeitung abgebrochen, falls dieser in der Eingabe auftritt. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 7. Das Ergebnis eines XSLT-Stylesheets ist immer ein wohlgeformtes XML Dokument. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 8. DTD steht für Document Type Definition. | wahr <input checked="" type="radio"/> falsch <input type="radio"/> |
| 9. Um die Wohlgeformtheit eines XML-Dokuments zu überprüfen wird eine DTD oder ein XML-Schema benötigt. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |
| 10. Attribute ohne Prefix sind im Default Namespace. | wahr <input type="radio"/> falsch <input checked="" type="radio"/> |

(Für jede korrekte Antwort 1.5 Punkte, **für jede falsche Antwort -1.5 Punkte**, unbeantwortete Fragen 0 Punkt, Insgesamt nicht weniger als 0 Punkte)

Aufgabe 2:

(15)

Beantworten Sie, die folgenden Fragen kurz und bündig (Für jede korrekte Antwort 1.5 Punkte).

1. Welches Datenmodell verwenden wir für semistrukturierte Daten?

Antwort: Bäume mit Labels auf den Kanten.

2. Geben Sie einen Formalismus (ein Format) an um semistrukturierte Daten zu speichern:

Antwort: Object Exchange Model (OEM); JavaScript Object Notation (JSON); eXtensible Markup Language (XML)

3. Wofür steht die Abkürzung DTD?

Antwort: Document Type Definition

4. Wie verhält sich der Speicherbedarf von Event-based Parsern wie SAX zur Größe des geparsten XML Dokuments?

Antwort: Der Speicherbedarf ist unabhängig von der Größe des geparsten XML Dokuments.

5. Welche in der Vorlesung behandelte API erlaubt wahlfreien Zugriff auf das gesamte XML Dokument?

Antwort: DOM

6. Welche der in der Vorlesung behandelten Sprachen enthalten XPath?

Antwort: XQuery, XSLT, (XML-Schema (enthält Teile von XPath))

7. Wir haben verschiedene Sprachen um Schemata zu definieren kennen gelernt. Welche hat die größte Ausdrucksstärke?

Antwort: XML Schema (XSD)

8. Zu welchem Zweck werden XML Namespaces verwendet?

Antwort: Um Elemente/Attribute mit gleichem Namen aus unterschiedlichen Quellen und/oder unterschiedlicher Semantik zu unterscheiden. Um Elemente/Attribute zu gruppieren.

9. Wieso können URIs nicht direkt als prefix für XML Elemente/Attribute verwendet werden?

Antwort: Die resultierenden Elementnamen sind keine gültigen XML-Namen.

10. Was macht das Default-Template in XSLT für Attribute?

Antwort: Es gibt den Wert des Attributs aus.

Aufgabe 2:

(15)

Beantworten Sie, die folgenden Fragen kurz und bündig (Für jede korrekte Antwort 1.5 Punkte).

1. Wie verhält sich der Speicherbedarf von Tree-based Parsern wie DOM zur Größe des geparsen XML Dokuments?

Antwort: Der Speicherbedarf steigt linear mit der Größe des geparsen XML Dokuments.

2. Welche Art von Parsern, neben Tree-based Parsern, haben wir in der VO noch kennengelernt?

Antwort: Event-based Parser, SAX.

3. Was macht das Default-Template in XSLT für Elemente?

Antwort: Es wendet die Templates der Kinder an.

4. Wie geht XSLT mit der Situation um wenn mehrere Templates auf ein Element matchen?

Antwort: Es wird nur eines ausgeführt, dabei wird das spezifischste Template ausgewählt.

5. Welches Datenmodell verwenden wir für semistrukturierte Daten?

Antwort: Bäume mit Labels auf den Kanten.

6. Was versteht man unter FLWOR Expressions?

Antwort: Die typische Struktur einer XQuery: For Let Where Order by Return

7. Beschreiben Sie kurz den Unterschied zwischen wohl-geformten (well-formed) XML-Dokumenten und validen XML Dokumenten.

Antwort: Wohl-geformte XML Dokumente entsprechen dem XML-Standard und valide Dokumente erfüllen zusätzlich noch die Bedingungen eines Schema das z.B. als DTD oder XML-Schema gegeben ist.

8. Erläutern Sie kurz die Einschränkungen von DTDs bei der Definition von mehreren Schlüsseln für ein Dokument.

Antwort: Die Werte der Schlüsselattribute müssen eindeutig über alle Schlüssel sein.

9. Auf welche Teile eines XML Dokuments wird der Default Namespace angewendet?

Antwort: Auf Elemente ohne Prefix.

10. Geben Sie einen der signifikanten Unterschiede zwischen HTML und XML an.

Antwort: z.b.: Fix vorgegebene Menge von Elementen vs. frei definierbare Menge von Elementen

Aufgabe 2:

(15)

Beantworten Sie, die folgenden Fragen kurz und bündig (Für jede korrekte Antwort 1.5 Punkte).

1. Erläutern Sie kurz die Einschränkungen von DTDs bei der Definition von Fremdschlüsseln.

Antwort: Es kann nicht definiert auf welches Schlüsselattribut sich der Fremdschlüssel bezieht.

2. Welche Sprachen um Schemata zu definieren haben wir in der Vorlesung kennengelernt?

Antwort: DTDs, XML Schema

3. Welche in der Vorlesung behandelte API hat nur konstanten Speicherbedarf?

Antwort: SAX

4. Zu welchem Zweck werden Namespaces verwendet?

Antwort: Um Elemente/Attribute mit gleichem Namen aus unterschiedlichen Quellen zu unterscheiden. Um Elemente/Attribute zu gruppieren.

5. Wie unterscheiden sich Elementen von Attributen bezüglich der Signifikanz der Reihenfolge im XML Dokument?

Antwort: Die Reihenfolge ist nur für Elemente nicht aber für Attribute signifikant.

6. Geben Sie drei der möglichen Achsen in XPath an.

Antwort: ancestor, ancestor-or-self, attribute, child, descendant, descendant-or-self, following, following-sibling, namespace, parent, preceding, preceding-sibling, self

7. Was macht das Default-Template in XSLT für Attribute?

Antwort: Es gibt den Wert des Attributs aus.

8. Welches Datenmodell verwenden wir für semistrukturierte Daten?

Antwort: Bäume mit Labels auf den Kanten.

9. Wieso können URIs nicht direkt als prefix für XML Elemente/Attribute verwendet werden?

Antwort: Die resultierenden Elementnamen sind keine gültigen XML-Namen.

10. Erläutern Sie kurz den Unterschied zwischen for und let statements in XQuery.

Antwort: Mit for wird jedes Element einer Sequenz einzeln an die Variable gebunden (es wird über die Elemente der Sequenz iteriert). Mit let wird die vollständige Sequenz auf einmal an die Variable gebunden.

Aufgabe 2:

(15)

Beantworten Sie, die folgenden Fragen kurz und bündig (Für jede korrekte Antwort 1.5 Punkte).

1. Geben Sie zwei Formalismen (zwei Formate) an um semistrukturierte Daten zu speichern:

Antwort: Object Exchange Model (OEM), JavaScript Object Notation (JSON), eXtensible Markup Language (XML)

2. Wozu benötigt man die in XML vordefinieren entity references?

Antwort: Um die reservierten Symbole ("<" und "&") innerhalb von Elementen zu verwenden

3. Geben Sie ein Beispiel für eine Document Type Declaration.

Antwort: <!DOCTYPE smarthome SYSTEM "smarthome.dtd">

4. Geben Sie 5 der möglichen Attribut Typen in DTDs an.

Antwort: CDATA, NMTOKEN, NMTOKENS, Enumeration, ID, IDREF, IDREFS, ENTITY, ENTITIES, NOTATION

5. Welche Einschränkung bezüglich der Achsen gibt es für XPath Ausdrücke in XML-Schema Definitionen?

Antwort: Es darf nur die Kind (und zu Beginn die descendant-or-self) Achse verwendet werden.

6. Was macht das Default-Template in XSLT für Attribute?

Antwort: Es gibt den Wert des Attributs aus.

7. Definieren sie kurz die vier verschiedenen Arten von Inhalt (content) die ein xml-Element haben kann. (6)

Antwort: Empty- das Element hat leeren Inhalt; Simple - das Element enthält nur Text; Element content- das Element enthält nur andere Elemente; Mixed - das Element enthält sowohl Text als auch Elemente

Aufgabe 2:

(15)

Beantworten Sie, die folgenden Fragen kurz und bündig (Für jede korrekte Antwort 1.5 Punkte).

1. Erläutern Sie kurz die Einschränkungen von DTDs bei der Definition von Fremdschlüsseln.

Antwort: Es kann nicht definiert werden auf welches Schlüsselattribut sich der Fremdschlüssel bezieht.

2. Zu welchem Zweck werden Namespaces verwendet?

Antwort: Um Elemente/Attribute mit gleichem Namen aus unterschiedlichen Quellen zu unterscheiden. Um Elemente/Attribute zu gruppieren.

3. Geben Sie drei der möglichen Achsen in XPath an.

Antwort: ancestor, ancestor-or-self, attribute, child, descendant, descendant-or-self, following, following-sibling, namespace, parent, preceding, preceding-sibling, self

4. Welche Art von Parsern, neben Tree-based Parsern, haben wir in der VO noch kennengelernt?

Antwort: Event-based Parser, SAX.

5. Was versteht man unter FLWOR Expressions?

Antwort: Die typische Struktur einer XQuery: For Let Where Order by Return

6. Geben Sie ein Beispiel für eine Document Type Declaration.

Antwort:

7. Definieren sie kurz die vier verschiedenen Arten von Inhalt (content) die ein xml-Element haben kann. (6)

Antwort:

Aufgabe 2:

(15)

Beantworten Sie, die folgenden Fragen kurz und bündig (Für jede korrekte Antwort 1.5 Punkte).

1. Welche Sprachen um Schemata zu definieren haben wir in der Vorlesung kennengelernt?

Antwort: DTDs, XML Schema

2. Wie unterscheiden sich Elemente von Attributen bezüglich der Signifikanz der Reihenfolge im XML Dokument?

Antwort: Die Reihenfolge ist nur für Elemente nicht aber für Attribute signifikant.

3. Welches Datenmodell verwenden wir für semistrukturierte Daten?

Antwort: Bäume mit Labels auf den Kanten.

4. Erläutern Sie kurz den Unterschied zwischen let und for statements in XQuery.

Antwort: Mit for wird jedes Element einer Sequenz einzeln an die Variable gebunden (es wird über die Elemente der Sequenz iteriert). Mit let wird die vollständige Sequenz auf einmal an die Variable gebunden.

5. Was macht das Default-Template in XSLT für Elemente?

Antwort: Es wendet die Templates der Kinder an.

6. Wie geht XSLT mit der Situation um wenn mehrere Templates auf ein Element matchen?

Antwort: Es wird nur eines ausgeführt, dabei wird das spezifischste Template ausgewählt.

7. Beschreiben Sie kurz den Unterschied zwischen wohl-geformten (well-formed) XML-Dokumenten und validen XML Dokumenten.

Antwort: Wohl-geformte XML Dokumente entsprechen dem XML-Standard und valide Dokumente erfüllen zusätzlich noch die Bedingungen eines Schema das z.B. als DTD oder XML-Schema gegeben ist.

8. Geben Sie einen der signifikanten Unterschiede zwischen HTML und XML an.

Antwort: z.b.: Fix vorgegebene Menge von Elemente vs. frei definierbare Menge von Elemente

9. Welche in der Vorlesung behandelte API erlaubt wahlfreien Zugriff auf das gesamte XML Dokument?

Antwort: DOM

10. Welche in der Vorlesung behandelte API hat nur konstanten Speicherbedarf?

Antwort: SAX

Aufgabe 2:

(15)

Beantworten Sie, die folgenden Fragen kurz und bündig (Für jede korrekte Antwort 1.5 Punkte).

1. Erläutern Sie kurz den Unterschied zwischen `let` und `for` statements in XQuery.

Antwort: Mit `for` wird jedes Element einer Sequenz einzeln an die Variable gebunden (es wird über die Elemente der Sequenz iteriert). Mit `let` wird die vollständige Sequenz auf einmal an die Variable gebunden.

2. Wie verhält sich der Speicherbedarf von Tree-based Parsern wie DOM zur Größe des geparsen XML Dokuments?

Antwort: Der Speicherbedarf steigt linear mit der Größe des geparsen XML Dokuments.

3. Was macht das Default-Template in XSLT für Elemente?

Antwort: Es wendet die Templates der Kinder an.

4. Wie geht XSLT mit der Situation um wenn mehrere Templates auf ein Element matchen?

Antwort: Es wird nur eines ausgeführt, dabei wird das spezifischste Template ausgewählt.

5. Welches Datenmodell verwenden wir für semistrukturierte Daten?

Antwort: Bäume mit Labels auf den Kanten.

6. Auf welche Teile eines XML Dokuments wird der Default Namespace angewendet?

Antwort: Auf Elemente ohne Prefix.

7. Geben Sie einen Formalismus (ein Format) an um semistrukturierte Daten zu speichern:

Antwort: Object Exchange Model (OEM), JavaScript Object Notation (JSON), eXtensible Markup Language (XML)

8. Wir haben verschiedene Sprachen um Schemata zu definieren kennen gelernt. Welche hat die größte Ausdrucksstärke?

Antwort: XML Schema

9. Zu welchem Zweck werden Namespaces verwendet?

Antwort: Um Elemente/Attribute mit gleichem Namen aus unterschiedlichen Quellen zu unterscheiden. Um Elemente/Attribute zu gruppieren.

10. Wieso können URIs nicht direkt als prefix für XML Elemente/Attribute verwendet werden?

Antwort: Die resultierenden Elementnamen sind keine gültigen XML-Namen.

Aufgabe 2:

(15)

Beantworten Sie, die folgenden Fragen kurz und bündig (Für jede korrekte Antwort 1.5 Punkte).

1. Welche Sprachen um Schemata zu definieren haben wir in der Vorlesung kennengelernt?

Antwort: DTDs, XML Schema

2. Zu welchem Zweck werden Namespaces verwendet?

Antwort: Um Elemente/Attribute mit gleichem Namen aus unterschiedlichen Quellen zu unterscheiden. Um Elemente/Attribute zu gruppieren.

3. Wie unterscheiden sich Elemente von Attributen bezüglich der Signifikanz der Reihenfolge im XML Dokument?

Antwort: Die Reihenfolge ist nur für Elemente nicht aber für Attribute signifikant.

4. Welche Art von Parsern, neben Tree-based Parsern, haben wir in der VO noch kennengelernt?

Antwort: Event-based Parser, SAX.

5. Was macht das Default-Template in XSLT für Elemente?

Antwort: Es wendet die Templates der Kinder an.

6. Wie geht XSLT mit der Situation um wenn mehrere Templates auf ein Element matchen?

Antwort: Es wird nur eines ausgeführt, dabei wird das spezifischste Template ausgewählt.

7. Beschreiben Sie kurz den Unterschied zwischen wohl-geformten (well-formed) XML-Dokumenten und validen XML Dokumenten.

Antwort: Wohl-geformte XML Dokumente entsprechen dem XML-Standard und valide Dokumente erfüllen zusätzlich noch die Bedingungen eines Schema das z.B. als DTD oder XML-Schema gegeben ist.

8. Erläutern Sie kurz die Einschränkungen von DTDs bei der Definition von mehreren Schlüsseln für ein Dokument.

Antwort: Die Werte der Schlüsselattribute müssen eindeutig über alle Schlüssel sein.

9. Geben Sie einen der signifikanten Unterschiede zwischen HTML und XML an.

Antwort: z.B.: Fix vorgegebene Menge von Elemente vs. frei definierbare Menge von Elemente

10. Welche in der Vorlesung behandelte API erlaubt wahlfreien Zugriff auf das gesamte XML Dokument?

Antwort: DOM