



Fachprüfung: Hyper- und Multimedia

Hypermedia- und Multimediasysteme (1,2)

16.07.2003

Name (bitte in Blockschrift)	
Matrikelnummer	
Unterschrift	

Hinweise:

- Überprüfen Sie Ihr Klausur-Exemplar bitte vor Beginn der Klausur auf Vollständigkeit.
- Bitte halten Sie Ihren Lichtbildausweis sowie den Studentenausweis zur Kontrolle bereit.
- Dauer der Klausur: 120 min.
- maximal erreichbare Punktzahl: 200
- Bitte füllen Sie das Deckblatt vollständig aus, beschriften jedes Blatt mit Ihrer Matrikelnummer und unterschreiben Sie dieses Klausur-Exemplar.
- Jedes Verlassen des Prüfungsraums muss ausdrücklich mit der Aufsicht vereinbart werden.
- Zugelassene Hilfsmittel sind ausschließlich Schreibutensilien, nicht-programmierbare Taschenrechner und das eigene(!) Gedächtnis.
- Bitte vermeiden Sie die Verwendung von roter Farbe.
- Die nach jeder Frage eingeklammerte Zahl ist die bei dieser Frage maximal erreichbare Punktzahl.
- Beachten Sie die in vielen Fragen enthaltenen Teilfragen!
- Falls der Platz für die Beantwortung einer Frage nicht ausreichen sollte, verwenden Sie bitte die Rückseite.
- Nutzen Sie im Falle von Unklarheiten hinsichtlich der Fragestellung die Möglichkeit zu Rückfragen!

Viel Erfolg!

Punkte Teil 1	
Punkte Teil 2	
Punkte gesamt	

1. Prüfer Note
2. Prüfer	



– I. Prüfungsfragen zur Veranstaltung im SS 2002 (HMS-1) –

1. Nennen und charakterisieren Sie kurz 3 Scriptsprachen für HTML. Gehen Sie dabei besonders darauf ein, wo die jeweilige Sprache ausgeführt wird. (12)

- JavaScript (+)
 - Client-seitig (+), Objektorientiert, C++/Java-ähnliche Syntax, implementiert DOM, Netscape-Produkt, ... (++)
- JScript (+)
 - Server-seitige (+) Version von JavaScript, MS-Produkt (++)
- VBScript (+)
 - Client-seitig (+), Visual Basic Script, MS-Produkt, nicht auf allen Browsern implementiert, keine Groß-/Klein-Unterscheidung (++)

oder ...

- ASP (mit VBScript) (+)
 - Server-seitig (+), Active Server Pages, MS-Produkt, MS-spezifisch (++)
- PHP (server-seitig, plattformunabh., ...), Perl (server-seitig, OO, Mix aus C++ und Unix Shell Script, ...) (je +/+/++)

2. Wie werden JavaScript Events (z.B. OnClick) in HTML verarbeitet? Warum bietet sich gerade diese Methode aus HTML-Sicht dafür an? (4)

JavaScript Events können jedem Tag im DOM zugewiesen werden; (++)
Bubbling äußert sich dergestalt, dass ein Event durch die gesamte Hierarchie aufsteigt, d.h. ein Event im innersten Tag aktiviert ggf. auch einen umschließenden Tag. (+) HTML verarbeitet Events auf jeder hierarchischen Ebene. (+)

3. Welche (vier) verschiedenen Möglichkeiten bestehen zur Einbettung von JavaScript in HTML-Dokumente. Zeigen Sie Beispiele. (12)

- über eigene Datei: (+)
`<script LANGUAGE="JavaScript" src="JSdatei.js" type="text/javascript"></script>` (++)
- als Markup im Script-Tag: (+)
`<script LANGUAGE="JavaScript"> function brmpf() { // code } </script>` (++)
- in beliebigem Tag: (+)
` Sag was` (++)
- als Entity in regulärem HTML-Tag: (+)
`<hr width=&{breite};>` (++)
- oder auch: in JavaScript: (+)
`<script language="JavaScript"> breite = 10; </script>` (++)

4. Wie werden in JavaScript mehrdimensionale Arrays erzeugt? (6)

1. Erzeugung eines eindimensionalen Arrays über Konstruktor:
`new Array()` (+++)
2. Funktionsdefinition: `function ...() {} ;` (+++)
enthält geschachtelte for-Schleifen mit jeweils neuen Konstruktor-Aufrufen

5. Wie werden PHP-Kommandos in HTML-Dokumente eingebunden? (3)

als eigenes Tag: `<?php ... ?>` oder auch nur `<? ... ?>` (+++)
(oder: in ASP: `<%@language="PHP"%> <% ... %>` (+++))



6. Welche Besonderheiten unterscheiden Java Applets von Java Applikationen? (12)

- URLs lesen, keine Dateien (++)
- Netzwerkverbindungen nur zum Ursprung (++)
- public Methoden anderer Applets nutzen, keine nativen Methoden (++)
- keine main() Methode (++)
- kein Laden von Bibliotheken (++)
- Steuerung über Browser (++)
- ...

7. Erläutern Sie knapp die Prinzipien der Prioritätenvergabe und der Synchronisation beim „Multithreading“ in Java-Programmen. (6)

- **Priorität:** CPU-Zuteilung an gleichzeitig ablaufende Threads; gegenseitige Beeinflussung der Thread-Prioritäten möglich. (+++)
- **Synchronisation:** gleichzeitiger Zugriff auf gemeinsame (veränderliche) Speicherbereiche/Variablen. (+++)

8. Beschreiben Sie zwei grundsätzlich verschiedene Möglichkeiten, Threads in Java zu programmieren. Gehen Sie dabei auch besonders auf die Unterschiede ein. (10)

- Thread-Klasse: (+)**
- eigene Klasse extends Thread(++)
 - run() Methode wird durch start() aufgerufen (++)
- Runnable-Interface: (+)**
- eigene Klasse implements Runnable(++)
 - start() Methode muss programmiert werden (++)



9. Wie und mit welchen Aufrufen kann ein Java-Thread sich selbst und andere Threads zur Laufzeit beeinflussen? (12)

- sich selbst in den Wartezustand versetzen (+): `sleep()` (+)
 - anderen Thread `t` starten (+): `t.start()` (+)
 - sich selbst CPU entziehen (+): `yield()` (+)
 - sich selbst und anderen (`t`) (+) anhalten (+): `[t.]suspend()` (+)
 - eigene und fremde (`t`) (+) Priorität ändern (+): `[t.]setPriority()` (+)
- oder ...
- sich selbst und anderen (`t`) (+) beenden (+): `[t.]stop()` (+)

10. Nennen Sie drei verschiedene Java Layout Manager und beschreiben Sie kurz deren Eigenschaften. (9)

- `FlowLayout` (+) - Anordnung der Elemente zuerst horizontal, dann vertikal (++)
 - `BorderLayout` (+) - Anordnung der Elemente nach Ortsangabe [`Center`, `North`, `West`, `East`, `South`] (++)
 - `GridLayout` (+) - Anordnung der Elemente nach Reihenfolge des Auftretens innerhalb eines vorgegebenen Gitters mit gleichmäßig großen Zellen (++)
- oder ...
- `CardLayout` (+) - Anordnung der Elemente hintereinander in z-Ebene mit Karteireitern (++)
 - `GridBagLayout` (+) - Anordnung der Elemente innerhalb eines frei konfigurierbaren Gitters (++)
 - `BoxLayout` (+) - Anordnung der Elemente nach Reihenfolge des Auftretens untereinander (++)



11. Beschreiben Sie das Konzept der keyframe Animation („tweening“) in Macromedia Director oder Flash. Welche Eigenschaften lassen sich auf diese Weise animieren? (9)

- Einzelne Bilder werden als Sprites in bestimmtem Zustand bestimmten Zeitpunkten (frames) zugeordnet, Zu denen sie Schlüsselzustände annehmen (--> keyframes). (++)
- Für die zwischen den Keyframes liegenden Einstellungen werden die jeweiligen Zustände (für jeden frame) linear interpoliert und so eine kontinuierliche Veränderung erreicht. (++)
- Folgende Eigenschaften von Sprites können „getweent“ werden: Position, Orientierung, Größe, Form, Farbe, ... (+++++)

12. Worin unterscheiden sich die Ausgabeformate von Macromedia Director? (5)

- .dir - editierbares Projekt (+)
- .exe - ausführbare (Windows-)Datei (+)
- .swf - Shockwave-Datei, hochkomprimiert für Internetanwendung; nur über Plugin im Browser lauffähig (+++)



- II. Prüfungsfragen zur Veranstaltung im SS 2003 (HMS-2) -

1. Was sind „DataIslands“, wie und in welchem Kontext werden sie verwendet? Wie kann auf „DataIslands“ zugegriffen werden? (10)

Mit Hilfe von DataIslands können XML-Daten direkt innerhalb einer HTML-Seite enthalten sein. Der Vorteil ist dabei, dass die Daten dem Client zur ständigen Verfügung stehen, sobald die betreffende Seite im Browser geladen ist. Eine mögliche Anwendung ist zum Beispiel das Verwalten von Tabelleninhalten, die auf Anforderung hin umsortiert werden sollen ohne dass die Seite oder die Informationen in der Tabelle erneut vom Server geladen werden sollen.

Eingebunden werden können DataIslands u.a. wie folgt:

```
<XML ID="XMLID">
  <XMLDATA>
    <DATA>TEXT</DATA>
  </XMLDATA>
</XML>
```

oder

```
<XML SRC="http://localhost/xmlFile.xml"></XML>
```

oder

```
<SCRIPT LANGUAGE="XML" SRC="http://localhost/xmlFile.xml"></SCRIPT>
```

oder

```
<SCRIPT ID="XMLID" LANGUAGE="XML">
  <XMLDATA>
    <DATA>TEXT</DATA>
  </XMLDATA>
</SCRIPT>
```

Der Zugriff geschieht dann ganz normal mit JavaScript, z.B.:

```
function returnXMLData()
{
  return document.all("XMLID").XMLDocument.nodeValue;
}
```

```
function returnXMLData()
{
  return XMLID.documentElement.text;
}
```

2. In DHTML gibt es die Properties „innerHTML“, „outerHTML“, „innerText“ und „outerText“. Diese können lesend und schreibend verwendet werden. Beschreiben Sie die jeweiligen Unterschiede anhand des gedachten Beispiels „Das Fernsehprogramm des WDR“ und einer Zuweisung „<i>Neuer Text</i>“. (10)

- Lesen
 - innerHTML : Das Fernsehprogramm des WDR
 - outerHTML: Das Fernsehprogramm des WDR
 - innerText und outerText: Das Fernsehprogramm des WDR
- Schreiben
 - innerHTML: <i>Neuer Text</i>
 - outerHTML: <i>Neuer Text</i>
 - innerText: Neuer Text
 - outerText: Neuer Text

3. Wie lassen sich innerhalb von JavaScript eigene Objekte erstellen? Erstellen Sie ein Beispiel. (5)

```
function Farbe(Farbwert_R, Farbwert_G, Farbwert_B)
{
  this.Farbwert_R = Farbwert_R;
  this.Farbwert_G = Farbwert_G;
  this.Farbwert_B = Farbwert_B;
}

function TestObjekt()
{
  Test = new Farbe("33","99","C0");
  alert("Der Rotwert meiner Farbe ist Hexadezimal " +
  Test.Farbwert_R);
}
```




4. Wie lassen sich im MS Internet Explorer HTML-Elemente zoomen? Nennen Sie auch mögliche Maßeinheiten. (5)

Über die CSS-Eigenschaft "style.zoom=...".

Der Wert kann in Prozent oder als Fließkommazahl angegeben werden (100% entsprechen dabei 1.0).

5. Nennen Sie typische Bestandteile von DHTML und deren Funktion. (10)

Das Dokumentenobjektmodell (DOM) beschreibt den hierarchischen Aufbau der Elemente einer HTML-Seite, auf die (und deren Eigenschaften und Methoden) mit Hilfe von Scriptsprachen (dynamic content) wie JavaScript und Cascading Style Sheets (CSS) (dynamic styles) zur Laufzeit lesend und schreibend zugegriffen werden kann. Events schließlich dienen der Möglichkeit, auf Benutzeraktionen (button click, scrolling, ...) mit Hilfe des Scriptings reagieren zu können.

6. Auf welche (4) verschiedenen Arten können Cascading Stylesheets (CSS) in HTML Seiten verwendet werden? Lassen sich die unterschiedlichen Arbeitsweisen in Bezug auf ein HTML-Element kombinieren? (10)

Inline über das Attribut Style auf Ebene des einzelnen Elementes, Embedded im Style-Tag im Headbereich, External über das Link-Tag als Verweis auf eine separate CSS-Datei oder per Scripting. Die genannten Formen lassen sich mischen bzw. kombinieren. Dabei gilt folgende Priorität in absteigender Reihenfolge bei sich widersprechenden Anweisungen: Scripting > Inline > Embedded > External.



7. Was ist der Unterschied zwischen den CSS-Anweisungen „`style.display=none`“ und „`style.visibility=hidden`“? (5)

Beide Anweisungen führen dazu, dass das entsprechende Element nicht auf der Seite angezeigt wird mit dem Unterschied, dass bei „`display=none`“ der nachfolgende HTML-Streaminhalt neu positioniert wird, während bei „`visibility=hidden`“ keine Neupositionierung der übrigen Elemente erfolgt.

8. Was sind die wesentlichen Unterschiede zwischen XML-Schema und DTD's? Wofür werden sie verwendet? (5)

Schemas werden selbst auch in XML geschrieben, Element- und Attributinhalt lassen sich näher typisieren. Das Contentmodell lässt sich gezielter, aber auch flexibler beschreiben.

9. Gegeben sei folgender Ausschnitt eines HTML-Dokumentes: „`<i><U>Das ist Text mit einem
 Zeilenumbruch</i></u>`“. Ist das Beispiel auch ein XML-Dokument? Begründen Sie Ihre Entscheidung und schreiben Sie das Beispiel ggf. „richtig“ um. (10)

Der Ausschnitt ist nicht wellformed. Für XHTML gilt u. a.: case-sensitive, schließende Tags, geordnetes Nesting, Attributwerte in Anführungszeichen, Script- und Styleblöcke in CDATA-Abschnitten.

Richtig wäre:

```
<a href="wdr.de" id="test"><i><u>Das ist Text mit einem <br/>
Zeilenumbruch</u></i></a>
```



10. Wie lassen sich Datenbankabfragen mit ADO weiterverarbeiten? (10)

Die Ergebnisse von SQL-Abfragen über ADO sind zunächst in einer temporären Tabelle enthalten („recordset“). Die Inhalte können in einer („for-while“) Schleife für die einzelnen Datensätze sowie einer Schleife über die Feldinhalte ausgelesen und weiterverarbeitet werden. Der Inhalt kann aber auch als XML-Stream („recordset.Save(xml, adPersistXML“) oder als String („Variant = recordset.GetString(StringFormat, NumRows, ColumnDelimiter, RowDelimiter, NullExpr“) ausgegeben werden.

11. Beschreiben Sie die Funktion von Xpath und die wichtigsten Anwendungsprinzipien. (10)

Xpath ist eine Abfragesyntax, mit deren Hilfe sich Elemente eines XML-Dokumentes genau lokalisieren lassen. Xpath ist damit das SQL für XML. Ein wesentlicher Bestandteil ist das „/“-Zeichen, mit dem ein Knotenpunkt symbolisiert wird. Der Platzhalter „.“ bezeichnet den gerade gültigen Knoten. „./“ kennzeichnet somit Knoten die dem gerade gültigen in der nächsten Hierarchiestufe innerhalb des DOM's folgen. „//“ bezeichnet Knoten, die vom aktuellen ausgehend beliebig tief verschachtelt liegen dürfen. Mit „@“ werden Attribute gekennzeichnet. In eckigen Klammern stehende Ausdrücke dienen der Filterung nach bestimmten Kriterien. Diese können auch logische Vergleichsoperatoren enthalten.

Die auch als „Patterns“ bezeichneten Abfrage-Strings können in XSLT oder in bestimmten DOM-Methoden („selectNodes (.AUSDRUCK“), „selectSingleNode (.AUSDRUCK“)“) des XML-Parsers Verwendung finden.



12. Beschreiben Sie, wie Sie wiederholte Datenbankzugriffe eines Anwenders auf einem Webservice optimieren können. (10)

- Nach erstmaligem Öffnen der Datenbank das `Connection-Object` in einer `Session-Variablen` festhalten.
- Mit `RemoteScripting` überflüssige Seiten-Reloads vermeiden
- Mit `xmlHttpRequest` überflüssige Seiten-Reloads vermeiden
- Ggf. Daten als `DataIslands` an den Client schicken

Punkte Teil 2	/ 100
---------------	-------

Viel Erfolg!