

Erstellung interaktiver Lernobjekte mit HTML 5

Diplomarbeit an der
Technischen Universität Dresden
März 2015

Hendrik Neubert

Betreuer: Dr. rer. nat. Holger Rohland
Hochschullehrer: Prof. Dr. paed. habil. Steffen Friedrich
AG Didaktik der Informatik/Lehrerbildung
Institut für Software- und Multimediatechnik
Fakultät Informatik



Aufgabenstellung für die Diplomarbeit

Name, Vorname des Studenten: Neubert, Hendrik
Immatrikulationsnummer: 3485700

Thema: "Erstellung interaktiver Lernobjekte mit HTML 5"

Zielstellung:

Im Rahmen des Projekts „Hochschulübergreifende Angebote für den Studieneinstieg zur Verminderung der Abbrecherquote im Bereich Informatik“ sind für verschiedene Lernbereiche interaktive Lernobjekte mit HTML 5 zu erstellen. Ziel der Arbeit soll es sein, einen Leitfaden zur Herstellung solcher Objekte zu erstellen, anhand dessen auch Anfänger einfache interaktive Lernobjekte realisieren können.

In einem ersten Teil der Arbeit sollen im Sinne einer "best practice" - Analyse frei verfügbare, webbasierte Animationen, Simulationen u.ä. mit unterschiedlichem Interaktionsgrad aus den Bereichen Programmierung, Datenbanken und Theoretische Informatik recherchiert werden. Die Ergebnisse sind in einer kategorisierten und bewerteten Liste zusammenzufassen.

Im zweiten Teil der Arbeit ist der oben genannte Leitfaden zu erstellen. Auf dessen Basis sind abschließend mindestens drei eigene interaktive Lernobjekte mit unterschiedlichem Interaktionsgrad aus dem Themenbereich der Theoretischen Informatik zu erstellen.

Betreuer: Dr. rer. nat. Holger Rohland
Betreuender Hochschullehrer: Prof. Dr. paed. habil. Steffen Friedrich
Institut: Software- und Multimediatechnik
Beginn am: 01.10.2014
Einzureichen am: 31.03.2015
Unterschrift

Erklärung

Hiermit erkläre ich, Hendrik Neubert, die vorliegende Diplomarbeit zum Thema

„Erstellung interaktiver Lernobjekte mit HTML 5“

selbständig und ausschließlich unter Verwendung der im Quellenverzeichnis aufgeführten Literatur- und sonstigen Informationsquellen verfasst zu haben.

Dresden, am 31.03.2015

Unterschrift

Inhaltsverzeichnis

1	Einleitung	1
1.1	Ziel der Arbeit.....	2
2	Analyse von frei verfügbaren webbasierten Lernobjekten.....	4
2.1	Auswahl der Bewertungskriterien.....	4
2.1.1	Grundsätze der Dialoggestaltung	4
2.1.2	Gestaltungsgrundsätze und Rahmenbedingungen.....	8
2.1.3	Interaktionsstufen – Taxonomie.....	9
2.1.4	Lizenzmodell.....	10
2.1.5	Bewertungsskala.....	11
2.2	Lernobjekte für die Theoretische Informatik	12
2.2.1	Formale Sprachen und Grammatiken.....	12
2.2.2	Die Turingmaschine	16
2.2.3	Der Kellerautomat	20
2.2.4	Endliche Automaten.....	22
2.3	Lernobjekte für die Programmierung.....	25
2.3.1	Lineare Datenstrukturen	26
2.3.2	Graphen	28
2.3.3	Bäume.....	32
2.3.4	Sortierverfahren.....	36
3	Ein Leitfaden für das Erstellen interaktiver Lernobjekte mit HTML5	42
3.1	Werkzeuge und Grundlagen für das Erstellen von dynamischen Webinhalten.....	42
3.1.1	Grundlagen	42
3.1.2	Werkzeuge und Tools für die Implementierung	44
3.1.3	Nützliche Frameworks	45
3.2	Visuelle Darstellungen erstellen	46
3.2.1	Visuelle Komponenten mit <svg> kreieren	46
3.2.2	Das <canvas>-Element.....	52
3.2.3	Visualisieren mit D3.js.....	55
3.3	DOM-Elemente dynamisch erzeugen und modifizieren.....	59
3.3.1	DOM-Elemente selektieren.....	59
3.3.2	Attribute erzeugen und verändern	61
3.3.3	DOM-Manipulation mit JavaScript und jQuery.....	63
3.4	Interaktions- und Navigationselemente erstellen und verwenden	66

3.4.1	Formulare erstellen und Eingaben verwenden	66
3.4.2	Direkte Manipulation	75
3.4.3	Möglichkeiten mit jQuery UI	78
3.5	Animationen und dynamische Abläufe generieren	79
3.5.1	Animationen erzeugen	80
3.5.2	Simulationen und Abläufe generieren	84
3.5.3	Beispiel zur Umsetzung einer Simulationssteuerung	86
4	Umsetzung eines eigenen Prototyps	91
4.1	Idee und Zielsetzung	91
4.2	Funktionalität	91
4.2.1	Die Eingabe	92
4.2.2	Das Transformationsmenü	93
4.2.3	Die Simulationsansicht	93
4.3	Implementierung	94
4.3.1	Transformationsalgorithmen	94
4.3.2	Visualisierung	95
4.3.3	Simulation	96
4.3.4	Benutzeroberfläche	97
4.4	Erweiterungsmöglichkeiten	98
5	Zusammenfassung und Ausblick	100
	Glossar	104
	Literaturverzeichnis	106
	Abbildungsverzeichnis	109
	Tabellenverzeichnis	110
	Rechtliche Hinweise für die Benutzung der evaluierten Anwendungen ...	111

1 Einleitung

Das World Wide Web ist nicht erst seit gestern, das wohl beliebteste Medium der Gegenwart. Im digitalen Zeitalter ist es vor allen Dingen die jüngere Generation, die das Internet alltäglich benutzt. Insbesondere wird es als Kommunikations- und Informationsplattform verwendet und zeichnet sich durch Aktualität, Vielseitigkeit und Unabhängigkeit aus. Auch für das selbstständige Lernen bietet das Web, gerade durch die in den letzten Jahren entwickelten Technologien, neue und innovative Möglichkeiten. Seit der Einführung des Web 2.0 und der Cloud-Speicherung sind neue Lernszenarien im e-Learning-Bereich, wie beispielsweise das „Kollaborative Lernen“ für Gruppen, realisierbar. Für den einzelnen Lernenden lassen sich Unmengen von multimedialen Lerninhalten im Netz finden, mit denen er sich selbstständig mit dem Lerngegenstand befassen kann. Neben wissenschaftlichen Artikeln, Lernvideos und e-Learning-Kursen sind es vor allem Lernprogramme, mit denen sich der Lerner interaktiv einer Problemstellung widmen kann.

Online-Angebote bieten besonders aufgrund ihrer Verfügbarkeit, Unabhängigkeit und mittlerweile auch der technischen Möglichkeiten großes Potenzial. Die Verfügbarkeit der Lernmedien verspricht aber nicht automatisch Erfolg. Die Inhalte können durch ihre Präsentation zu Verunsicherung und Unverständlichkeit beitragen, was auch mit dem unterschiedlichen Bildungsstand der Benutzer zusammenhängt. Da die Medien auch häufig ausschließlich vom Lernenden genutzt werden und ihm demzufolge kein Tutor zur Verfügung steht, müssen die Inhalte an den Ansprüchen und Belangen des Benutzers angepasst werden können und somit geeignete Unterstützungsmethoden bereitgestellt werden. Durch das selbstständige Lernen fehlt auch oft eine Motivationsstütze. *„E-Learning Instrumente sollen [daher] in erster Linie ein attraktives Lernumfeld schaffen, das zur Motivation des Lernenden beiträgt.“* [El] Zudem müssen insbesondere komplexe Applikationen benutzbar und strukturiert sein und sollten sich an den üblichen Konventionen ausrichten. Unzählige Aspekte spielen also bei der Gestaltung und Umsetzung der Lerninhalte eine wichtige Rolle.

Der Begriff „Lernobjekt“

„Lernobjekte sind kleinste in sich sinnvolle Lerngegenstände: ein Bild, eine Videosequenz, eine Testaufgabe, ein Lehrtext, ein Simulationsprogramm.“ [Ni08, S.144]

Diese Definition beschreibt sehr gut, aus welcher Sicht ein Lernobjekt in dieser Arbeit betrachtet werden soll, nämlich als eine eigenständige Präsentationsform eines Lerninhalts oder Lerngegenstands.

Ursprünglich wurde der Begriff „Lernobjekt“ erschaffen, um die Wiederverwendbarkeit von Content in Lernkursen zu ermöglichen. Dabei sollte eine Trennung des Kurses in einzelne Lernobjekte vollzogen werden, um somit einzelne Kurselemente effizient zu speichern und diese mühelos zwischen Lernplattformen austauschen zu können. [Pa05] Dafür wurde im Jahr 2002 der Standard „Learning Object Metadata“ (LOM) vom Institute of Electrical and Electronics Engineers verabschiedet, um die Metadaten zur Beschreibung von Lernobjekten festzulegen. Denn Lernobjekte sollen als autonome Komponenten verstanden werden und auch so weitergegeben werden. Um diese aber auch weiterverwenden zu können, werden wichtige Informationen, welche das Lernobjekt betreffen, wie beispielsweise Lernziele, Vorkenntnisse und andere Rahmenbedingungen in den Metadaten angegeben.

Die Betrachtung dieser Metadaten spielt aber in dieser Arbeit weder bei der Beurteilung noch bei der Implementierung eine Rolle. Die Lernobjekte sollen hier als eigenständige Anwendungen verstanden werden, die dem Lernenden bei der Erreichung des Lernziels unterstützen und weiterbringen. Die Eignung des Lernobjekts für den Nutzer steht somit im

Vordergrund und nicht der ursprüngliche Gedanke der Wiederverwendung. Zudem wird der Begriff des Lernobjekts immer im Zusammenhang mit interaktiven Anwendungen und Präsentationen verwendet. Mediale Formen wie Grafiken, Texte, Animationen oder Videos, insofern sie nicht Teilkomponenten des Lernobjekts sind, sondern eigenständig, werden in der Arbeit vernachlässigt.

Warum HTML5?

Die Entwicklung einer Anwendung hängt zuallererst von den technischen Voraussetzungen ab, die dem Entwickler und dem Nutzer zur Verfügung stehen. Allerdings sollten diese die Realisierung der Applikation nicht übermäßig stark beeinflussen, so dass zu Beginn des Entwicklungsprozesses eine geeignete Technologie gefunden werden muss.

Bisher wurden auf Webseiten fehlende Funktionen mit eingebetteten Java- und Flash-Applets nachgerüstet, um Simulationen oder Spiele zu ermöglichen. Damit diese vom Nutzer verwendet werden können, wird jedoch ein zusätzliches Plug-In benötigt. Flash-Anwendungen haben zudem den Nachteil, dass diese nur mit Flash-Editoren bearbeitet werden können und für Suchmaschinen nur schwer lesbar sind. [Ha12] Zudem konnten auch Videosequenzen bisher nur über ein Plug-In eingebunden werden. Mit den neuen Funktionen, die HTML5 bietet, kann diese Lücke geschlossen werden, in dem dynamische und interaktive Inhalte erstellt und direkt in das Objektmodell des HTML-Dokuments integriert werden und daher kein zusätzliches Plug-In benötigt wird. Da gerade multimediale Inhalte den Mittelpunkt einer Webseite darstellen, ist es besonders wichtig den Zugriff auf diesen Content sicherzustellen.

Desweiteren ist HTML5 unabhängig von Betriebssystem und Plattform des Endgerätes, so dass auch möglichst unabhängig und kostengünstig entwickelt werden kann und die Lernobjekte problemlos wiederverwendet werden können.

HTML5 ist die fünfte Version der Hypertext Markup Language und wurde am 28. Oktober 2014 vom World Wide Web Consortium (W3C) spezifiziert. Sie ist damit der Nachfolger von HTML4, die als „Kernsprache“ des Internets bezeichnet wurde. [Wi6]

1.1 Ziel der Arbeit

Ziel der Arbeit soll es sein, einen Leitfaden anzufertigen, in dem das Erstellen interaktiver Lernobjekte mit HTML5 beschrieben wird, so dass anhand des Inhalts Einsteiger, die Möglichkeit erhalten, Lernobjekte selbstständig zu implementieren. Dabei kann natürlich nur auf einzelne Bereiche eingegangen werden, da die Vielfalt der Möglichkeiten, die sich hierbei anbieten den Rahmen der Arbeit bei Weitem sprengen würde. Der Leitfaden wird dabei in verschiedene Teilkapitel unterteilt, die sich jeweils mit der Umsetzung einer bestimmten Problemstellung beschäftigen. Zu dieser Problemstellung werden mehrere Lösungsansätze vorgestellt und deren Implementierung anhand von Beispielen präsentiert. Der Inhalt des Leitfadens beginnt dabei nicht bei den Grundlagen von HTML und CSS, sondern setzt Erfahrungen mit beiden Technologien voraus. In diesem Zusammenhang sollen eher neue Funktionalitäten, die sich durch HTML5 anbieten, in den Vordergrund rücken. Den Mittelpunkt stellt aber vor allem die Skriptsprache JavaScript dar, mit der erst Interaktivität ermöglicht werden kann. Neben den standardmäßigen JavaScript-Methoden werden auch Frameworks wie D3.js oder jQuery vorgestellt, die besonders für die Visualisierung und Manipulation von Kontext- und Dialogobjekten effizient verwendet werden können und deren Realisierung vorantreiben. Mit den im Leitfaden vorgestellten Technologien und Vorgehensweisen sollen dann als Abschluss der Arbeit eigene Lernobjekte implementiert werden. Der Lerngegenstand soll sich dabei mit einem Teilgebiet der Theoretischen Informatik befassen.

Doch bevor die Umsetzung des Leitfadens erfolgen soll, wird zunächst eine „Best-Practice“-Analyse von frei verfügbaren Webanwendungen vorgenommen. „Best-Practice“ wird auch als Erfolgsmethode bezeichnet, in der Produkte (in dem Fall Web-Anwendungen) die auf eine bestimmte Art und Weise in Verbindung stehen, gegenübergestellt werden, um die bestmögliche Lösung zu identifizieren. Durch die Evaluierung dieser Applikationen können wichtige Erfahrungen im Hinblick auf die eigene Implementierung gewonnen werden. Hierbei wird die Analyse nicht nur auf „echte“ Webanwendungen beschränkt, solche also die mit einer Webtechnologie, wie JavaScript oder PHP umgesetzt wurden. Sondern auch Java- und Flash-Applets, die über das Web zur Verfügung gestellt werden, sollen in die Untersuchung mit einbezogen werden. Die Auswertung der Anwendungen wird in Bewertungslisten abschließend übersichtlich dargestellt. Die Auswahl der Bewertungskriterien wird dabei im Kapitel erläutert und begründet.

5 Zusammenfassung und Ausblick

Die Analyse der Lernobjekte für die Bereiche „Theoretische Informatik“ und „Grundlagen der Programmierung“ hat gezeigt, dass für die ausgewählten Themenschwerpunkte genügend Material vorhanden ist. Dabei musste aber festgestellt werden, dass viele Anwendungen sehr überaltert sind und nicht mehr dem neuesten technischen Stand entsprechen. So können viele bereitgestellte Java-Applets aufgrund von Sicherheitsrisiken nicht mehr mit der aktuellen Laufzeitumgebung ausgeführt werden. Mit einer älteren Version, sind die Applikationen zwar benutzbar, diese Vorgehensweise aber eher umständlich ist.

Im deutschen Raum wurden bis auf wenige Ausnahmen fast ausschließlich Anwendungen solcher Art gefunden, die häufig im Rahmen einer studentischen Arbeit an einer universitären Einrichtung entwickelt und im Netz zur Verfügung gestellt wurden, teilweise ohne ausreichende Hinweise zur Weiternutzung.

Außerdem wurden auch häufig Applets in einem Lernkurs oder einem interaktiven Skript integriert. Diese zeichnen sich häufig durch eine hohe Qualität aus und sorgen im Zusammenspiel mit dem bereitgestellten Content für eine umfangreiche und anschauliche Darstellung des Lerninhalts.

Die größten Probleme, bei der Analyse, konnten in den Bereichen Fehlertoleranz und Individualisierbarkeit festgestellt werden. Die Anwendungen decken bei der Fehlererkennung häufig zu wenige Szenarien ab, so dass fehlerhafte Eingaben nicht entdeckt werden und dadurch undefinierte Zustände und Systemabbrüche die Folge sind. Auch erkannte Fehler werden häufig nicht ausreichend erklärt. Besonders für die Nutzereingabe werden syntaktische Regeln nicht hinreichend, weder in der Fehlermeldung noch in der Anwendung selbst, erläutert. Durch die mangelnde Bereitstellung von Einstellungsoptionen und der Beschränkung auf vorgegebene Werte, wird zudem keine ausreichende Individualisierbarkeit ermöglicht. So dass eine Anpassung der Anwendungsoberfläche an die zu bearbeitende Aufgabe nicht stattfinden kann.

Insgesamt konnte aber der Großteil der evaluierten Anwendungen bei der Analyse gute Ergebnisse erzielen. Vor allem aktuelle Applikationen zeichnen sich nicht nur durch eine innovative und moderne Benutzeroberfläche aus, sondern sorgen auch mit zahlreichen Hilfsfunktionen und Darstellungsoptionen für eine individuelle Anpassbarkeit und somit für eine geeignete Lernumgebung.

Die Entwicklung der Lernobjekte mit HTML5 führt auch in den meisten Fällen dazu, dass keine Trennung zwischen den Inhalt und der Anwendung mehr stattfindet, sondern diese ineinander übergehen. Dadurch können interaktive Lernkurse besser umgesetzt werden, der Trend geht auch in diese Richtung. Diese Zusammenführung von Anwendung und Inhalt macht auf jeden Fall Sinn, widerspricht damit aber der ursprünglichen Idee des Lernobjekts.

Bei der Bereitstellung von Java-Applets werden meist Hilfeseiten, Benutzeranleitungen und weitere Materialien auf andere Seiten ausgelagert. Durch das Aufrufen dieser Inhaltsseiten und dem damit verbundenen Verlassen der Anwendung, gehen Veränderungen, die durch Nutzerhandlungen innerhalb des Applets getätigt wurden, verloren. Bei HTML5-Anwendungen werden hingegen diese Informationen zusammen mit der Anwendung auf eine Seite dargestellt und können bei Bedarf auch ausgeblendet werden. Durch die dynamische Veränderung der visuellen Darstellung kann somit die Oberfläche weiter an die Ansprüche des Lernenden angepasst werden.

Die Evaluation der Lernobjekte führte dazu, dass ein Eindruck gewonnen werden könnte, welche Aspekte bei der Entwicklung eines Lernprogramms relevant sind. Vor allem konnten aber auch allgemeine Probleme festgestellt werden, die vermieden werden sollten und von kreativen Lösungsansätzen profitiert werden. Diese Erfahrungen flossen sowohl in die

Entwicklung des Prototypen mit ein, als auch in den Leitfaden, der auf diese Schwerpunkte ausgerichtet werden sollte.

Beim Erstellen des Leitfadens konnte festgestellt werden, dass bereits zahlreiche Inhalte im Web und in der Literatur zu HTML5 zur Verfügung stehen. Die Inhalte bilden meist die gesamte Komplexität der möglichen Funktionalität ab, so dass der Suchvorgang nach bestimmten Sachverhalten aufwendig sein kann. Im Leitfaden wurde daher speziell auf relevante Aspekte für die Implementierung eines Lernobjekts eingegangen und diese übersichtlich strukturiert, so dass ein guter Einstieg in dieses Thema ermöglicht wird.

Nach einem Einleitungsteil wurde der Leitfaden in vier weitere Abschnitte untergliedert. Dabei sind die Abschnitte so ausgerichtet, dass die Inhalte und Beispiele aufeinander aufbauen. Mit Verweisen auf bereits erläuterte Sachverhalte, sollte der unmittelbare Einstieg in ein beliebiges Kapitel keine weiteren Probleme darstellen. Zudem wurde darauf geachtet, dass auch immer alternative Lösungsvorschläge und Vorgehensweisen präsentiert werden. Diese sollen zum einen die Vielfalt der Möglichkeiten darstellen, zum anderen die Sachverhalte verdeutlichen, in dem diese aus mehreren Sichtweisen beleuchtet werden. Dabei werden auch immer wieder Frameworks in die Erläuterung mit einbezogen, die für einige Problemstellungen komfortable Methoden und Strategien zur Verfügung stellen und dessen Verwendung die Implementierung grundsätzlich vereinfachen kann.

Die präsentierten Codebeispiele sollen vor allem die theoretischen Erklärungen veranschaulichen und diesen somit einen praktischen Rahmen verleihen. Viele Beispiele gehen dabei aus der Implementierung des Prototyps hervor und geben damit einen Einblick, wie bestimmte Bestandteile umgesetzt wurden.

Der Prototyp geht als Produkt aus der Analyse der Lernobjekte und der Erstellung des Leitfadens hervor. Er zeigt, welche Möglichkeiten sich mit der Benutzung von HTML5 anbieten und wie damit eine benutzerfreundliche Lernanwendung umgesetzt werden kann. Die Usability der Anwendung wurde dabei durch die Ergebnisse der Analyse und damit auch durch den Einbezug der Bewertungskriterien beeinflusst. Die Erfüllung der Grundsätze soll dabei an einigen Beispielen verdeutlicht werden:

- **Aufgabenangemessenheit:** Anzeige von ausschließlich relevanten Informationen bezüglich des Dialogschritts
- **Selbstbeschreibungsfähigkeit:** Offensichtliche Interaktion durch beschriftete Bedienelemente und zur Verfügung stehende Benutzungshinweise
- **Erwartungskonformität:** Dialogverhalten und Informationsdarstellung sind innerhalb ähnlicher Aufgaben konsistent
- **Steuerbarkeit:** Eingaben können nach getätigten Umwandlungen wiederhergestellt werden
- **Fehlertoleranz:** Detaillierte Fehlerbeschreibung bei fehlerhafter Eingabe
- **Individualisierbarkeit:** Individuelle Benennung der Kontextobjekte möglich, sowohl alphabetische als auch numerische Symbole verwendbar
- **Lernförderlichkeit:** Vorgegebene Beispiele verfügbar, um einen schnellen Einstieg zu ermöglichen
- **Eignung für das Kommunikationsziel:** Relevante Informationen werden während eines Simulationsschritts hervorgehoben

- **Eignung für Wahrnehmung und Verständnis:** Vermeidung von Informationsüberlastung durch zeitabhängige Darstellungen mit Hilfe der Simulationssteuerung
- **Eignung für die Exploration:** Unterschiedliche Medienperspektiven während der Simulation (Tabelle, Visualisierung des Diagramms, textuelle Beschreibung des Algorithmus)
- **Eignung für die Benutzungsmotivation:** Direkte Interaktion

Dabei sind die aufgezählten Aspekte nur einige Beispiele, die zur Erfüllung der jeweiligen Kriterien beitragen. Die Benutzerfreundlichkeit der Anwendung kann sicherlich noch gesteigert werden. Erweiterungsmöglichkeiten wurden dabei präsentiert und ihr Nutzen ausführlich beschrieben.

Da die Aufgabenstellung ursprünglich darauf abzielte mehrere Lernobjekte mit jeweils unterschiedlichem Interaktionsgrad zu implementieren, diese aber zu einem Prototyp zusammengefasst wurden, sollte die Anwendung auch über diverse Interaktionsvarianten verfügen. Diese wird durch die Bedienung von Navigations- und Steuerelementen, Eingabemöglichkeiten und direkte Interaktion mit den Kontextobjekten sichergestellt.

Der Prototyp wurde mit HTML5 und den damit verbundenen Technologien CSS3 und JavaScript implementiert und wird in einem beliebigen Browser ausgeführt. Bei der Umsetzung wurde darauf geachtet, dass die Anwendung in allen üblichen Browsern fehlerfrei funktioniert, Voraussetzung ist dafür die aktuelle Version des jeweiligen Webbrowsers. Die JavaScript-Bibliotheken *jQuery* und *d3.js* wurden für die Visualisierung und Manipulation der Dialog- und Kontextelemente verwendet und sind nur zu empfehlen.

In der Arbeit konnte mit Hilfe des Leitfadens und mit der Umsetzung des Prototyps gezeigt werden, dass die Entwicklung eines Lernobjekts für das Web aus technischer Sicht ohne größere Einschränkungen realisierbar ist. Schon jetzt werden immer mehr Applikationen für das Web konzipiert. Da aufgrund der hohen Reichweite und Verfügbarkeit, diese eine viel größere Masse erreichen können. Zudem können durch die Vernetzung zu anderen Lernteilnehmern neue Lernmodelle entwickelt und umgesetzt werden. Kommerzielle Lernprogramme und -tools tummeln sich schon jetzt zu Genüge im Netz und versprechen dem Nutzer optimale Ergebnisse.

Bei den frei verfügbaren Angeboten befinden sich die reinen Web-Applikationen allerdings noch in der absoluten Minderheit. Eher werden ältere Anwendungen, die nicht mehr den neuesten Standard entsprechen kostenfrei zur Verfügung gestellt. Die Entwicklung von Web-Anwendungen auch für e-Learning-Content schreitet jedoch immer weiter voran und daher ist es nur eine Frage der Zeit, bis auch hier die Vielfalt der Angebote zunimmt.

Glossar

AJAX	Abkürzung für „ <i>Asynchronous JavaScript and XML</i> “. Ein Konzept der asynchronen Datenübertragung zwischen einem Browser und dem Server, welches es ermöglicht, Inhalte auf einer HTML-Seite zu verändern, ohne diese neu zu laden. [Wi7]
CC-Lizenz	Abkürzung für „ <i>Creative-Commons</i> “-Lizenz. Eine Open-Source-Lizenz, für die Verbreitung und Vervielfältigung eines Werks.
CSS	Abkürzung für Cascading Style Sheets. Eine Beschreibungssprache für Web-Dokumente.
Debugger	Ein Werkzeug zum Diagnostizieren von Fehlern in Computersystemen.
DFA	Abkürzung für „ <i>Deterministic Finite Automaton</i> “, (deutsch: Deterministischer Endlicher Automat). Ein endlicher Automat, der von einem Zustand, in dem er sich befindet, in einen eindeutig bestimmten Folgezustand wechselt. [Wi9]
Drag & Drop	Eine grafische Bedienfunktion, bei der Objekte durch einen Mausklick ausgewählt und bei gedrückter Maustaste innerhalb der grafischen Benutzeroberfläche verschoben werden können. [It4]
DOM	Abkürzung für „ <i>Document Object Model</i> “. Eine vom W3C-Consortium definierte Spezifikation einer Schnittstelle für den Zugriff auf HTML- oder XML-Dokumente. [It1]
GPL	Abkürzung für „ <i>GNU General Public Licence</i> “. Eine Open-Source-Lizenz für die freie Nutzung einer Software. Die Verbreitung und Vervielfältigung ist unter Verwendung der gleichen Lizenzbedingungen gestattet.
HTML	Abkürzung für „ <i>Hypertext Markup Language</i> “. Sie ist die Beschreibungssprache für Dokumente im World Wide Web. [It2]
HTML5	Die fünfte Fassung der Hypertext Markup Language, für die Auszeichnung und Vernetzung von Webinhalten.
JavaScript	Eine standardisierte Skriptsprache für die dynamische Erstellung von Webseiten.
JSON	Abkürzung für „ <i>JavaScript Object Notation</i> “. Ein Format für den Datenaustausch. [It3]
Metadaten	Daten, die Informationen über andere Daten enthalten.
MIT-Lizenz	Eine Open-Source Lizenz vom „Massachusetts Institute of Technology“, mit der die Wiederverwendung von Software und deren Quelltext genehmigt wird. [Wi2]
NFA	Abkürzung für „ <i>Nondeterministic Finite Automaton</i> “, (deutsch: Nichtdeterministischer Endlicher Automat). Ein endlicher Automat, bei dem es für den einen Zustandsübergang mehrere gleichwertige Möglichkeiten geben kann. [Wi10]

Nutzungskontext:	Wird in ISO 9241-11 definiert als: „Die Benutzer, Arbeitsaufgaben, Ausrüstung (Hardware, Software und Materialien) sowie physische und soziale Umgebung, in der das Produkt genutzt wird.“ [Ge12]
SVG	Abkürzung für „ <i>Scalable Vector Graphics</i> “. Ein Format für die Erstellung von vektorbasierten Grafiken.
Usability	Auch Benutzerfreundlichkeit oder Gebrauchstauglichkeit, „bezeichnet die vom Nutzer erlebte Nutzungsqualität bei der Interaktion mit einem System.“ [Wi8]
W3C	Abkürzung für „ <i>World Wide Web Consortium</i> “. Eine Einrichtung, welche Standards für das World Wide Web festlegt.
Widget	Komponente einer grafischen Benutzeroberfläche
XML	Abkürzung für „ <i>Extensible Markup Language</i> “. Eine Auszeichnungssprache für die Beschreibung und dem Austausch von komplexen Datenstrukturen. [It5]

Literaturverzeichnis

- [Cl03] V. Claus, A. Schwill (2003), *Schülerduden Informatik*, 4. Auflage, Dudenverlag.
- [Di03] DIN Deutsches Institut für Normung e.V. (2003), Software-Ergonomie für Multimedia-Benutzungsschnittstellen - Teil 1, *Gestaltungsgrundsätze und Rahmenbedingungen (DIN EN ISO 14915-1)*, Beuth Verlag GmbH.
- [Di08] DIN Deutsches Institut für Normung e.V. (2008), Ergonomie der Mensch-System-Interaktion - Teil 110, *Grundsätze der Dialoggestaltung (DIN EN ISO 9241-110)*, Beuth Verlag GmbH.
- [El] Artikel auf der Webseite eLearningTipps.de: *Methoden*.
(<http://www.elearningtipps.de/E-Learning/Methoden/>) [Stand 24.03.15]
- [Ge12] T. Geis (2012), *Nutzungskontext, Erfordernisse, Anforderungen und Lösung – das Arbeitsmodell des Usability-Engineering*.
(<http://www.procontext.com/aktuelles/2012/06/nutzungskontext-erfordernisse-anforderungen-und-loesung-das-arbeitsmodell-des-usability-engineering.html>)
[Stand 12.11.14]
- [Gi] H. Gierhardt, *Einführung in die Theoretische Informatik I*.
(<http://www.oberstufeninformatik.de/theorie/TheorieI.pdf>) [Stand 03.11.14]
- [Ha12] A. Harris (2012), *HTML5 Schnelleinstieg für Dummies*, Übersetzung aus dem Amerikanischen von Gerhard Franken, WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim.
- [Ho09] D.W. Hoffmann (2009), *Theoretische Informatik*, Hanser Verlag, München.
- [It1] Artikel auf der Webseite ITWissen.info: *DOM (document object model)*.
(<http://www.itwissen.info/definition/lexikon/document-object-model-DOM.html>)
[Stand 09.03.15]
- [It2] Artikel auf der Webseite ITWissen.info: *HTML (hypertext markup language)*.
(<http://www.itwissen.info/definition/lexikon/hypertext-markup-language-HTML.html>) [Stand 09.03.15]
- [It3] Artikel auf der Webseite ITWissen.info: *JSON (JavaScript object notation)*.
(<http://www.itwissen.info/definition/lexikon/JSON-JavaScript-object-notation.html>) [Stand 29.03.15]
- [It4] Artikel auf der Webseite ITWissen.info: *Drag-und-Drop*.
(<http://www.itwissen.info/definition/lexikon/Drag-und-Drop-drag-and-drop.html>)
[Stand 29.03.15]
- [It4] Artikel auf der Webseite ITWissen.info: *XML (extensible markup language)*.
(<http://www.itwissen.info/definition/lexikon/extended-markup-language-XML.html>) [Stand 29.03.15]
- [Kl06] A. Kleijn (2006), *Open-Source Lizenzen*.
(<http://www.heise.de/open/artikel/Open-Source-Lizenzen-221957.html>) [Stand 29.03.15]
- [Me1] Artikel auf der Webseite Mediaevent.de: *CSS Transition – Animationen und Effekte ohne Javascript*.
(<http://www.mediaevent.de/css/transition.html>) [Stand 16.03.2015]

- [Me2] Artikel auf der Webseite Mediaevent.de: *DOM .style, getAttribute, setAttribute, attributes*.
(<http://www.mediaevent.de/javascript/DOM-Style-CSS.html>) [Stand 16.03.2015]
- [Mi08] D. Mies (2008), *Webseiten erstellen für Einsteiger*, Galileo Press, Bonn.
- [Ni08] H.M. Niegemann (2008), *Kompodium Multimediales Lernen*, Springer-Verlag, Berlin Heidelberg.
- [Pa05] V. Pankratius, A. Oberweis, W. Stucky (2005), 9. Workshop Multimedia in Bildung und Wirtschaft, *Lernobjekte im E-Learning - Eine kritische Beurteilung zugrunde liegender Konzepte anhand eines Vergleichs mit komponentenbasierter Software-Entwicklung*, Technische Universität Ilmenau.
- [Ro14] H. Rohland (2014), *Rechtliche Aspekte für die Gestaltung von eLearning-Umgebungen*“, Vorlesungsfolien „Virtuelle Lernumgebungen“, Technische Universität Dresden.
- [Sc02] R. Schulmeister (2002), *Taxonomie der Interaktivität von Multimedia – Ein Beitrag zur aktuellen Metadaten-Diskussion*, Universität Hamburg.
(<http://www.zhw.uni-hamburg.de/pdfs/interaktivitaet.pdf>) [Stand 29.03.15]
- [Sc05] R. Schulmeister (2005), *Interaktivität in Multimedia-Anwendungen*.
(<http://www.e-teaching.org/didaktik/didaktik/gestaltung/interaktiv/InteraktivitaetSchulmeister.pdf>) [Stand 29.03.15]
- [St14] R. Steyer (2014), *jQuery – Das universelle JavaScript-Framework für das interaktive Web und mobile Anwendungen*, Carl Hanser Verlag, München.
- [We07] G. Weuffen (2007), Lernmodul auf der Webseite matheprisma.uni-wuppertal.de, *Die Turingmaschine „Biber am laufenden Band“*
(<http://www.matheprisma.uni-wuppertal.de/Module/Turing/>) [Stand 14.11.14]
- [Wi1] Wikipedia-Artikel „*Datenstruktur*“, Version vom 22. Oktober 2014 21:58 Uhr.
(<https://de.wikipedia.org/wiki/Datenstruktur>) [Stand 20.12.2014]
- [Wi2] Wikipedia-Artikel „*MIT-Lizenz*“, Version vom 6. März 2015 12:50 Uhr.
(<http://de.wikipedia.org/wiki/MIT-Lizenz>) [Stand 29.03.2015]
- [Wi3] Wikipedia-Artikel „*Theoretische Informatik*“, Version vom 10. September 2014 12:43 Uhr.
(http://de.wikipedia.org/wiki/Theoretische_Informatik) [Stand 18.11.2014]
- [Wi4] Wikipedia-Artikel „*Graph (Graphentheorie)*“, Version vom 14. Dezember 2014 03:06 Uhr.
([http://de.wikipedia.org/wiki/Graph_\(Graphentheorie\)](http://de.wikipedia.org/wiki/Graph_(Graphentheorie))) [Stand 05.01.2015]
- [Wi5] Wikipedia-Artikel „*JavaScript*“, Version vom 11. März 2015 17:28 Uhr.
(<http://de.wikipedia.org/wiki/JavaScript>) [Stand 29.03.2015]
- [Wi6] Wikipedia-Artikel „*HTML5*“, Version vom 10. Februar 2015 22:27 Uhr.
(<http://de.wikipedia.org/wiki/HTML5>) [Stand 29.03.2015]
- [Wi7] Wikipedia-Artikel „*Ajax (Programmierung)*“, Version vom 5. März 2015 15:29 Uhr.
([http://de.wikipedia.org/wiki/Ajax_\(Programmierung\)](http://de.wikipedia.org/wiki/Ajax_(Programmierung))) [Stand 29.03.2015]
- [Wi8] Wikipedia-Artikel „*Benutzerfreundlichkeit*“, Version vom 29. März 2015 17:07 Uhr.
(<http://de.wikipedia.org/wiki/Benutzerfreundlichkeit>) [Stand 29.03.2015]

- [Wi9] Wikipedia-Artikel „*Deterministischer Endlicher Automat*“, Version vom 8. Januar 2015 01:45 Uhr.
(http://de.wikipedia.org/wiki/Deterministischer_endlicher_Automat)
[Stand 29.03.2015]
- [Wi10] Wikipedia-Artikel „*Nichtdeterministischer Endlicher Automat*“, Version vom 18. April 2014 15:22 Uhr.
(http://de.wikipedia.org/wiki/Nichtdeterministischer_endlicher_Automat)
[Stand 29.03.2015]

Links der vorgestellten Software und verwendeten Webseiten:

D3.js

<http://d3js.org/>

Firebug

<https://addons.mozilla.org/de/firefox/addon/firebug/>

jQuery

<https://jquery.com/>

jQuery UI

<https://jqueryui.com/>

Notepad++

<http://notepad-plus-plus.org/>

W3schools

<http://www.w3schools.com/>

Abbildungsverzeichnis

Abbildung 1: Konzept Gebrauchstauglichkeit [Di08, S.3]	7
Abbildung 2: Teilgebiete der Theoretischen Informatik [Wi3]	12
Abbildung 3: Beispiel einer formalen Sprache mit zugehöriger Grammatik, sowie Darstellung des äquivalenten Ausdrucks und Automaten	13
Abbildung 4: Screenshot des JAVA-Applets "FS-Visio"	15
Abbildung 5: Turingmaschinen-Simulator von "MathePrisma"	18
Abbildung 6: Visualisierung von Datenstrukturen.....	26
Abbildung 7: Ein gerichteter Graph mit zugehöriger Adjazenzmatrix..	28
Abbildung 8: Webseite „bluffton.edu“	30
Abbildung 9: Webseite "sorting.at".....	39
Abbildung 10: Grundgerüst einer HTML-Seite im Quellcode und im Objektmodell (DOM)	44
Abbildung 11: Eine einfache SVG-Grafik, bestehend aus Rechteck und Kreis	47
Abbildung 12: Drei Polygone erzeugt mit <polygon>.....	49
Abbildung 13: Zwei Pfade gezeichnet mit <path>.....	49
Abbildung 14: Eine SVG-Gruppe mit <rect>, <circ> und <text>	51
Abbildung 15: Zwei SVG-Texte, deren Textzeichen einzeln positioniert wurden.....	52
Abbildung 16: Zwei Rechtecke gezeichnet mittels <canvas>	53
Abbildung 17: Zwei Pfade erzeugt mit <canvas>	54
Abbildung 18: Erstellte Grafik mit d3.js	56
Abbildung 19: Ein Balkendiagramm erzeugt mit d3.js.....	58
Abbildung 20: Zwei Buttons erzeugt mit dem <input>-Tag.....	67
Abbildung 21: Button, der eine Grafik enthält.....	68
Abbildung 22: Zwei Radiobuttons, zur Bestimmung der Textfarbe.....	69
Abbildung 23: Drei Eingabefelder erzeugt mit dem <input>-Tag.....	70
Abbildung 24: Ein Slider, erzeugt mit dem <input>-Tag	71
Abbildung 25: Zwei Auswahlménüs.....	72
Abbildung 26: Zwei Auswahlménüs, zur Bestimmung der Farbe	73
Abbildung 27: Ein mehrzeiliges Eingabefeld, erzeugt mit dem <textarea>-Tag.....	73
Abbildung 28: Accordion-Widget, bei dem das erste Element selektiert ist	79
Abbildung 29: Dynamisches Ein- und Ausblenden von DOM-Elementen.	83
Abbildung 30: „FA-Simulation-Tool“-Eingabefenster.....	92
Abbildung 31: Aktivitätsdiagramm – Umwandlung in einen DFA	925

Tabellenverzeichnis

Tabelle 1: Lernobjekte für Formale Sprachen und Grammatiken.....	16
Tabelle 2: Lernobjekte für Turingmaschinen.....	19
Tabelle 3: Lernobjekte für Kellerautomaten.....	21
Tabelle 4: Lernobjekte für Endliche Automaten.....	24
Tabelle 5: Lernobjekte für Lineare Datenstrukturen.....	28
Tabelle 6: Lernobjekte für Graphen.....	32
Tabelle 7: Lernobjekte für Bäume.....	36
Tabelle 8: Sortieralgorithmen im Überblick.....	40
Tabelle 9: Lernobjekte für Sortierverfahren.....	41

Rechtliche Hinweise für die Benutzung der evaluierten Anwendungen

Formale Sprachen

cs.montana.edu (http://www.cs.montana.edu/webworks/projects/cfg_new/):
Copyright © Montana State University

debuggex.com (<https://www.debuggex.com/>):
Copyright © 2013 www.debuggex.com

diotavelli.net (<http://www.diotavelli.net/people/void/demos/cky.html>):
Keine Angabe zum Entwickler verfügbar.

FS-Visio (<http://parsys.informatik.uni-oldenburg.de/~skript/fs-pub/fsvisio/>):
Copyright © 2015 Carl von Ossietzky Universität Oldenburg

mpeveler.com (<http://mpeveler.com/assets/content/projects/cfg2cnf.php>):
MIT-Lizenz (<https://github.com/MasterOdin/CFG2CNF/blob/master/LICENSE>)

oursland.net (<http://oursland.net/projects/regexp/>):
Keine Angabe zum Entwickler zu finden. Quellcode wird zum Download frei zur Verfügung gestellt

regex101.com (<http://regex101.com/>):
Keine Angaben zum Urheberrecht verfügbar, Kontaktadresse: contact@regex101.com, auf Anfrage keine Antwort erhalten.

SeeYK (<http://www2.tcs.ifi.lmu.de/SeeYK/>):
Keine Angaben zum Urheberrecht verfügbar, Entwickler der Anwendung ist Martin Lange, Kontaktadresse: martin.lange@ifi.lmu.de, auf Anfrage keine Antwort erhalten.

uni-leipzig.de (<http://www.informatik.uni-leipzig.de/alg/lehre/ss08/AUTOSPRACHEN/Java-Applets/CYK-Algorithmus.html>):
Die Anwendung wurde von Roy Mennicke im Rahmen eines SHK-Vertrages an der Universität Leipzig entwickelt. Über die rechtliche Lage konnte auf Anfrage keine genaue Auskunft gegeben werden. Vermutlich liegen die Urheberrechte aber bei der Universität Leipzig.

Turingmaschine

iti.fh-flensburg.de (<http://www.iti.fh-flensburg.de/lang/theor/turing-maschine.htm>):
Copyright © Prof. Dr. Hans Werner Lang
Originaltext: „Der Inhalt der Webseiten ist urheberrechtlich geschützt. Die Vervielfältigung, Änderung, Verbreitung oder Speicherung von Informationen oder Daten, insbesondere von

Texten, interaktiven Anwendungen oder Bildmaterial – auch auszugsweise – ist nur mit meiner [Prof. Dr. Hans Werner Lang] Zustimmung zulässig.“

klickagent.ch (<http://turingmaschine.klickagent.ch/>):

Copyright © by klickagent.ch

Martinugarte.com (<https://martinugarte.com/turingmaschine/>):

Copyright © 2014 Martin Ugarte

MathePrisma (<http://www.matheprisma.uni-wuppertal.de/Module/Turing/Navi/Simulator.htm>):

Copyright © 1999 G. Weuffen

morphett.info (<http://morphett.info/turing/turing.html>):

Quellcode steht frei zur Verfügung. Das Herunterladen, Kopieren oder Verändern des Quellcodes für die eigenen Bedürfnisse ist erlaubt.

Originaltext: „Source code for this simulator is available on GitHub. You are welcome to download, copy, or adapt the code to your own needs. If you are doing something interesting with it, I'd be happy to hear from you about it.“

turing.org.uk (<http://www.turing.org.uk/turing/scrapbook/tmjava.html>):

Copyright © Andrew Hodges

Turingsimulator.net (<http://www.turingsimulator.net/>):

MIT-Lizenz

Uni-freiburg.de (<http://ais.informatik.uni-freiburg.de/turing-applet/turing/TuringMachineHtml.html>):

Copyright © 2013 Universität Freiburg

Uni-leipzig.de (<http://www.informatik.uni-leipzig.de/theo/RingVorl/0517/InteraktSkript/skripten/1/3/2.htm#Def.TuringAutomat>):

Copyright © 2015 Universität Leipzig, Diese Anwendung ist Teil eines Interaktiven Skripts von Prof. Dr. Siegmur Gerber.

xTuringMachine (<http://math.hws.edu/TMCM/java/xTuringMachine/>):

Copyright © David Eck

Kellerautomat

automatonsimulator (<http://automatonsimulator.com/>):

MIT-Lizenz (<https://github.com/kdickerson/automatonSimulator/blob/master/LICENSE.txt>)

cosc.canterbury.ac.nz (<http://www.cosc.canterbury.ac.nz/mukundan/thco/aplthco.html>):

Copyright © 2006 R.Mukundan

iti.fh-flensburg.de (<http://www.iti.fh-flensburg.de/lang/theor/stackautomat.htm>):

Copyright © Prof. Dr. Hans Werner Lang

Originaltext: „Der Inhalt der Webseiten ist urheberrechtlich geschützt. Die Vervielfältigung, Änderung, Verbreitung oder Speicherung von Informationen oder Daten, insbesondere von Texten, interaktiven Anwendungen oder Bildmaterial – auch auszugsweise – ist nur mit meiner [Prof. Dr. Hans Werner Lang] Zustimmung zulässig.“

uni-leipzig.de (<http://www.informatik.uni-leipzig.de/theo/RingVorl/0517/InteraktSkript/skripten/1/3/1.htm#Def.Kellerautomat>):

Copyright © 2015 Universität Leipzig. Diese Anwendung ist Teil eines Interaktiven Skripts von Prof. Dr. Siegmur Gerber.

Endliche Automaten

ABCEZ (<http://www.cs.columbia.edu/~zeph/software/ABCEZ/abcez.html#>):

Copyright © 2000 Mindspring Software. All Rights Reserved.

AutomatenSimulator (<http://www.informatik.uni-leipzig.de/theo/RingVorl/0517/InteraktSkript/mumed/2/start.html>):

Copyright © 2015 Universität Leipzig, Diese Anwendung ist Teil eines Interaktiven Skripts von Prof. Dr. Siegmur Gerber.

automatonsimulator (<http://automatonsimulator.com/>):

MIT-Lizenz (<https://github.com/kdickerson/automatonSimulator/blob/master/LICENSE.txt>)

cosc.canterbury.ac.nz (<http://www.cosc.canterbury.ac.nz/mukundan/thco/applthco.html>):

Copyright © 2006 R.Mukundan

fh-dortmund.de (<https://kik.informatik.fh-dortmund.de/abschlussarbeiten/endllicheAutomaten/>):

Copyright © 2012 KI-Labor der FH-Dortmund

hackingoff.com (<http://hackingoff.com/compilers/nfa-to-dfa-conversion>):

Copyright © 2012 HackingOff.com

iti.fh-flensburg.de (<http://www.iti.fh-flensburg.de/lang/theor/automat-epsilon.htm>):

Copyright © Prof. Dr. Hans Werner Lang

Originaltext: „Der Inhalt der Webseiten ist urheberrechtlich geschützt. Die Vervielfältigung, Änderung, Verbreitung oder Speicherung von Informationen oder Daten, insbesondere von Texten, interaktiven Anwendungen oder Bildmaterial – auch auszugsweise – ist nur mit meiner [Prof. Dr. Hans Werner Lang] Zustimmung zulässig.“

jAutomata (<http://www.logic.at/staff/searles/jautomata/applet.html>):

Copyright © 2002 Karl S. Gmeiner

NDEA-Simulator (<http://www.informatik.uni-leipzig.de/theo/RingVorl/0517/InteraktSkript/mumed/8/index.html>):

Copyright © 2015 Universität Leipzig, Diese Anwendung ist Teil eines Interaktiven Skripts von Prof. Dr. Siegmur Gerber.

regex-automaton.com (<http://regex-automaton.com/applet/index.php>):

Vervielfältigung und Verbreitung mit expliziter Angabe des Urhebers/Ursprungs (P. Anderson, School of Computer Science and Information Systems, Birkbeck College, University of London) erlaubt.

Originaltext: „*The report may be freely copied and distributed provided the source is explicitly acknowledged.*“

Lineare Datenstrukturen

cosc.canterbury.ac.nz (<http://cosc.canterbury.ac.nz/mukundan/dsal/appldsal.html>):

Copyright © 2006 R.Mukundan

cs.usfca.edu (<https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>):

Copyright © 2011 David Galles

cs.vt.edu (<http://courses.cs.vt.edu/csonline/DataStructures/Lessons/QueuesImplementationView/applet.html>):

Copyright © 2015 Virginia Polytechnic Institutes and State University

MathePrisma (<http://www.matheprisma.uni-wuppertal.de/Module/LinDatSt/index.htm>):

Copyright © 1999 G. Weuffen

Graphen

bluffton.edu

(<http://www.bluffton.edu/homepages/facstaff/nesterd/java/graphalgorithms.html>):

Copyright © 2014 Daryl Nester, Weiterverwendung erlaubt

Originaltext: „*The contents of this page are © 2014 Darryl Nester. It is available to anyone who wishes to use it (like most things on the Internet).*“

cosc.canterbury (<http://cosc.canterbury.ac.nz/mukundan/dsal/appldsal.html>):

Copyright © 2006 R.Mukundan

cs.usfca.edu (<https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>):

Copyright © 2011 David Galles

Graphen's Test (<http://www.informatik.uni-leipzig.de/theo/RingVorl/0517/InteraktSkript/mumed/5/Graphen.html>):

Copyright © 2015 Universität Leipzig, Diese Anwendung ist Teil eines Interaktiven Skripts von Prof. Dr. Siegmur Gerber.

iti.fh-flensburg.de

(<http://www.iti.fh-flensburg.de/lang/algorithmen/graph/shortest-paths.htm>):

Copyright © Prof. Dr. Hans Werner Lang

Originaltext: „*Der Inhalt der Webseiten ist urheberrechtlich geschützt. Die Vervielfältigung, Änderung, Verbreitung oder Speicherung von Informationen oder Daten, insbesondere von Texten, interaktiven Anwendungen oder Bildmaterial – auch auszugsweise – ist nur mit meiner [Prof. Dr. Hans Werner Lang] Zustimmung zulässig.*“

MathePrisma (<http://www.matheprisma.uni-wuppertal.de/Module/Graphen/index.htm>):
Copyright © 1999 G. Weuffen

tum.de (<https://www-m9.ma.tum.de/material/de/spp-dijkstra/>):

Urheberrechtlich geschützt, verantwortlich für den Internetauftritt ist Dr. Michael Ritter von der Technischen Universität München (m.ritter@ma.tum.de)

Originaltext: „Die durch die Seitenbetreiber erstellten Inhalte und Werke auf diesen Seiten unterliegen dem deutschen Urheberrecht. Beiträge Dritter sind als solche gekennzeichnet. Die Vervielfältigung, Bearbeitung, Verbreitung und jede Art der Verwertung außerhalb der Grenzen des Urheberrechtes bedürfen der schriftlichen Zustimmung des jeweiligen Autors bzw. Erstellers. Downloads und Kopien dieser Seite sind nur für den privaten, nicht kommerziellen Gebrauch gestattet.“

ViSuAl (<https://kik.informatik.fh-dortmund.de/abschlussarbeiten/suchalgorithmen/grapheditor.html>):

Copyright © 2012 KI-Labor der FH-Dortmund

Bäume

cs.usfca.edu (<https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>):

Copyright © 2011 David Galles

durzinsky.net (<http://www.durzinsky.net/develop/java/algodat/ueb10bin.html>):

Copyright © 2003 Markus Durzinsky, Freie Verwendung erlaubt, Quellcode ist mit einer GNU-General Public Lizenz versehen.

Originaltext auf der Webseite:

„Sämtliche auf dieser Homepage befindlichen Inhalte, Texte, Bilder und Quellcodes sind, wenn nicht anders gekennzeichnet, von mir selbst erstellt worden und dürfen frei verwendet werden. Die Quelltexte sind größtenteils unter der GNU General Public License bzw. der Lesser General Public License mit meinem Copyright versehen.“

fh-regensburg.de (<http://fbim.fh-regensburg.de/~saj39122/bruhi/>):

Copyright © OTH Regensburg

Originaltext: „Falls nicht anders angegeben, unterliegen die Internetseiten der OTH Regensburg dem Urheberrecht (Copyright). Dies gilt insbesondere für das Layout der Homepage, Texte, Bilder, Grafiken, Ton-, Video- oder Animationsdateien, Filme und andere multimediale Dateien einschließlich deren Anordnung auf den Web-Seiten. Die Vervielfältigung, Verwendung, Bearbeitung, Einspeicherung, Verarbeitung und Wiedergabe solcher Seiten (oder Teilen davon) in anderen elektronischen oder gedruckten Publikationen und deren Veröffentlichung (auch im Internet) ist nur nach vorheriger schriftlicher Zustimmung gestattet. Der Nachdruck und die Auswertung von Pressemitteilungen ist mit Quellenangabe gestattet. Weiterhin können Bilder, Grafiken, Filme, Videoaufzeichnungen, Text- oder sonstige Dateien ganz oder teilweise dem Urheberrecht Dritter unterliegen. Alle innerhalb des Internetangebotes genannten und ggf. durch Dritte geschützten Marken- und Warenzeichen unterliegen uneingeschränkt den Bestimmungen des jeweils gültigen Kennzeichenrechts und den Besitzrechten der jeweiligen eingetragenen Eigentümer. Allein aufgrund der bloßen Nennung im Internetangebot der OTH Regensburg ist nicht der Schluss zu ziehen, dass Markenzeichen nicht durch Rechte Dritter geschützt sind.“

MathePrisma (<http://www.matheprisma.uni-wuppertal.de/Module/BinSuch/index.htm>):
Copyright © 1999 G. Weuffen

people.ksp.sk (<http://people.ksp.sk/~kuko/bak/index.html>):
Weiterverwendung als Lehrmittel und für den privaten Gebrauch erlaubt.
Originaltext: „*This applet animates the functioning of several dictionary data structures. It can be used as a teaching aid or for self-study.*“

qmatica.com (<http://www.qmatica.com/DataStructures/Trees/AVL/AVLTree.html>):
Copyright © 2007 Arsen Gogeshvili

Tree Animation Tool
(<http://groups.engin.umd.umich.edu/CIS/course.des/cis350/treetool/>):
Copyright ©2002 Initech and The University of Michigan - Dearborn.

uni-leipzig.de (<http://www.informatik.uni-leipzig.de/theo/RingVorl/0517/InteraktSkript/mumed/1/AVLApplet.html>):
Copyright © 2015 Universität Leipzig, Diese Anwendung ist Teil eines Interaktiven Skripts von Prof. Dr. Siegmар Gerber.

unizar.es (<http://webdiis.unizar.es/asignaturas/EDA/AVLTree/avltree.html>):
Keine Angaben zum Urheberrecht verfügbar, Entwickler der Anwendung ist Arsen Gogeshvili, Kontaktadresse: arsen@seanet.com, auf Anfrage keine Antwort erhalten.

uottawa.ca (<http://www.site.uottawa.ca/~stan/csi2514/applets/avl/BT.html>):
Copyright © 2006-2015 Stan Matwin

Sortierverfahren

bluffton.edu
(<http://www.bluffton.edu/homepages/facstaff/nesterd/java/SortingDemo.html>):
All rights reserved © 2012 Bluffton University, Entwickler der Anwendung ist Daryl Nester

cosc.canterbury.ac.nz (<http://cosc.canterbury.ac.nz/mukundan/dsal/aplpldsal.html>):
Copyright © 2006 R.Mukundan

cs.smith.edu (<http://cs.smith.edu/~thiebaut/java/sort/demo.html>):
Copyright (c) 1997 Biliаna Kaneva, Dominique Thiébaud

cs.usfca.edu (<https://www.cs.usfca.edu/~galles/visualization/>):
Copyright © 2011 David Galles

hermann-gruber.com (<http://www.hermann-gruber.com/lehre/sorting/sort.html>):
Keine Angaben zum Urheberrecht verfügbar, Entwickler der Anwendung sind Hermann Gruber und Felix Wimmer, Kontaktadresse: info@hermann-gruber.com, auf Anfrage keine Antwort erhalten.

huebsch-gemacht.de (<http://www.huebsch-gemacht.de/programmieren/java/ViA/Applets/Finals/Sort/index.html>):

Copyright © Sven Bürgel

iti.fh-flensburg.de (<http://www.iti.fh-flensburg.de/lang/algorithmen/sortieren/index.htm>):

Copyright © Prof. Dr. Hans Werner Lang

Originaltext: „Der Inhalt der Webseiten ist urheberrechtlich geschützt. Die Vervielfältigung, Änderung, Verbreitung oder Speicherung von Informationen oder Daten, insbesondere von Texten, interaktiven Anwendungen oder Bildmaterial – auch auszugsweise – ist nur mit meiner [Prof. Dr. Hans Werner Lang] Zustimmung zulässig.“

MathePrisma (<http://www.matheprisma.uni-wuppertal.de/Module/Sortieren/index.htm>):

Copyright © 1999 G. Weuffen

public.asu.edu (<http://www.public.asu.edu/~aiadicic/sort.html>):

Die Visualisierungen befinden sich auf dem persönlichen Webspaces von Alex Iadicicco, der von der Arizona State University bereitgestellt wurde. Zum Entwickler selbst und auch zum Urheberrecht sind keine Informationen verfügbar. In der Regel werden die Webseiten der Arizona State University mit folgendem Urheberrecht geschützt: Copyright © 2014 Arizona Board of Regents

sorting.at (<http://sorting.at/>):

Creative Commons BY-NC-SA License

sorting-algorithms (<http://sorting-algorithms.com/>):

Copyright © 2007 David R. Martin

uni-oldenburg.de

(<http://www-cg-hci.informatik.uni-oldenburg.de/~da/iva/baer/start/start.html>):

Keine Angaben zum Entwickler verfügbar.

xSortLab (<http://math.hws.edu/TMCM/java/xSortLab/>):

Copyright © David Eck