

Themen

	Seite	
Automaten in der Informatik	240	1
Häufig vorkommende Automatentypen	244	2
Zustandsübergänge anschaulich darstellen	249	3
 Automat mit Python programmieren I	252	4
 Automat mit Python programmieren II	257	5

weitere Themen siehe Seite 239

Automaten und formale Sprachen II

In fünf Lektionen lernen die Schülerinnen und Schüler mit HTML5 und CSS3 – einer weiteren formalen Sprache – den Standard für die Erstellung einfacher, statischer Webseiten kennen und anwenden. Dabei liegt ein besonderes Augenmerk auf der Trennung von Struktur und Layout und der responsiven Darstellung der Inhalte.

Für die Erstellung der Dateien wird ein UTF-8-fähiger Editor benötigt. Das kann beispielsweise Notepad++ sein, ein freier Texteditor für Windows und kompatible Betriebssysteme.

Um die erstellten HTML-Dateien auf dem Bildschirm anzeigen zu können, wird ein aktueller Browser benötigt. Eine Übersicht der Browserversionen, mit denen die in den Lektionen 4 und 5 erzeugten responsiven Webseitenlayouts finden Sie auf der Seite 180.

Beim Erstellen von HTML- und CSS-Dateien schleichen sich leicht Syntax- oder Strukturfehler ein, die zu Anzeigefehlern im Browser führen können. Diese Fehler lassen sich mit Hilfe eines Validators ermitteln.

Einen frei verfügbaren HTML- und CSS-Validator stellt das World Wide Web Consortium (kurz W3C), also das Gremium zur Standardisierung der Techniken im World Wide Web, unter der URL <https://validator.w3.org/> zur Verfügung. Dort kann man auch HTML- und CSS-Dateien hochladen und überprüfen.



Vertiefende Projekte

Die fünf Lektionen vermitteln alle Grundlagen, mit deren Hilfe die Schülerinnen und Schüler beispielsweise im Rahmen eines Projekts eigene Webseiten erstellen können.

Themen

	Seite	
Struktur von HTML-Dateien	261	6
Texte, Listen, Links und Bilder	269	7
HTML und CSS	277	8
Anordnung und Abstände	281	9
Farben und Schriften	284	10

Beispielseiten zur Ansicht – Eigentum von EMS Kraus

Zustandsübergänge anschaulich darstellen

Ausführliche Beschreibungen helfen dabei, sich die Funktionsweise eines Automaten vorzustellen. Dabei stehen meist die Besonderheiten des beschriebenen Automaten im Vordergrund. Das Vergleichen unterschiedlicher Automaten ist auf dieser Basis aber nicht leicht.

Deshalb werden die Zustandsübergänge und Ausgaben von Automaten in Tabellen und Diagrammen dargestellt und mit allgemein gültigen Buchstaben und Zahlen benannt.

Zustandsübergangstabelle

Der Literaturautomat aus den vorherigen Kapiteln kann beispielsweise drei Zustände einnehmen, es gibt fünf mögliche Eingabe- und fünf mögliche Ausgabeaktionen. Welche Zustände und Eingaben am Literaturautomaten welche Ausgabeaktionen und Folgezustände hervorrufen, ist in dieser Tabelle zusammengefasst:

Aktueller Zustand	Eingabeaktionen	Ausgabeaktionen	Folgezustand
Z1	E1	A1	Z1
	E2	A2	Z2
Z2	E3	A3	Z1
	E4	A4	Z3
Z3	E5	A5	Z1

In anderen Varianten der Zustandsübergangstabelle werden die Eingaben in die Kopfzeile geschrieben. Darunter werden dann die Folgezustände notiert, die bei dieser Eingabe und dem jeweiligen Ausgangszustand auftreten.

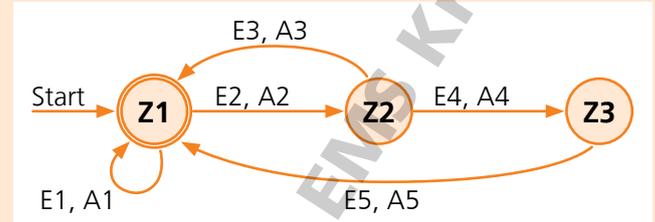
Aktueller Zustand	Eingabeaktionen				
	E1	E2	E3	E4	E5
Z1	Z1	Z2			
Z2			Z1	Z3	
Z3					Z1

Auf die gleiche Weise können auch die jeweiligen Ausgabeaktionen tabellarisch dargestellt werden.

Aktueller Zustand	Eingabeaktionen				
	E1	E2	E3	E4	E5
Z1	A1	A2			
Z2			A3	A4	
Z3					A5

Zustandsübergangsdiagramm

In Zustandsdiagrammen werden die Zustände üblicherweise als Kreise gezeichnet. Der Anfangszustand ist mit einem Pfeil und dem Wort „Start“ markiert.

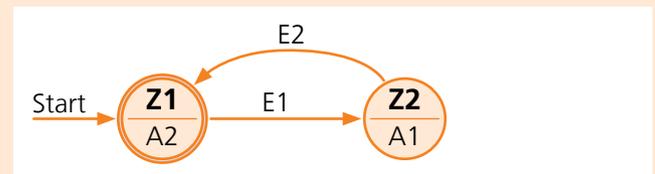


Die Vorgänge, die einen Zustandsübergang bewirken, sind als Pfeil zum Folgezustand eingezeichnet. Vorgänge, die keinen Zustandsübergang hervorrufen, sind durch einen Pfeil gekennzeichnet, der im Bogen wieder zurück zum alten Zustand führt.

Der Endzustand, den der Literaturautomat einnimmt, nachdem die geöffnete Schublade wieder geschlossen wurde, ist wieder der anfängliche Wartezustand Z1. Der Endzustand eines Automaten wird durch einen doppelten Kreis gekennzeichnet.

Im vorherigen Kapitel hatten wir festgestellt, dass der Literaturautomat zur Gruppe der Mealy-Automaten gehört, da die Ausgaben sowohl vom aktuellen Zustand als auch von der aktuellen Eingabe abhängen. In Zustandsdiagrammen für Mealy-Automaten werden die Eingaben und Ausgaben für jeden Zustandsübergang am Pfeil notiert.

In Zustandsübergangsdiagrammen für Moore-Automaten hingegen werden nur die Eingaben an den Pfeilen platziert. Die Ausgaben stehen hier in den Kreisen unter dem jeweiligen Folgezustand.



Durch die verallgemeinerte Darstellung in Tabellen und Diagrammen kann man die Modellierung ganz unterschiedlicher Automaten vergleichen. So lassen sich möglicherweise ähnliche Verhaltensweisen bei Automaten aus ganz verschiedenen Anwendungsbereichen entdecken.

Zustandsübergänge anschaulich darstellen

Aufgabe 1

- Notiere die Zustände und Zustandsübergänge des Händetrockners aus den vorherigen Kapiteln in einer Zustandsübergangstabelle in Abhängigkeit von den Eingabeaktionen.
- Ordne in einer zweiten Tabelle die Ausgabeaktionen den jeweiligen Zuständen und Eingabeaktionen zu.
- Zeichne das Zustandsübergangsdiagramm für den Händetrockner.

Beispiellösung

a)

	E1	E2
Z1	Z2	
Z2		Z1

b)

	E1	E2
Z1	A1	
Z2		A2



Aufgabe 2

- Notiere die Zustände und Zustandsübergänge des Futterautomaten aus den vorherigen Kapiteln in einer Zustandsübergangstabelle in Abhängigkeit von den Eingabeaktionen.
- Ordne in einer zweiten Tabelle die Ausgabeaktionen den jeweiligen Zuständen und Eingabeaktionen zu.
- Zeichne das Zustandsübergangsdiagramm für den Futterautomaten.

Beispiellösung

a)

	E1	E2	E3
Z1	Z2		
Z2		Z1	Z1

b)

	E1	E2	E3
Z1	A1		
Z2		A2	A3



Aufgabe 3

- Notiere die Zustände und Zustandsübergänge des Drehkreuzes aus den vorherigen Kapiteln in einer Zustandsübergangstabelle in Abhängigkeit von den Eingabeaktionen.
- Ordne in einer zweiten Tabelle die Ausgabeaktionen den jeweiligen Zuständen und Eingabeaktionen zu.
- Zeichne das Zustandsübergangsdiagramm für das Drehkreuz.

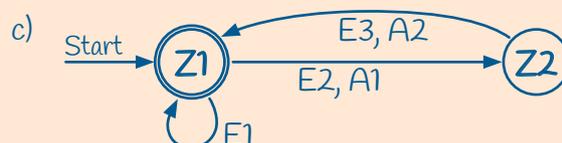
Beispiellösung

a)

	E1	E2	E3
Z1	Z1	Z2	
Z2			Z1

b)

	E1	E2	E3
Z1		A1	
Z2			A2



Zustandsübergänge anschaulich darstellen

Aufgabe 4

- Notiere die Zustände und Zustandsübergänge des Türdrückers mit Nummerntasten aus den vorherigen Kapiteln in einer Zustandsübergangstabelle in Abhängigkeit von den Eingabeaktionen.
- Ordne in einer zweiten Tabelle die Ausgabeaktionen den jeweiligen Zuständen und Eingabeaktionen zu.
- Zeichne das Zustandsübergangsdiagramm für den Türdrücker mit Nummerntasten.

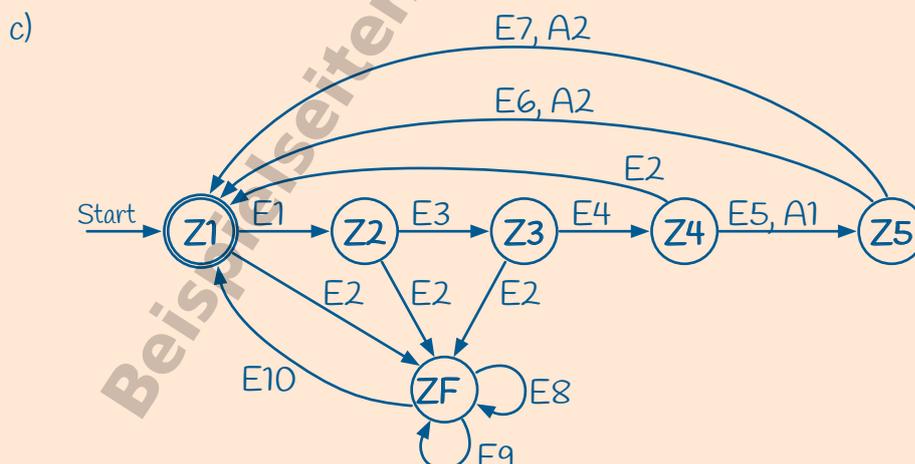
Beispiellösung

a)

	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10
Z1	Z2	ZF								
Z2		ZF	Z3							
Z3		ZF		Z4						
Z4		Z1			Z5					
Z5						Z1	Z1			
ZF								ZF	ZF	Z1

b)

	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10
Z1										
Z2										
Z3										
Z4					A1					
Z5						A2	A2			
ZF										



HTML und CSS

Mit der Auszeichnungssprache HTML werden die Inhalte einer Webseite strukturiert, aber nicht gestaltet. Für das Definieren der Gestaltung der Webseite wird eine so genannte Stylesheet-Datei benötigt (von englisch style sheet „Formatvorlage“).

Stylesheet-Dateien werden im CSS-Format erstellt. CSS (Cascading Style Sheet, sinngemäß „gestufte Formatvorlage“) ist eine Stylesheet-Sprache, die es erlaubt, für jedes einzelne Element einer HTML-Datei das Aussehen festzulegen.

Da der Inhalt und die Gestaltungsvorgaben auf die HTML- und die CSS-Datei aufgeteilt sind, müssen beide Dateien miteinander verknüpft werden. Das geschieht durch einen Verweis im `<head>` der HTML-Datei.

```
<head>
...
<link rel="stylesheet" href="style.css"/>
</head>
```

Das Attribut `rel` definiert, dass ein Stylesheet verknüpft werden soll, das Attribut `href` nennt die verknüpfte CSS-Datei `style.css`.

Auch die einzelnen Elemente der HTML-Datei müssen mit den Vorgaben in der CSS-Datei verknüpft werden. Dafür werden so genannte Selektoren verwendet, von denen es vier unterschiedliche Typen gibt.

*	Universalselektor	alle Elemente
A	Typselektor	alle Elemente vom Typ A
.xyz	Klassenselektor	alle Elemente der Klasse xyz
#abc	ID-Selektor	das Element mit der ID "abc"

Elemente mit einer ID darf es in einer HTML-Datei jeweils nur einmal geben. Von einer Klasse dürfen hingegen beliebig viele Elemente in einer HTML-Datei vorkommen.

Möchte man in der CSS-Datei Vorgaben definieren, die alle Elemente der HTML-Datei betreffen, nutzt man den Universalselektor `*`.

```
* {
}
```

Alle Gestaltungsvorgaben werden in der CSS-Datei in geschweiften Klammern nach dem jeweiligen Selektor notiert.

Sollen in der CSS-Datei Vorgaben für Elemente wie beispielsweise die Navigation definiert werden, notiert man das jeweilige Element in der CSS-Datei ohne ein vorangestelltes Zeichen.

HTML	CSS
<pre><nav> </nav></pre>	<pre>nav { } nav li { }</pre>

Möchte man auf einer Webseite mit mehreren Listen nur den Listenpunkten der Navigation eine bestimmte Gestaltung zuweisen, schreibt man vor dem `li`-Element das `nav`-Element.

Häufig gibt es in einer HTML-Datei mehrere Elemente eines Typs, also beispielsweise mehrere Bilder, von denen aber nur ein Teil auf eine bestimmte Weise gestaltet werden soll. In diesem Fall weist man den betroffenen Bildern eine gemeinsame Klasse zu.

In der CSS-Datei werden dann die Gestaltungsvorgaben für die Klasse definiert. Die Klasse wird dabei mit einem vorangestellten Punkt notiert. Auch hier können untergeordnete Elemente wie die Verknüpfung `a` in unserem Beispiel direkt angesprochen werden, indem sie hinter die Klasse geschrieben werden.

```
<figure class="logo">
  <a href="index.html">Start</a>
</figure>
```

```
.logo {
}
.logo a {
}
```

Möchte man mehrere Elemente zu einer gestalterischen Einheit zusammenfassen, die als Ganzes auf der Webseite angeordnet wird, fasst man sie in der HTML-Datei in `<div></div>`-Tags ein, die mit einer Klasse gekennzeichnet werden.

```
<div class="beschreibung">
...
</div>
```

HTML und CSS

Aufgabe 1

- Erstelle in dem Ordner, in dem die HTML-Dateien über die Sonnenuhren liegen, einen Unterordner namens styles.
- Erstelle eine Datei style.css und lege sie im Unterordner styles ab.
- Füge in die drei HTML-Dateien den Verweis auf die CSS-Datei ein.

Beispiellösung

```
<!DOCTYPE html>
<html lang="de">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sonnenuhren | Geschichte</title>
    <link rel="stylesheet" href="styles/style.css"/>
  </head>

  <body>
  ...
```

```
<!DOCTYPE html>
<html lang="de">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sonnenuhren | Bauarten</title>
    <link rel="stylesheet" href="styles/style.css"/>
  </head>

  <body>
  ...
```

```
<!DOCTYPE html>
<html lang="de">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sonnenuhren | Horizontalsonnenuhren</title>
    <link rel="stylesheet" href="styles/style.css"/>
  </head>

  <body>
  ...
```

HTML und CSS

Aufgabe 2

Ergänze in der Datei index.html die Klassen für die Verknüpfung mit der CSS-Datei.

- Füge im Element `<header>`, in dem das Banner verknüpft ist, in den Starttag die class `"banner"` ein.
- Füge in den Starttag des Textabsatzes unter der `h1`-Überschrift die class `"untertitel"` ein.
- Füge in den Starttag des Bildes die class `"rechts"` ein.
- Schließe die vier `<section>`-Elemente zusammen in `<div></div>`-Tags ein und füge in den Starttag des `<div>`-Elements die class `"beschreibung"` ein.

Beispiellösung

```
<body>
  <header class="banner">
    
  </header>
  ...
  <main>
    <h1>Geschichte der Sonnenuhr</h1>
    <p class="untertitel">vom Altertum bis in die Gegenwart</p>
    <div class="beschreibung">
      <section>
        ...
      </section>
    </div>
    <figure class="rechts">
      
      <figcaption>
        ...
      </figcaption>
    </figure>
  </main>
</body>
```

Aufgabe 3

Ergänze in der Datei bauarten.html die Klassen für die Verknüpfung mit der CSS-Datei.

- Füge im Element `<header>`, in dem das Banner verknüpft ist, in den Starttag die class `"banner"` ein.
- Füge in den Starttag des Textabsatzes unter der `h1`-Überschrift die class `"untertitel"` ein.

Beispiellösung

```
<body>
  <header class="banner">
    
  </header>
  ...
  <main>
    <h1>Unterschiedliche Sonnenuhr-Bauarten</h1>
    <p class="untertitel">unterscheiden sich vor allem in Form und Ausrichtung
      ihres Zifferblattes</p>
    ...
  </main>
</body>
```

HTML und CSS

Aufgabe 4

Ergänze in der Datei horizontal.html die Klassen für die Verknüpfung mit der CSS-Datei.

- Füge im Element `<header>`, in dem das Banner verknüpft ist, in den Starttag die `class "banner"` ein.
- Füge in den Starttag des Textabsatzes unter der `h1`-Überschrift die `class "untertitel"` ein.
- Füge in den ``-Starttag der Liste zur Sonnenuhr im Bild die `class "box"` ein.
- Füge in den Starttag des Bildes die `class "rechts"` ein.
- Schließe die zwei `<section>`-Elemente zusammen in `<div></div>`-Tags ein und füge in den Starttag des `<div>`-Elements die `class "beschreibung"` ein.

Beispiellösung

```

<body>
  <header class="banner">
    
  </header>
  ...

  <main>
    <h1>Horizontalsonnenuhren</h1>
    <p class="untertitel">die Ganztagssonnenuhren</p>
    <div class="beschreibung">
      <section>
        ...
        <ul class="box">
          <li>Material: Messing, Holz</li>
          <li>Abmessungen: 30,5 cm x 30,5 cm</li>
          <li>Hergestellt: 1568 in Prag</li>
        </ul>
        ...
      </section>
    </div>
    <figure class="rechts">
      
      <figcaption>
        ...

```