Sprachen und Automaten – Unterrichtseinheit

Theoretische Informatik: Reguläre Sprachen und endliche Automaten

Ein Beitrag von Johann-Georg Vogelhuber

Die theoretische Informatik bildet mit endlichen Automaten und formalen Sprachen das Grundgerüst für moderne Programmiersprachen. Mit dieser Unterrichtseinheit können sich Ihre Schülerinnen und Schüler die Grundlagen dieses Themengebiets mithilfe von handlungsorientierten Situationen und differenzierten Aufgaben erarbeiten. Unterstützt wird die Erarbeitung durch das verlinkte Hilfematerial und eine interaktive Software zur Erstellung von endlichen Automaten.

KOMPETENZPROFIL – UNTERRICHTSEINHEIT

Klassenstufe:	Sek. II
Dauer:	10–15 Unterrichtsstunden
Lernziele:	Die Lernenden … 1. entwerfen Zustandsdiagramme endlicher Automaten zur Erkennung von korrekten Eingaben, 2. unterscheiden deterministische und nichtdeterministische endliche Automaten und wandeln diese ineinander um.
Kompetenzen:	Modellieren und Implementieren
Themenbereiche:	Theoretische Informatik, Formale Sprachen, Reguläre Sprachen und Grammatiken, deterministische und nichtdeterministische endliche Automaten, deterministische Kellerautomaten

Auf einen Blick

Benötigt

☐ Tablet/Laptop/PC pro Schüler/in oder pro Schülerpaar
☐ Internetzugang

Einstieg

Thema:	**Endliche Automaten mit Ausgabe**
M 1	**Wie kann man einen Automaten formal beschreiben?**
M 2	**Formale Definition für endliche Automaten mit Ausgabe**
M 3	**Übungsaufgaben: Mealy-Automaten**

Erarbeitung

Thema:	**Endliche deterministische und nichtdeterministische Automaten**
M 4	**Wie können E-Mail-Adressen automatisiert überprüft werden?**
M 5	**Übungsaufgaben: Deterministische endliche Automaten**
M 6	**Reguläre Sprachen und Grammatiken**
M 7	**Übungsaufgaben: Reguläre Sprachen und Grammatiken**
M 8	**Nichtdeterministische Automaten – Mit welchen Regeln lässt sich eine Firewall [illegible]en?**
M 9	**Äquivalenz von NEA und DEA – Die Potenzmengenkonstruktion**

Vertiefung (optional)

Thema:	Grenzen endlicher Automaten und regulärer Sprachen
M 10	**Grenzen endlicher Automaten und regulärer Sprachen**
M 11	**Endliche Kellerautomaten – Wie kann man korrekte Klammerausdrücke erkennen?**

Sicherung

M 12	**Bist du fit in regulären Sprachen und endlichen Automaten?**
Benötigt:	*Kahoot!-Quiz:* https://raabe.click/Kahoot-Lernerfolgskontrolle

Erklärung zu den Symbolen

Dieses Symbol markiert differenziertes Material. Wenn nicht anders ausgewiesen, befinden sich die Materialien auf mittlerem Niveau.		
leichtes Niveau	mittleres Niveau	schwieriges Niveau
Zusatzaufgabe	Alternative	

M 2 Formale Definition für endliche Automaten mit Ausgabe

Echte Automaten lassen sich durch ein theoretisches Modell beschreiben. Man bezeich et es **endlicher Automat mit Ausgabe**. Das Modell wurde 1955 in dem Artikel *A Method for Sy thesizing Sequential Circuits* von dem US-amerikanischen Mathematiker George H. Mealy ver fentlicht.

Aufgabe 1

Vervollständigen Sie die folgende Definition mithilfe der verlinkten Quellen.

Endliche Automaten

Definition: Endlicher Automat mit Ausgabe

Ein **endlicher Automat mit Ausgabe** (https://raabe.click/Info-E dliche-Automaten) ist e Tupel $(Q, \Sigma, \Omega, \delta, \lambda, q_0)$.

Dabei ist:

- Q die ____________________. Dies sind die versc iedenen ____________________, die der Automat annehmen kann. Diese Menge muss ichtleer und e in
- Σ das ____________________________. Dies ist der Ze nvorrat, der für die ______________ zur Verfügung steht. Dies st eine nichtleere, liche Menge.
- Ω das ____________________________ Auch Ω ist eine nichtlee , endliche Menge. Die in Ω enthalten Zeichen sind die mögl en Ze ____________________.
- δ die ____________________. Di se Funktion ordnet jedem Paar aus Eingabezeichen und Zustand genau einen ____________________ zu.
- λ die ____________________. Diese Fu tion legt fü es Paar aus Eingabezeichen und Zustand fest, welches Z ____________ werden soll.
- $q_0 \in Q$ ist der ____________________.

Mealy-Automat

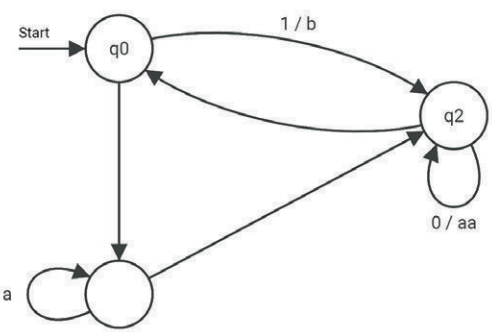
Ein endlicher Automat mit Ausgabe w auch **Mealy-Automat** genannt (https://raabe.click/Info-Mealy-Automat).

Aufgabe 2

Im Folgenden sind ein Über sgraph, eine Übergangsfunktion sowie eine Ausgabefunktion gegeben. Ver ndigen Sie di Tabellen und den Graphen so, dass die Tabellen und der Graph denselb Autom n beschreibe

δ	0	1
	q_1	
q_1		q_2
q_2		[illegible]

λ	0	1
	a	
q_1		b
q_2		b



Grafik: Johann-Georg Vogelhuber

M 8 Nichtdeterministische Automaten – Mit welchen Regeln lässt sich eine Firewall umsetzen?

Anwendungssituation

Da es in letzter Zeit häufiger zu einer missbräuchlichen Nutzung der Rechner in den Computerräumen gekommen ist, sollen die eingegebenen Web-Adressen auf dem Proxy-Server der Sch... gefiltert und bei Bedarf geblockt werden. Aus technischen und datenschutzrecht...hen Gründ...n da... die URL überprüft werden. Enthält die URL eine der vorgegebenen Zeichenke... so wird sie. Eine Unterscheidung nach Groß- und Kleinschreibung findet nicht statt. ... URL soll a... einschlägige Substrings geprüft werden. Dazu soll zuerst ein Automat entwickelt werden, um ein Kon... für die Programmierung dieses Filters zu entwickeln.

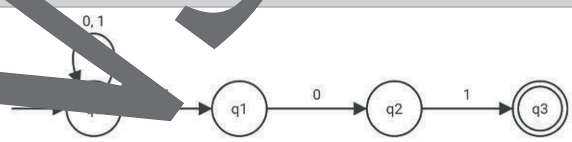
Aufgabe 1: Analyse

Lesen Sie sich die Situationsbeschreibung durch und be...orten dana... folg...den Fragen:

- Welche Informationen sind in der Situationsbeschreibung ...eben?
- Welche Daten fehlen noch?
- Welche Schritte sind zur Durchführung der ...ufgabe notwendig?
- Welche Schwierigkeiten könnten bei der Be...beitung a...?

Infotext – Nichtdeterministische endliche Automaten

Die bisher betrachteten Automaten werden als deterministische endliche Automaten (DEA) bezeichnet. Das Wort determin... bedeutet hier, dass für jeden Zustand mit jedem Eingabeze...n das Verhalten des Automaten eindeutig festgelegt ist.



Grafik: Johann-Georg Vogelhuber

Der Automat hat keine Wahl, welche Anweisung auszuführen ist. In d... theoretischen Informatik ist neben dem DEA auch das Modell des **nichtdeterministischen endli... ...tomaten (NEA)** v... Bedeutung. Die Abbildung zeigt so einen Automaten. Im Gegensatz zu einem DEA kann es meh...re Überg... mit dem gleichen Eingabezeichen von einem Zustand aus geben. D. h. der Weg durch den Übergangsgraphen ... du...n die Ei...gabe nicht länger eindeutig bestimmt.

Der NEA akzeptiert eine Eingabe, wenn mindestens ...ner der möglichen Werte in einem der akzeptierten Zustände (Endzustände) endet.

Aufg... 2

Beschreib... Sie, ...elche Sprache der oben abgebildete Automat akzeptiert und erstellen eine entsprechende Gr...matik. Vergleichen Sie ihre Lösung mit ihrem Ergebnis zu Aufgabe 6 von M 7.

Aufgabe...

Erstellen Sie ... die oben beschriebene Situation einen NEA. Verwenden Sie dazu mindestens drei unterschiedlic... Schlüsselwörter. Nutzen Sie das Tool *flaci.com/autoedit*.

Vorlage Aufgabe 2

Tipp: ...edarf können sie die Vorlage verwenden: https://raabe.click/Flaci-Vorlage-Aufgabe2

M 11 Endliche Kellerautomaten – Wie kann man korrekte Klammerausdrücke erkennen?

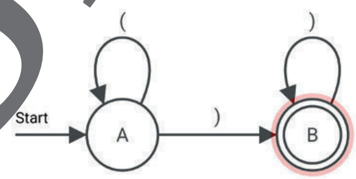
Für die Überprüfung von kontextfreien Sprachen muss das Konzept des endlichen deterministischen Automaten erweitert werden. Dazu wird ein DEA um einen Stack (Kellerspeicher) ergänzt. Diese Art von Automat bezeichnet man als **Kellerautomat** oder *Stackmachine*. Der Stackspeicher enthält Inhalte des Kelleralphabets. Es ist immer nur ein Zugriff auf das oberste Element möglich. Mit den Zugriffsmethoden *push* und *pop* können Elemente auf den Stapel gelegt oder genommen werden.

Definition

Ein **Kellerautomat** wird durch ein Tupel $A = (Z, \Sigma, K, z_0, Z_E, \delta, \lambda)$ festgelegt. Dabei sind die Bezeichnungen analog zu denen eines DEA. Zusätzlich ist K das Kelleralphabet. Dieses Alphabet besteht aus einem endlichen Zeichenvorrat. Das Symbol $\# \in K$ das **Anfangssymbol**. Dieses Symbol ist immer das unterste Symbol des Kellerspeichers. Die Zustandsübergangsfunktion ordnet jedem Zustand, Eingabezeichen und Kellerinhalt einen Folgezustand zu. Die Ausgabefunktion λ ordnet jedem Zustand, Eingabezeichen und Kellerinhalt und einen Kellerinhalt zu.
Ein Kellerautomat **akzeptiert** ein Eingabewort, wenn am Ende einer der akzeptierten Zustände erreicht wird und der Kellerspeicher am Ende leer ist.

Beispiel

Um die Sprache der korrekten Klammerausdrücke zu erkennen, wird der folgende Kellerautomat benötigt. $Z = \{A, B\}$, $\Sigma = \{(,)\}$, $K = \{(\}$, $z_0 = A$, $Z_E = \{B\}$. Die Übergangsfunktion kann am besten mit einer Tabelle beschrieben werden:



Grafik: Johann-Georg Vogelhuber

Zustand	Eingabe	Keller	Folgezustand	[illegible]
A	(	#	A	push (
A	(	(	A	push (
B	)	(	B	pop
B	)	(	[illegible]	pop

Aufgabe 1

Entwickeln Sie einen deterministischen endlichen Kellerautomaten, der wohlgeformte Klammerausdrücke erkennen kann. D. h. der Automat soll alle Ausdrücke akzeptieren, bei denen die Anzahl der öffnenden Klammern mit der Anzahl der schließenden Klammern übereinstimmt.

Aufgabe 2

Geben Sie eine Grammatik für die in Aufgabe 1 beschriebene Sprache an.

Aufgabe 3

Entwickeln Sie eine möglichst einfache kontextfreie Grammatik, mit der alle Terme erzeugt werden können, die aus Klammern, Ziffern und den vier Grundrechenarten bestehen. Verwenden sie dazu das Onlinetool flaci.com. Kontrollieren sie ihr Ergebnis mithilfe der Simulationsfunktion.