

Einführung in die Automatentheorie - Konzeption

Alexandra Maximova

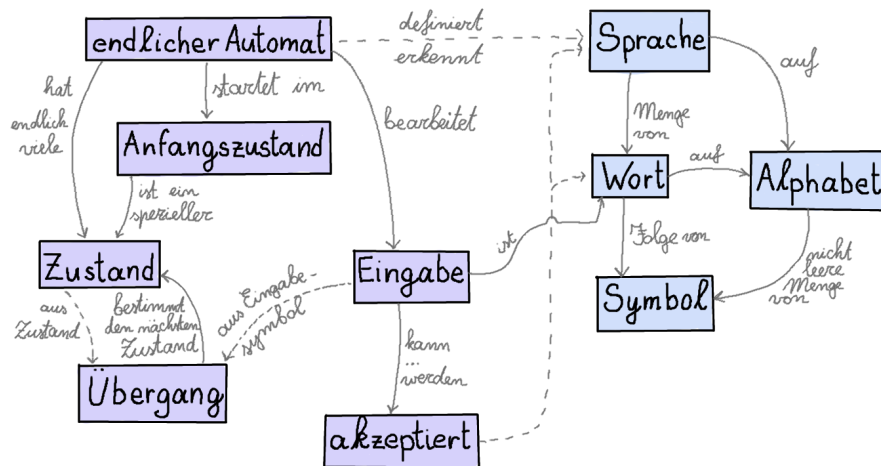
26. Oktober 2020

1 Rahmenbedingungen und Vorwissen

Diese Lektion wird im Rahmen vom Einführungspraktikum an der Kantonschule Menzingen gehalten und richtet sich an SuS des vierten Jahres am Kurzzeitgymnasium, die das Ergänzungsfach Informatik absolvieren.

In dieser Lektion geht es darum, endliche Automaten und wichtige Grundbegriffe einzuführen, so dass in den nächsten Lektionen darauf aufgebaut werden kann.

2 Stoffanalyse



Ein **endlicher Automat** ist ein Programm ohne Variablen, welches eine **Eingabe** einmal von links nach rechts liest und in Abhängigkeit nur vom aktuell eingelesenen **Symbol** und vom aktuellen **Zustand** ins nächste Zustand **übergeht**.

Das Programm startet die Ausführung im **Anfangszustand** und bestimmt den jeweils nächsten Zustand in Abhängigkeit vom eingelesenen Symbol und vom Zustand. Wenn der Automat nach der Bearbeitung einer Eingabe in einem der speziellen **akzeptierenden Zustände** endet, dann sagt man, dass der Automat diese Eingabe akzeptiert.

Damit wir einen endlichen Automat formaler und genauer definieren können, brauchen wir folgende Begriffe.

Ein **Alphabet** ist eine nichtleere Menge von Symbolen. Ein **Wort** auf einem Alphabet ist eine (möglicherweise leere) Folge von Symbolen aus dem Alphabet. Eine (formale) **Sprache** ist eine (möglicherweise leere) Menge von Wörter.

Formal können wir nun den endlichen Automaten als Quintupel $M = (Q, \Sigma, \delta, q_0, F)$ beschreiben, wobei Q eine endliche Menge der **Zustände**, Σ das **Eingabealphabet**, $\delta : Q \times \Sigma \rightarrow Q$ die **Übergangsfunktion**, $q_0 \in Q$ der **Anfangszustand** und $F \subseteq Q$ die Menge der **akzeptierenden Zustände** sind.

Ein endlicher Automat bearbeitet Wörter auf Σ . Er startet im Zustand q_0 und liest von links nach rechts die Eingabe Symbol für Symbol. Die Übergangsfunktion δ beschreibt, wie er dabei von einem Zustand zum anderen abhängig vom eingelesenen Symbol übergeht. Wenn der Automat sich nach der Ausführung auf der Eingabe w in einem Zustand aus F befindet, dann sagt man, dass der Automat w **akzeptiert**. Die Menge aller Wörter, die von einem Automaten akzeptiert werden, kann man als Sprache anschauen.

3 Lernziele

Leitidee Jeder von uns hat in der Tasche einen Computer, der mächtiger ist als der, der die Menschen auf den Mond brachte. Diese Computer können über Netz noch mehr Rechenkapazität beanspruchen. Wir haben das Gefühl, wir können alles berechnen, was wir uns nur vorstellen können. Aber lässt sich tatsächlich alles berechnen? Und was bedeutet es überhaupt, etwas zu berechnen?

Endliche Automaten bieten einen spannenden Einstieg in die mathematische Modellierung von Berechnungsarten. Sie sind an konkreten alltäglichen Objekten angelehnt, wie Aufzüge und Getränkeautomaten, und lassen sich elegant mathematisch formulieren. Die SuS sehen, wie alltägliche Objekte abstrahiert und formalisiert werden, und welche Erkenntnisse über die Existenz und Grösse von Automaten sie aus einer sauberen Formulierung gewinnen können.

Dispositionsziele

- Die Abstraktionsfähigkeiten der SuS werden gestärkt. Bei einer Problemstellung konzentrieren sich die SuS auf das Wesentliche und können unwichtige Details ignorieren.
- Die SuS schauen Automaten mit anderen Augen an. Wenn sie beim Kaffeeautomaten eine heisse Schokolade bestellen, denken sie während der Wartezeit, wie viele Zustände so ein Automat haben könnte.

Operationalisierte Lernziele

- Die SuS zeichnen einen in Worten beschriebenen Alltagsautomaten, wie z.B. ein Getränkeautomat, ein Parkautomat, ein Aufzug, usw. Dabei achten die SuS, dass
 - der Anfangszustand gekennzeichnet ist
 - die akzeptierende Zustände markiert sind
 - aus jedem Zustand gibt es für jedes in der Eingabe vorkommendes Symbol einen Pfeil zu einem anderen Zustand.
- Die SuS beschreiben in Worten die Funktionsweise von einem als Diagramm gegebenen Alltagsautomaten.
- Gegeben ein Diagramm von einem Automaten und eine Eingabe, die SuS sagen, ob die Eingabe akzeptiert wird oder nicht.
- Gegeben die formale Beschreibung von einem endlichen Automaten, wobei die Übergangsfunktion in tabellarischen Form gegeben wird, zeichnen die SuS das dazugehörige Diagramm.
- Gegeben einen Automaten als Diagramm, die SuS beschreiben ihn formal als Quintupel. Die Übergangsfunktion schreiben die SuS in tabellarischer Form auf.

4 Informativer Unterrichtseinstieg

In dieser Lektion werden wir herausfinden, was Aufzüge, Getränkeautomaten, Ampelschalter und Backöfen gemeinsam haben und wie wir solche Maschinen, auch **endliche Automaten** genannt, mathematisch modellieren können.

In der ersten Stunde werden wir uns mit dem Konzept von Automat vertraut machen, indem wir einige davon zuerst zusammen und dann in Gruppen entwerfen.

In der zweiten Stunde werden wir versuchen, diesen Begriff formal und mathematisch zu formulieren. Eine mathematische Formulierung erlaubt uns, ein Objekt genau und eindeutig zu definieren und seine Eigenschaften zu analysieren, Aussagen darüber zu machen und diese zu beweisen.

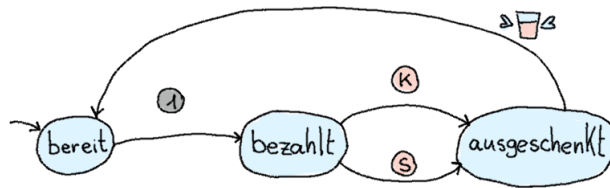
Nehmen wir zum Beispiel einen einfachen Kaffeeautomaten. Dieser Automat verkauft für einen Franken eine Tasse Kaffee oder eine Tasse heiße Schokolade. Der Automat akzeptiert nur 1CHF Münzen und hat zwei Knöpfe: Einen Knopf für Kaffee, einen Knopf für heiße Schokolade. Die Anweisungen für den Benutzer lauten:

1. Werfe eine 1CHF Münze.
2. Drücke die Taste "Kaffee" oder die Taste "heisse Schokolade".
3. Nehme deinen Becher aus der Maschine.



Wie würden wir eine solche Maschine Programmieren? Wie muss sich die Maschine verhalten?

Am Anfang wartet der Automat, dass der Benutzer eine Münze einwirft. Wenn eine 1CHF Münze kommt, dann wartet der Automat, welcher Knopf gedrückt wird. Nachdem ein Knopf selektiert wurde, bereitet der Automat das Getränk vor, lässt es unten erscheinen und wartet darauf, dass der Benutzer sein Getränk herausnimmt. Jetzt kann der Automat wieder auf eine Münze warten. Schematisch können wir diesen Verlauf so darstellen.



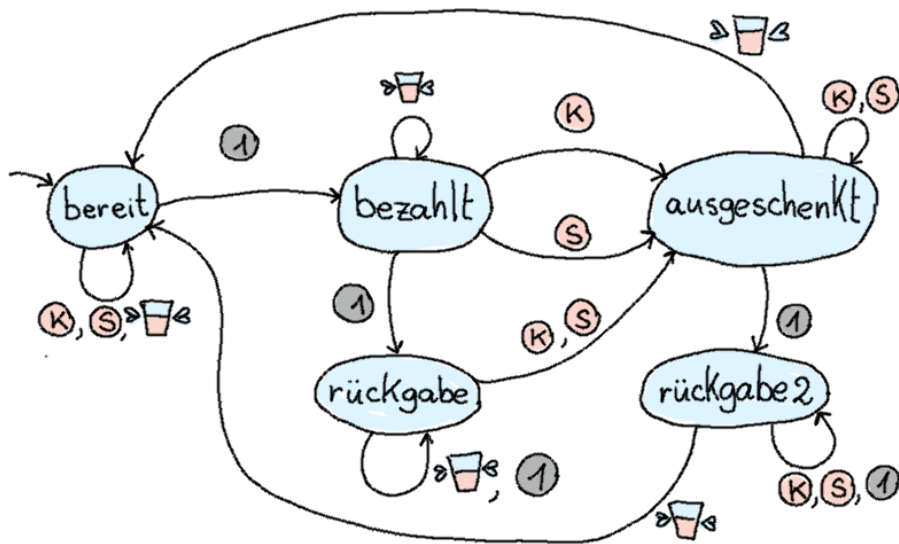
Die hellblaue Kreise stellen die **Zustände** des Automates dar. Im Zustand **bereit** wartet der Automat, dass eine Münze eingeworfen wird. Wenn eine Münze eintrifft (kleiner grauer Kreis mit einer '1' über dem Pfeil nach **bezahlt**), geht der Automat in den nächsten Zustand über. Im Zustand **bezahlt** wartet der Automat, bis einer der zwei Knöpfe betätigt wird. Wenn der Knopf Kaffee (kleiner roter Kreis mit 'K') oder der Knopf Schokolade (kleiner roter Kreis mit 'S') gedrückt wird, dann bereitet der Automat das entsprechende Getränk und

geht in den Zustand **ausgeschenkt** über. Im Zustand **ausgeschenkt** wartet der Automat, bis das Getränk verschwindet. Dann geht er wieder in den Zustand **bereit** über und wartet wieder auf die Münze.

Die kleinen Symbole auf den Pfeilen (1, K, S, ☕) stellen die **Eingabe** dar. Eine Eingabe ist etwas, was der Automat von aussen erhält. In diesem Fall, zur Eingabe gehören all die erwarteten und möglichen Aktionen vom Benutzer: Münzeneinwurf, Knopf Drücken, Becher Entnehmen.

Wie soll der Automat reagieren, wenn der Benutzer die Anweisungen nicht aufmerksam gelesen hat, und eine Eingabe zur falschen Zeit macht? Wenn der Benutzer, zum Beispiel, den Knopf 'Kaffee' betätigt, bevor er die Münze eingeworfen hat? Oder vergisst, den Becher zu entnehmen und schon die nächste Münze wirft? Diese Reaktionen auf 'Fehler' müssen im Automaten auch programmiert werden. Wir müssen das Diagramm so erweitern, dass aus jedem Zustand für jede mögliche Eingabe ein Pfeil zu einem (möglicherweise selben) Zustand führt.

Der einfache Kaffeeautomat aus unserem Beispiel ist ganz nett und gibt alle überflüssigen Münzen zurück. Um dieses Verhalten zu programmieren, brauchen wir noch zwei weitere Zustände: **rückgabe** und **rückgabe2**. Knöpfe, die zur falschen Zeit gedrückt werden, werden ignoriert.





In unserem Diagramm haben wir jetzt alle Zustände gezeichnet und für jeden Eingabesymbol haben wir aus jedem Zustand einen Pfeil gezeichnet. Das sind

aber noch nicht alle Informationen, die wir brauchen, um die Funktionalität vom Kaffeeautomaten zu beschreiben. Wir müssen noch definieren, in welchem Zustand der Automat ist, wenn er zum ersten Mal eingeschaltet wird. Dieser spezielle Zustand nennen wir **Anfangszustand** und kennzeichnen ihn auf dem Diagramm mit einem kurzen Pfeil aus dem nichts. Der Zustand **bereit** ist bereits als Anfangszustand gekennzeichnet.

Neben dem Anfangszustand, wird in Automaten auch eine andere Art von speziellen Zuständen hervorgehoben, und zwar die, wo eine korrekte Eingabe

endet. Im Fall von unserem Kaffeeautomaten, eine korrekte Eingabe ist 

(Münze werfen),  ('Kaffee' wählen),  (Becher nehmen). Nach der Bearbeitung dieser Eingabe befindet sich der Automat im Zustand **bereit**. Solche Zustände nennen wir **akzeptierende Zustände**. Auf dem Diagramm markiert man sie üblicherweise mit einem doppelten Kreis.

5 Literatur

Bei der Vorbereitung vom Skript und Unterricht wurde das Buch "Formale Sprachen" von J. Hromkovic verwendet.