

Kellerautomaten

Eine formale Sprache kann auf verschiedene Arten beschrieben werden. Begonnen haben wir mit umgangssprachlichen Beschreibungen oder durch Aufzählung aller Wörter einer Sprache. Diese eher intuitiven Festlegungen stoßen jedoch schnell an ihre Grenzen. Daher haben Sie endliche Automaten als Erkennendensysteme (Akzeptoren) sowie Grammatiken als Erzeugendensysteme zur Beschreibung einer Sprache kennengelernt.

Mittels Grammatiken können Sie sowohl reguläre als auch kontextfreie oder sogar kontextsensitive Sprachen erzeugen. Mit dem bisher bekannten Erkennendensystem des deterministischen endlichen Automaten (DEA) können jedoch nur reguläre Sprachen erkannt werden. Ziel der folgenden Überlegungen ist daher eine Erweiterung des Automatenmodells auf die Klasse der kontextfreien Sprachen.

Wiederholende Aufgabe

Wir haben gesehen, dass Sprachen, die aus Zeichenpaaren bestehen, die beliebig oft ineinander verschachtelt werden können, nicht regulär sind.

- 1) Begründen Sie, warum es keinen DEA für die Sprache $\{a^n b^n \mid n=1, 2, 3, \dots\}$ geben kann.

Idee eines Kellerautomaten

Die Idee eines Kellerautomaten ist es, das Konzept des endlichen Automaten um eine Art „unendlich großen Gedächtnisspeicher“ zu erweitern. Dafür verwendet man einen sogenannten Kellerspeicher – daher auch der Name Kellerautomat.

Ein Kellerspeicher oder Keller im Automatenmodell entspricht der Ihnen bekannten Datenstruktur Stapel mit den Operationen `Stack()`, `isEmpty()`, `top()`, `push()` und `pop()`.

Für diesen Kellerspeicher wird ein eigenes Kelleralphabet Γ sowie ein Kellervorbelegungszeichen, in unserem Fall das Zeichen `#`, festgelegt. Das Kellervorbelegungszeichen ist Teil des Kelleralphabets. Im Startzustand befindet sich lediglich das Kellervorbelegungszeichen `#` im Keller. Die Übergänge des Kellerautomaten hängen jetzt nicht mehr nur vom aktuellen Zustand und aktuellen Eingabezeichen ab, sondern auch vom obersten Zeichen des Kellerspeichers.

- 2) Diskutieren Sie mit Mitschülerinnen und Mitschülern, welche Vorteile die Hinzunahme eines Kellerspeichers bei der Modellierung einer Sprache haben könnte. Was vermuten Sie: Reicht diese Hinzunahme zur Modellierung der Sprache $\{a^n b^n \mid n=1, 2, 3, \dots\}$ aus?

Beispiel für einen Kellerautomaten

Bevor wir eine formale Definition eines Kellerautomaten geben, verdeutlichen wir uns zunächst das Prinzip am Beispiel der nicht regulären Sprache $\{a^n b^n \mid n=1, 2, 3, \dots\}$:

Eingabealphabet: $\Sigma=\{a, b\}$

Kellularphabet: $\Gamma=\{\#, A\}$ mit Keller-Vorbelegungszeichen #

Zustandsübergangsgraph:

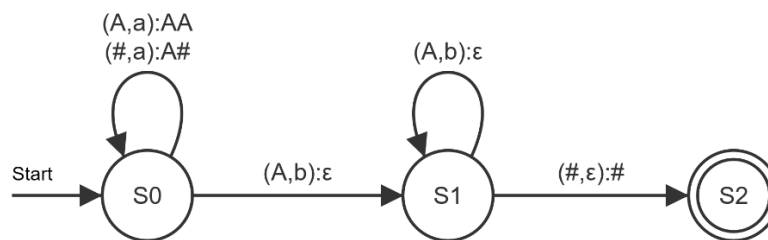


Abbildung 1

Bedeutung der Zustandsübergänge: An einem Zustandsübergang wird in Klammern zuerst das oberste Zeichen des Kellerspeichers gefolgt vom aktuellen Eingabezeichen notiert. Das vom Keller gelesene Zeichen wird bei der Verarbeitung vom Keller entfernt. Hinter dem Doppelpunkt wird angegeben, welche Zeichen im Keller abgelegt werden. Das am weitesten rechts notierte Zeichen wird dabei zuerst auf den Keller gelegt. Das Symbol ϵ steht dabei für das leere Wort.

Beispiele:

- Beim Übergang von S0 zu S1 mit der Beschriftung $(A,b):\epsilon$ liegt A oben auf dem Keller. Das aktuelle Eingabezeichen b wird verarbeitet, A vom Keller entfernt, nichts auf den Keller gelegt und in den Zustand S1 gewechselt.
- Beim Übergang von S1 zu S2 mit der Beschriftung $(\#, \epsilon):\#$ liegt # oben auf dem Keller. Es wird kein Zeichen verarbeitet, # vom Keller entfernt, # wieder auf den Keller gelegt und in den Zustand S2 gewechselt.

Akzeptierte Wörter: Eine Eingabe wird von dem Kellerautomaten aus Abbildung 1 genau dann akzeptiert, wenn der Kellerautomat sie so verarbeiten kann, dass er sich nach vollständiger Verarbeitung der Eingabe im Endzustand S2 befindet.

Beispiele:

- Der Kellerautomat kann die Eingabe ab verarbeiten, indem er zunächst das Zeichen a verarbeitet, indem er von S0 zu S0 wechselt, dabei # vom Stapel entfernt, anschließend erst # und danach das Zeichen A auf den Stapel legt. Im zweiten Schritt wird b verarbeitet, A vom Stapel entfernt, in den Zustand S1 gewechselt und nichts auf den Stapel zurückgelegt sowie abschließend in den Zustand S2 gewechselt. Die Eingabe wird so vollständig verarbeitet, ab daher vom Automaten akzeptiert.
- Bei der Eingabe aba befindet sich der Kellerautomat nach der Verarbeitung der ersten beiden Zeichen in Zustand S1 und wechselt anschließend in den Zustand S2 ohne Verarbeitung des letzten Zeichens. Dieses kann in S2 nicht weiterverarbeitet werden, da keine entsprechenden Übergänge definiert sind. Daher wird die Eingabe aba vom Kellerautomaten nicht akzeptiert.

Darstellung der Verarbeitung einer Eingabe

Die Verarbeitung eines Eingabewortes durch einen Kellerautomaten lässt sich ähnlich wie bei einem DEA durch Folgen von Konfigurationen darstellen, die der Automat nacheinander annimmt. Eine Konfiguration besteht immer aus dem aktuellen Zustand, der aktuellen Belegung des Kellersprechers sowie der noch zu verarbeitenden Eingabe. Für das Beispiel `aabb` ergibt sich als Konfigurationsfolge:

aktueller Zustand	S0	S0	S0	S1	S1	S2
Belegung des Kellers	#	A #	A A #	A #	#	#
noch zu verarbeitende Eingabe	aabb	abb	bb	b		

Da sich der Automat nach vollständiger Verarbeitung der Eingabe `aabb` in einem Endzustand befindet, wird diese Eingabe vom Automaten akzeptiert.

- 3) Untersuchen Sie unter Angabe zugehöriger Konfigurationsfolgen, ob die Eingaben `aaabbb`, `abb` und `bab` vom Kellerautomaten in Abbildung 1 akzeptiert werden. Simulieren Sie den Automaten und die Verarbeitung der gegebenen Eingaben anschließend mit dem Werkzeug flaci.com¹ (Link: <https://flaci.com/autoedit>) und vergleichen Sie die Ergebnisse.

Definition eines Kellerautomaten

Die Überlegungen zum Beispiel eines Kellerautomaten für die Sprache $\{a^n b^n \mid n=1, 2, 3, \dots\}$ wollen wir nun verallgemeinern²:

Ein **Kellerautomat** wird definiert durch die Angabe eines Eingabealphabets Σ , die Angabe eines Kellularphabetes Γ mit Vorbelegungszeichen $\#$ sowie der Darstellung eines Zustandsübergangsgraphen. Wie beim DEA besteht auch bei einem Kellerautomaten der Zustandsübergangsgraph aus einer endlichen Menge von Zuständen, die als Kreise dargestellt werden. Einer der Zustände ist durch einen Pfeil als Startzustand gekennzeichnet, außerdem gibt es einen oder mehrere Endzustände, gekennzeichnet durch eine doppelte Umrandung. Im Startzustand befindet sich zu Beginn lediglich das Kellerstartsymbol $\#$ im Keller. An einem Zustandsübergang wird in Klammern zuerst das oberste Zeichen des Kellers gefolgt vom aktuellen Eingabezeichen notiert. Das vom Speicher gelesene Zeichen wird bei der Verarbeitung aus dem Keller entfernt. Nach den Klammern wird ein Doppelpunkt angegeben. Hinter dem Doppelpunkt wird notiert, welche Zeichen im Kellerspeicher abgelegt werden. Dabei kann es sich um eines, keines (in diesem Fall schreibt man ϵ) oder sogar mehrere Zeichen handeln. Sind es mehrere Zeichen, so wird das am weitesten rechts notierte Zeichen zuerst auf den Keller gelegt. Außerdem sind sogenannte ϵ -Übergänge möglich.

¹ FLACI.com wird von der Pädagogischen Hochschule Schwyz kostenlos für Bildungszwecke zur Verfügung gestellt: <https://flaci.com> [Zugriff am: 06.06.2023]

² Die Definition orientiert sich an „Ergänzende Hinweisen zum Kerncurriculum Informatik für die gymnasiale Oberstufe am Gymnasium, an der Gesamtschule sowie für das Kolleg“, Juni 2025, herausgegeben vom Niedersächsischen Kultusministerium https://cuvo.nibis.de/cuvo.php?p=search&k0_6=Fach&v0_6=Informatik (Link vom 13.01.2026)

Diese hängen nur vom aktuellen Kellersymbol, nicht aber vom aktuellen Eingabezeichen ab (im Beispiel ist der Übergang $(\#, \epsilon): \#$ ein solcher ϵ -Übergang). Es wird festgelegt, dass eine Eingabe genau dann vom Automaten akzeptiert wird, wenn es eine Möglichkeit gibt, so dass sich der Automat **nach vollständiger Verarbeitung** der Eingabe in einem Endzustand befindet.

Hinweise:

- Es kann passieren, dass der Kellerautomat wegen einer fehlenden Übergangsregel in einem Zustand verbleibt, **ohne** dass eine Eingabe vollständig verarbeitet wurde. In diesem Fall wird die Eingabe für die entsprechende Konfigurationsfolge nicht akzeptiert (auch wenn der aktuelle Zustand zufällig ein Endzustand ist).
- Während beim deterministischen endlichen Automaten (DEA) für jede Situation genau ein Übergang festgelegt ist, erlaubt ein Kellerautomat mehrere mögliche Übergänge für dieselbe Konfiguration (bestehend aus aktuellem Zustand, Belegung des Kellers und noch zu verarbeitender Eingabe). Bei der Verarbeitung einer Eingabe ist der gewählte Übergang daher gegebenenfalls nicht eindeutig. Ein Kellerautomat darf also nichtdeterministisch sein. Diesen Aspekt können Sie im Abschnitt „Nichtdeterministische Kellerautomaten“ vertiefen. Wir betrachten im Folgenden zunächst Beispiele von Kellerautomaten, in denen die zu wählenden Übergänge eindeutig sind (deterministische Kellerautomaten).

Aufgaben:

4)

a) In Abbildung 2 ist erneut der Kellerautomat zur Sprache $\{a^n b^n \mid n=1, 2, 3, \dots\}$ mit Eingabealphabet: $\Sigma=\{a, b\}$, Kelleralphabet: $\Gamma=\{\#, A\}$ mit Keller-Vorbelegungszeichen $\#$ und dem Übergangsgraphen abgebildet.

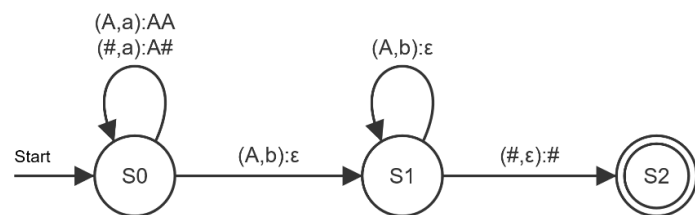


Abbildung 2: Kellerautomat zur Sprache $\{a^n b^n \mid n=1, 2, 3, \dots\}$

Erläutern Sie, wie der Vergleich der Anzahl des Zeichens a mit der Anzahl des Zeichens b in diesem Kellerautomaten modelliert wird.

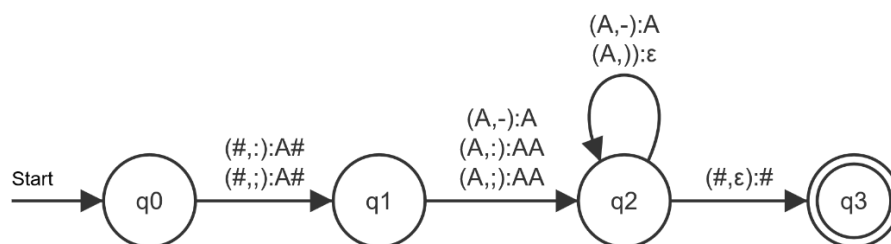
b) Ändern Sie den Kellerautomaten aus Abbildung 2 so, dass dieser die Sprache $\{a^n b^n \mid n=0, 1, 2, 3, \dots\}$ akzeptiert.

5) Gegeben ist der folgende Kellerautomat

Eingabealphabet: $\Sigma=\{;, :, -, ,\}$

Kelleralphabet: $\Gamma=\{\#, A\}$ mit Keller-Vorbelegungszeichen $\#$

Übergangsgraph:



a) Zeigen Sie, dass der Automat die Wörter $; -)$ und $; : -) -)$ akzeptiert.

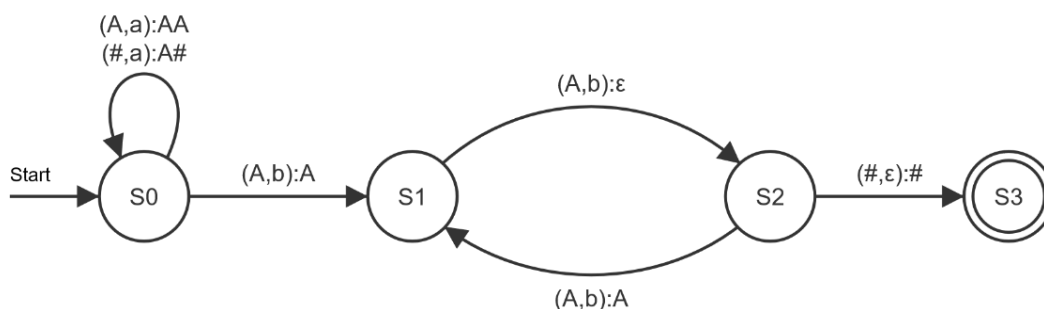
- b) Geben Sie zwei Wörter über dem Eingabealphabet an, die der Automat nicht akzeptiert.
- c) Beschreiben Sie, welche Rolle der Zustand q_2 bei der Entscheidung spielt, ob eine Eingabe akzeptiert wird oder nicht.
- d) Untersuchen Sie, ob es einen deterministischen endlichen Automaten gibt, der die gleiche Sprache akzeptiert wie der abgebildete Kellerautomat.

6) Gegeben ist der folgende Kellerautomat:

Eingabealphabet: $\Sigma = \{a, b\}$

Kellularphabet: $\Gamma = \{\#, A\}$ mit Keller-Vorbelegungszeichen $\#$

Übergangsgraph:



Beschreiben Sie die von diesem Kellerautomaten akzeptierte Sprache mit eigenen Worten.

- 7)
 - a) Geben Sie eine kontextfreie Grammatik an, die die Sprache $\{a^n b c^n \mid n=1, 2, 3, \dots\}$ erzeugt.
 - b) Entwickeln Sie außerdem einen Kellerautomaten zur Modellierung dieser Sprache.
- 8) Bei der Definition eines Kellerautomaten muss immer ein Vorbelegungszeichen des Kellers angegeben werden, welches im Startzustand zu Beginn bereits im Keller liegt. In unseren Beispielen wird als Vorbelegungszeichen $\#$ verwendet. Begründen Sie, warum man bei der Definition eines Kellerautomaten nicht auf die Angabe eines Vorbelegungszeichens verzichten kann.
- 9)
 - a) Entwickeln Sie einen Kellerautomaten mit dem Eingabealphabet $\Sigma = \{ (,) \}$, welcher gültige Klammerausdrücke für Taschenrechnereingaben akzeptiert. Beispiele für korrekte Klammerausdrücke sind: $() () ()$, $(()) ()$ oder $((()))$. Ungültige Ausdrücke sind dagegen beispielsweise $))$, $(()$ oder $() () ($.
 - b) Geben Sie eine kontextfreie Grammatik an, die die gleiche Sprache erzeugt.
 - c) Begründen Sie, warum es keinen deterministischen endlichen Automaten zur Modellierung der Sprache gibt.

10) Beurteilen Sie die Gültigkeit der folgenden Aussagen:

- Behauptung: *Lässt sich eine formale Sprache durch einen deterministischen endlichen Automaten modellieren, so existiert auch ein Kellerautomat, der diese Sprache akzeptiert.*
- Behauptung: *Lässt sich eine formale Sprache durch einen Kellerautomaten modellieren, so existiert auch ein deterministischer endlicher Automat, der diese Sprache akzeptiert.*

Nichtdeterministische Kellerautomaten

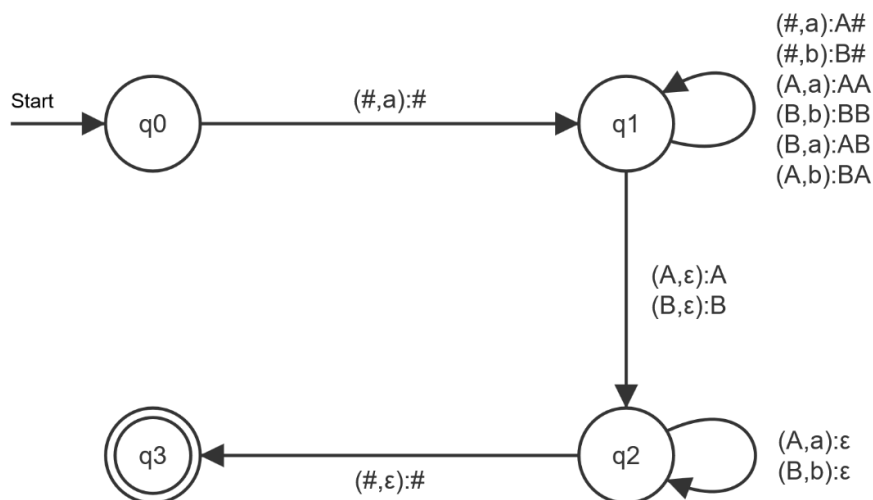
Bei unseren bisherigen Beispielen gab es für jede Konfiguration (bestehend aus aktuellem Zustand, Belegung des Kellers und noch zu verarbeitender Eingabe) immer maximal einen Zustandsübergang. Laut unserer Definition erlaubt ein Kellerautomat aber auch mehrere mögliche Übergänge für dieselbe Konfiguration. Bei der Verarbeitung einer Eingabe ist der gewählte Übergang daher gegebenenfalls nicht eindeutig. Der Kellerautomat ist in dem Fall nichtdeterministisch.

11) Gegeben ist der folgende Kellerautomat³ mit

Eingabealphabet: $\Sigma = \{a, b\}$

Kelleralphabet: $\Gamma = \{\#, A, B\}$ mit Keller-Vorbelegungszeichen #

Übergangsgraph:



- Sogenannte ε -Übergänge hängen nur vom aktuellen Kellersymbol, nicht aber vom aktuellen Eingabezeichen ab. Erklären Sie am Beispiel des obersten Kellersymbols A im Zustand q1, was es für die Verarbeitung einer Eingabe bedeutet, wenn von einem Zustand ein ε -Übergang und zusätzlich weitere Übergänge mit demselben aktuellen Kellersymbol ausgehen.
- Öffnen Sie die Vorlage Automaton_HinweiseBeispiel.json in flaci und simulieren Sie dort die Verarbeitung verschiedener Eingaben. Erklären Sie, warum es bei diesem Kellerautomaten für Eingaben wie aab oder abaab mehrere verschiedene Konfigurationsfolgen gibt.

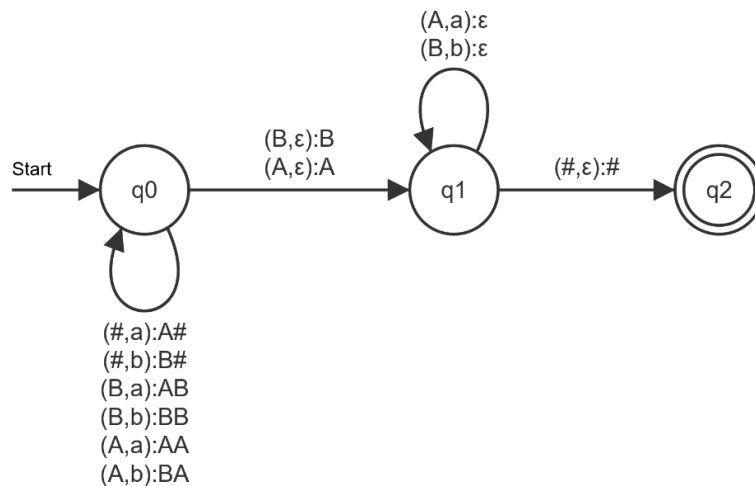
³ Vgl. S. 12 in „Ergänzende Hinweisen zum Kerncurriculum Informatik für die gymnasiale Oberstufe am Gymnasium, an der Gesamtschule sowie für das Kolleg“, Juni 2025, herausgegeben vom Niedersächsischen Kultusministerium https://cuvo.nibis.de/cuvo.php?p=search&k0_6=Fach&v0_6=Informatik (Link vom 13.01.2026)

12) Gegeben ist der folgende Kellerautomat mit

Eingabealphabet: $\Sigma = \{a, b\}$

Kellularphabet: $\Gamma = \{\#, A, B\}$ mit Keller-Vorbelegungszeichen #

Übergangsgraph:



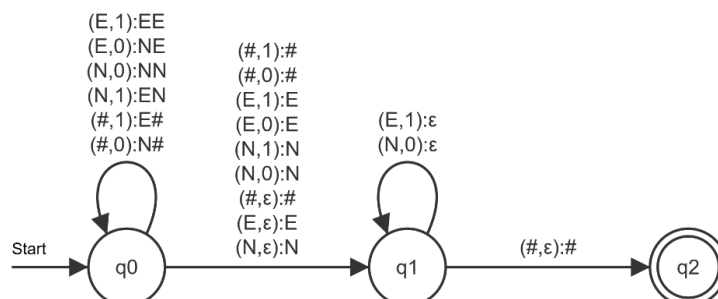
- Zeigen Sie, dass der Kellerautomat die Wörter *aabbaa* und *ababaababa* akzeptiert.
- Zeigen Sie, dass der Kellerautomat die Wörter *ab* und *baba* nicht akzeptiert.
- Der Kellerautomat akzeptiert Palindrome, also Zeichenketten, die vorwärts und rückwärts gelesen das Gleiche ergeben. Hier sind es spezielle Palindrome, die nur aus den Zeichen *a* und *b* bestehen und insgesamt eine gerade Anzahl an Zeichen enthalten. Entwickeln Sie eine kontextfreie Grammatik, die die gleiche Sprache erzeugt.
- Der gegebene Kellerautomat ist nichtdeterministisch. Begründen Sie, warum es für die durch ihn definierte Sprache keinen deterministischen Kellerautomaten gibt.

13) Gegeben ist der folgende Kellerautomat mit

Eingabealphabet: $\Sigma = \{0, 1\}$

Kellularphabet: $\Gamma = \{\#, E, N\}$ mit Vorbelegungszeichen #

Übergangsgraph:



- Untersuchen Sie, ob der Kellerautomat die Wörter *01011010*, *001*, *1*, *101*, *110* sowie das leere Wort jeweils akzeptiert.
- Geben Sie die vom Kellerautomaten akzeptierte Sprache an.
- Erläutern Sie, warum man bei diesem Automaten von einem **nichtdeterministischen** Kellerautomaten spricht. Gibt es auch einen deterministischen Kellerautomaten für diese Sprache?

Zusammenhang zwischen Kellerautomaten und kontextfreien Grammatiken

Kontextfreie Sprachen, also Sprachen, die von einer kontextfreien Grammatik erzeugt werden, erweitern die Sprachklasse der regulären Sprachen. Die Erweiterung des Automatenmodells DEA auf das Modell eines nichtdeterministischen Kellerautomaten (NKA) erweitert die Menge der von solchen Automaten akzeptierten Sprachen auf die Sprachklasse der kontextfreien Sprachen. Es gibt also zu jeder von einer kontextfreien Grammatik erzeugten Sprache einen entsprechenden nichtdeterministischen Kellerautomaten, der die Sprache akzeptiert, und umgekehrt zu jedem NKA eine entsprechende kontextfreie Grammatik.

14) Wir betrachten eine ASCII-Art-Sprache L_{Berg} von „Bergpanoramen“ (vgl. Abbildung 3). Gegeben ist dafür folgende Grammatik (Zeilenwechsel werden in dieser Grammatik nicht berücksichtigt):

$G=(V, \Sigma, P, A)$ mit

$V=\{A\}$

$\Sigma=\{/, \backslash, _ \}$

Produktionen P :

$A \rightarrow AA \mid /A \mid _ \mid \varepsilon$

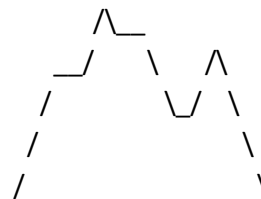


Abbildung 3: Beispiel für ein „Bergpanorama“

- Geben Sie zwei Wörter an, die von der Grammatik erzeugt werden.
- Konstruieren Sie einen Kellerautomaten für die Sprache L_{Berg} .
- Erläutern Sie, wie in der Grammatik und wie im Kellerautomaten jeweils dafür gesorgt wird, dass ein Wort gleich viele / und \ enthält.

15) Eine 2-Euro-Münze hat eine Wertseite (2) und eine nationale Motivseite (m). Bei einem Münzspiel legt eine Person zu Beginn des Spiels fest, wie oft eine 2-Euro-Münze geworfen werden soll. Es müssen mindestens zwei Würfe sein. Die Person gewinnt, wenn in der Wurfserie die Wertseite (2) genauso oft vorkommt wie die nationale Motivseite (m).

Beispiel: Die Wurfserie mm2m2m22 bedeutet „Gewinn“.

- Entwickeln Sie eine kontextfreie Grammatik, die die Sprache der Wurfserien, die einen Gewinn darstellen, erzeugt.
- Entwickeln Sie einen Kellerautomaten, der genau die Sprache der Wurfserien, die einen Gewinn darstellen, akzeptiert.

Ausblick: Grenzen von Kellerautomaten

Auch das Modell eines Kellerautomaten hat Grenzen:

16) Mit den Zeichen /, \ und _ kann man die folgenden „ausgeglichene Berg- und Talpanoramen“ zeichnen. Damit sind Zeichenfolgen gemeint, die aus jeweils gleich vielen /-Zeichen, \-Zeichen und _-Zeichen bestehen, z.B. //__/_ oder __/__/. Begründen Sie, warum kein Kellerautomat existiert, der diese Sprache beschreibt.

Hinweis

Die Materialien erheben keinen Anspruch auf Vollständigkeit hinsichtlich der für die Abiturprüfung erwarteten Kompetenzen. Verbindlich für das Abitur in Niedersachsen sind allein das niedersächsische Kerncurriculum für die gymnasiale Oberstufe sowie die ergänzenden Hinweise in der jeweils aktuellen Fassung.

Lizenz

Dieses Werk ist lizenziert unter einer [Creative Commons Namensnennung - Nicht-kommerziell - Weitergabe unter gleichen Bedingungen 4.0 International Lizenz](#). Sie erlaubt Bearbeitungen und Weiterverteilung des Werks unter Nennung meines Namens und unter gleichen Bedingungen, jedoch keinerlei kommerzielle Nutzung.

Abbildungsnachweise: Die Abbildungen wurden mithilfe der Software FLACI.com erzeugt. FLACI.com wird von der Pädagogischen Hochschule Schwyz kostenlos für Bildungszwecke zur Verfügung gestellt: <https://flaci.com> (Zugriff vom 13.01.2026)