17 RELATIONEN

17.1 ÄQUIVALENZRELATIONEN

17.1.1 Definition

In Abschnitt 11.2.1 hatten wir schon einmal erwähnt, dass eine Relation $R \subseteq M \times M$ auf einer Menge M, die

- reflexiv,
- symmetrisch und
- transitiv

ist, *Äquivalenzrelation* heißt. Das bedeutet also, dass

Äquivalenzrelation

- für alle $x \in M$ gilt: $(x, x) \in R$
- für alle $x, y \in M$ gilt: wenn $(x, y) \in R$, dann auch $(y, x) \in R$
- für alle $x, y, z \in M$ gilt: wenn $(x, y) \in R$ und $(y, z) \in R$, dann auch $(x, z) \in R$.

Für Äquivalenzrelationen benutzt man oft Symbole wie $\equiv$, $\sim$ oder $\approx$, die mehr oder weniger deutlich an das Gleichheitszeichen erinnern, sowie Infixschreibweise. Dann liest sich die Definition so: Eine Relation $\equiv$ ist Äquivalenzrelation auf einer Menge M, wenn gilt:

- $\forall x \in M : x \equiv x$,
- $\forall x \in M : \forall y \in M : x \equiv y \Longrightarrow y \equiv x$
- $\forall x \in M : \forall y \in M : \forall z \in M : x \equiv y \land y \equiv z \Longrightarrow x \equiv z$

Der Anlass für Symbole, die an das „=“ erinnnern, ist natürlich der, dass Gleichheit, also die Relation $\mathrm{I} = \{(x, x) \mid x \in M\}$, auf jeder Menge eine Äquivalenzrelation ist, denn offensichtlich gilt:

- $\forall x \in M : x = x$,
- $\forall x \in M : \forall y \in M : x = y \Longrightarrow y = x$
- $\forall x \in M : \forall y \in M : \forall z \in M : x = y \land y = z \Longrightarrow x = z$

Ein klassisches Beispiel sind die „Kongruenzen modulo n“ auf den ganzen Zahlen. Es sei $n \in \mathbb{N}_+$. Zwei Zahlen $x, y \in \mathbb{Z}$ heißen *kongruent modulo n*, wenn die Differenz $x - y$ durch n teilbar, also ein ganzzahliges Vielfaches von n, ist. Man schreibt typischerweise $x \equiv y \pmod{n}$. Dass dies tatsächlich Äquivalenzrelationen sind, haben Sie in Mathematikvorlesungen mehr oder weniger explizit gesehen.

kongruent modulo n

- Die Reflexivität ergibt sich aus der Tatsache, dass $x - x = 0$ Vielfaches von n ist.
- Die Symmetrie gilt, weil mit $x - y$ auch $y - x = -(x - y)$ Vielfaches von n ist.
- Transitivität: Wenn $x - y = k_1 n$ und $y - z = k_2 n$ (mit $k_1, k_2 \in \mathbb{Z}$), dann ist auch $x - z = (x - y) + (y - z) = (k_1 + k_2)n$ ein ganzzahliges Vielfaches von n.

17.1.2 Äquivalenzrelationen von Nerode

Äquivalenzrelation von Nerode

Als ein durchgehendes Beispiel in diesem und weiteren Abschnitten betrachten wir eine Äquivalenzrelation $\equiv_L$, die durch eine formale Sprache $L \subseteq A^*$ auf der Menge A^* aller Wörter induziert wird. Sie heißt die *Äquivalenzrelation von Nerode*. Die Relation ist wie folgt definiert. Für alle $w_1, w_2 \in A^*$ ist

$$w_1 \equiv_L w_2 \iff \big(\forall w \in A^* : w_1 w \in L \iff w_2 w \in L\big)$$

Wenn man das erste Mal mit dieser Definition konfrontiert ist, muss man sie erfahrungsgemäß mehrfach lesen. Zwei Wörter w_1 und w_2 sind dann und nur dann äquivalent, wenn gilt: Gleich, welches Wort $w \in A^*$ man an die beiden anhängt, immer sind entweder beide Produkte $w_1 w$ und $w_2 w$ in L, oder keines von beiden. Anders gesagt: Zwei Wörter sind genau dann *nicht* $\equiv_L$-äquivalent, wenn es ein Wort $w \in A^*$ gibt, so dass genau eines der Wörter $w_1 w$ und $w_2 w$ in L liegt, aber das andere nicht.

Betrachtet man das leere Wort $w = \varepsilon$, dann ergibt sich insbesondere, dass $w_1 \equiv_L w_2$ höchstens dann gelten kann, wenn beide Wörter in L liegen, oder beide nicht in L.

Nehmen wir als Beispiel das Alphabet $A = \{\mathtt{a}, \mathtt{b}\}$ und die formale Sprache $L = \langle \mathtt{a*b*} \rangle \subset A^*$ aller Wörter, in denen nirgends das Teilwort `ba` vorkommt. Versuchen wir anhand einiger Beispielpaare von Wörtern, ein Gefühl dafür zu bekommen, welche Wörter $\equiv_L$-äquivalent sind und welche nicht.

1. $w_1 = \mathtt{aaa}$ und $w_2 = \mathtt{a}$:
   - Hängt man an beide Wörter ein $w \in \langle \mathtt{a*} \rangle$ an, dann sind sowohl $w_1 w$ als auch $w_2 w$ in L.
   - Hängt man an beide Wörter ein $w \in \langle \mathtt{a*bb*} \rangle$ an, dann sind ebenfalls wieder sowohl $w_1 w$ als auch $w_2 w$ in L.
   - Hängt man an beide Wörter ein w an, das das Teilwort `ba` enthält, dann enthalten $w_1 w$ und $w_2 w$ beide `ba`, sind also beide nicht in L.
   - Andere Möglichkeiten für ein Suffix w gibt es nicht, also sind die beiden Wörter $\equiv_L$-äquivalent.
2. $w_1 = \mathtt{aaab}$ und $w_2 = \mathtt{abb}$:
   - Hängt man an beide Wörter ein $w \in \langle \mathtt{b*} \rangle$ an, dann sind sowohl $w_1 w$ als auch $w_2 w$ in L.
   - Hängt man an beide Wörter ein w an, das ein `a` enthält, dann enthalten $w_1 w$ und $w_2 w$ beide das Teilwort `ba`, sind also beide nicht in L.
   - Andere Möglichkeiten gibt es nicht, also sind die beiden Wörter $\equiv_L$-äquivalent.
3. $w_1 = \mathtt{aa}$ und $w_2 = \mathtt{abb}$:

- Hängt man an beide Wörter $w = \mathtt{a}$ an, dann ist zwar $w_1w = \mathtt{aaa} \in L$, aber $w_2w = \mathtt{abba} \notin L$.
- Also sind die beiden Wörter *nicht* $\equiv_L$-äquivalent.

4. $w_1 = \mathtt{aba}$ und $w_2 = \mathtt{babb}$:
   - Beide Wörter enthalten $\mathtt{ba}$. Egal welches $w \in A^*$ man anhängt, bleibt das so, d. h. immer sind $w_1w \notin L$ und $w_2w \notin L$.
   - Also sind die beiden Wörter $\equiv_L$-äquivalent.
5. $w_1 = \mathtt{ab}$ und $w_2 = \mathtt{ba}$:
   - Da $w_1 \in L$, aber $w_2 \notin L$, zeigt schon das Suffix $w = \varepsilon$, dass die beiden Wörter nicht $\equiv_L$-äquivalent sind.

Die wesentliche Behauptung ist nun:

17.1 Lemma. Für jede formale Sprache L ist $\equiv_L$ eine Äquivalenzrelation.

17.2 Beweis. Man prüft nach, dass die drei definierenden Eigenschaften erfüllt sind.

- Reflexivität: Ist $w_1 \in A^*$, dann gilt für jedes $w \in A^*$ offensichtlich: $w_1w \in L \iff w_1w \in L$.
- Symmetrie: Für $w_1, w_2 \in A^*$ und alle $w \in A^*$ gelte: $w_1w \in L \iff w_2w \in L$. Dann gilt offensichtlich auch immer $w_2w \in L \iff w_1w \in L$.
- Transitivität: Es seien $w_1, w_2, w_3 \in A^*$ und es möge gelten

$$\forall w \in A^* : w_1w \in L \iff w_2w \in L \tag{17.1}$$

$$\forall w \in A^* : w_2w \in L \iff w_3w \in L \tag{17.2}$$

  Wir müssen zeigen: $\forall w \in A^* : w_1w \in L \iff w_3w \in L$. Sei dazu ein beliebiges $w \in A^*$ gegeben. Falls $w_1w \in L$ ist, dann ist wegen (17.1) auch $w_2w \in L$ und daher wegen (17.2) auch $w_3w \in L$. Analog folgt aus $w_1w \notin L$ der Reihe nach $w_2w \notin L$ und $w_3w \notin L$. Also gilt $w_1w \in L \iff w_3w \in L$.

■

17.1.3 Äquivalenzklassen und Faktormengen

Für $x \in M$ heißt $\{y \in M \mid x \equiv y\}$ die *Äquivalenzklasse von x*. Man schreibt für die Äquivalenzklasse von x mitunter $[x]_\equiv$ oder einfach $[x]$, falls klar ist, welche Äquivalenzrelation gemeint ist. *Äquivalenzklasse*

Für die Menge aller Äquivalenzklassen schreibt man $M_{/\equiv}$ und nennt das manchmal auch die *Faktormenge* oder *Faserung von M nach $\equiv$*, also $M_{/\equiv} = \{[x]_\equiv \mid x \in M\}$. *Faktormenge* *Faserung*

Ist konkret $\equiv$ die Äquivalenzrelation „modulo n" auf den ganzen Zahlen, dann schreibt man für die Faktormenge auch $\mathbb{Z}_n$.

Mitunter ist es nützlich, sich anzusehen aus wievielen Äquivalenzklassen eine Faserung besteht. Nehmen wir als Beispiel wieder die durch $L = \langle \texttt{a*b*} \rangle$ induzierte Nerode-Äquivalenz $\equiv_L$. Schaut man sich noch einmal die Argumentationen im vorangegangenen Abschnitt an, dann merkt man, dass jedes Wort zu genau einem der drei Wörter ε, b und ba äquivalent ist. Mit anderen Worten besteht $A^*_{/\equiv_L}$ aus drei Äquivalenzklassen:

- $[\varepsilon] = \langle \texttt{a*} \rangle$
- $[\texttt{b}] = \langle \texttt{a*bb*} \rangle$
- $[\texttt{ba}] = \langle \texttt{a*bb*a(a|b)*} \rangle$

Die Wahl der Repräsentanten in dieser Aufzählung ist natürlich willkürlich. Wir hätten genauso gut schreiben können:

- $[\texttt{aaaaa}] = \langle \texttt{a*} \rangle$
- $[\texttt{aabbbbb}] = \langle \texttt{a*bb*} \rangle$
- $[\texttt{aabbaabbba}] = \langle \texttt{a*bb*a(a|b)*} \rangle$

Die durch eine formale Sprache L induzierte Nerode-Äquivalenz hat aber nicht immer nur endlich viele Äquivalenzklassen. Als Beispiel betrachte man das schon in Abschnitt 14.4 diskutierte

$$L = \{\texttt{a}^k\texttt{b}^k \mid k \in \mathbb{N}_0\} \ .$$

Für diese Sprache besteht $A^*_{/\equiv_L}$ aus unendlich vielen Äquivalenzklassen. Ist nämlich $k \neq m$, dann sind $w_1 = \texttt{a}^k$ und $w_2 = \texttt{a}^m$ nicht äquivalent, wie man durch Anhängen von $w = \texttt{b}^k$ sieht:

- $w_1 w = \texttt{a}^k\texttt{b}^k \in L$, aber
- $w_2 w = \texttt{a}^m\texttt{b}^k \notin L$.

Also ist zumindest jedes Wort $\texttt{a}^k$, $k \in \mathbb{N}_0$ in einer anderen Äquivalenzklasse. Jede dieser Äquivalenzklassen ist aber ihrerseits unendlich groß, denn Sie können sich überlegen, dass

$$[\texttt{a}^k]_{\equiv_L} = \{\texttt{a}^x\texttt{b}^y \mid x, y \in \mathbb{N}_0 \wedge x - y = k\}$$

ist.

Vielleicht lässt die Tatsache, dass es für die reguläre Sprache $\langle \texttt{a*b*} \rangle$ endlich viele Äquivalenzklassen gibt, aber für die nicht reguläre Sprache $L = \{\texttt{a}^k\texttt{b}^k \mid k \in \mathbb{N}_0\}$ unendlich viele, Sie schon etwas ahnen.

17.2 KONGRUENZRELATIONEN

Mitunter hat eine Menge M, auf der eine Äquivalenzrelation definiert ist, zusätzliche „Struktur“, bzw. auf M sind eine oder mehrere Operationen definiert. Als Beispiel denke man etwa an die ganzen Zahlen $\mathbb{Z}$ mit der Addition. Man kann sich dann z. B. fragen, wie sich Funktionswerte ändern, wenn man Argumente durch andere, aber äquivalente ersetzt.

17.2.1 Verträglichkeit von Relationen mit Operationen

Um den Formalismus nicht zu sehr aufzublähen, beschränken wir uns in diesem Unterabschnitt auf die zwei am häufigsten vorkommenden einfachen Fälle.

Es sei $\equiv$ eine Äquivalenzrelation auf einer Menge M und $f : M \to M$ eine Abbildung. Man sagt, dass $\equiv$ mit f *verträglich* ist, wenn für alle $x_1, x_2 \in M$ gilt: *verträglich*

$$x_1 \equiv x_2 \Longrightarrow f(x_1) \equiv f(x_2) .$$

Ist $\square$ eine binäre Operation auf einer Menge M, dann heißen $\equiv$ und $\square$ *verträglich* , wenn für alle $x_1, x_2 \in M$ und alle $y_1, y_2 \in M$ gilt: *verträglich*

$$x_1 \equiv x_2 \land y_1 \equiv y_2 \Longrightarrow x_1 \square y_1 \equiv x_2 \square y_2 .$$

Ein typisches Beispiel sind wieder die Äquivalenzrelationen „modulo n“. Diese Relationen sind mit Addition, Subtraktion und Multiplikation verträglich. Ist etwa

$$\begin{array}{rlll} & x_1 \equiv x_2 \pmod n & \text{also} & x_1 - x_2 = kn \\ \text{und} & y_1 \equiv y_2 \pmod n & \text{also} & y_1 - y_2 = mn \end{array}$$

dann ist zum Beispiel

$$(x_1 + y_1) - (x_2 + y_2) = (x_1 - x_2) + (y_1 - y_2) = (k + m)n .$$

Mit anderen Worten ist dann auch

$$x_1 + y_1 \equiv x_2 + y_2 \pmod n .$$

Eine Äquivalenzrelation, die mit allen gerade interessierenden Funktionen oder/und Operationen verträglich ist, nennt man auch eine *Kongruenzrelation*. *Kongruenzrelation*

Auch die Nerode-Äquivalenzen haben eine solche Eigenschaft. Sei $w' \in A^*$ ein beliebiges Wort und sei $f_{w'} : A^* \to A^*$ die Abbildung, die w' an ihr Argument anhängt, also $f_{w'}(v) = vw'$. Wir behaupten, dass $\equiv_L$ mit allen $f_{w'}$ verträglich ist. d. h.:

$$\forall w_1, w_2 \in A^* : w_1 \equiv_L w_2 \Longrightarrow w_1 w' \equiv_L w_2 w'$$

Wir müssen zeigen: Wenn $w_1 \equiv_L w_2$ ist, dann ist auch $w_1 w' \equiv_L w_2 w'$. Gehen wir also davon aus, dass für *alle* $w \in A^*$ gilt: $w_1 w \in L \iff w_2 w \in L$. Wir müssen zeigen, dass für alle $v \in A^*$ gilt: $(w_1w')v \in L \iff (w_2w')v \in L$. Das geht ganz einfach. Sei $v \in A^*$ beliebig; dann gilt

$$\begin{aligned} (w_1w')v \in L &\iff w_1(w'v) \in L \\ &\iff w_2(w'v) \in L \qquad \text{weil } w_1 \equiv_L w_2 \\ &\iff (w_2w')v \in L \,. \end{aligned}$$

17.2.2 Wohldefiniertheit von Operationen mit Äquivalenzklassen

induzierte Operation

Wann immer man eine Kongruenzrelation vorliegen hat, also z. B. eine Äquivalenzrelation $\equiv$ auf M die mit einer binären Operation $\square$ auf M verträglich ist, *induziert* diese Operation auf M eine Operation auf $M_{/\equiv}$. Analoges gilt für Abbildungen $f : M \to M$.

Betrachten wir wieder die Nerode-Äquivalenzen. L sei wie immer eine beliebige formale Sprache $L \subseteq A^*$. Eben hatten wir uns überlegt, dass insbesondere für jedes $x \in A$ die Abbildung $f_x : A^* \to A^* : w \mapsto wx$ mit $\equiv_L$ verträglich ist.

Wir schreiben nun einmal hin:

$$f'_x : A^*_{/\equiv_L} \to A^*_{/\equiv_L} : [w] \mapsto [wx]$$

Der ganz entscheidende Punkt ist: Dies ist eine vernünftige Definition. Wenn Sie so etwas zum ersten Mal sehen, fragen Sie sich vielleicht, warum es überhaupt Unsinn sein könnte. Nun: Es wird hier versucht eine Abbildung zu definieren, die jede Äquivalenzklasse auf eine Äquivalenzklasse abbildet. Aber die durch $[w]$ beschriebene Klasse enthält ja im allgemeinen nicht nur w, sondern noch viele andere Wörter. Zum Beispiel hatten wir uns weiter vorne überlegt, dass im Fall $L = \langle \texttt{a*b*} \rangle$ die Wörter ε, $\texttt{a}$, $\texttt{a}^2$, $\texttt{a}^3$, usw. alle in einer Äquivalenzklasse liegen. Es ist also $[\varepsilon] = [\texttt{a}] = [\texttt{a}^2] = \cdots$. D. h., damit das, was wir eben für f'_x hingeschrieben haben, wirklich eine Definition ist, die für jedes Argument *eindeutig* einen Funktionswert festlegt, sollte dann bitte auch $[\varepsilon x] = [\texttt{a}x] = [\texttt{a}^2x] = \cdots$ sein. Und das ist so, denn hier hinter steckt nichts anderes als die Forderung

$$\begin{aligned} & w_1 \equiv_L w_2 \implies w_1x \equiv_L w_2x \\ \text{also} \quad & w_1 \equiv_L w_2 \implies f_x(w_1) \equiv_L f_x(w_2) \end{aligned}$$

wohldefiniert

Und weil wir gesehen hatte, dass das gilt, sind wie man auch sagt, die Abbildungen $f'_x : A^*_{/\equiv_L} \to A^*_{/\equiv_L}$ *wohldefiniert*. Die Abbildungsvorschrift ist unabhängig von der Wahl des Repräsentanten der Äquivalenzklasse, die als Argument verwendet wird.

Allgemein gilt: Wenn $\equiv$ mit $f : M \to M$ verträglich ist, dann ist $f' : M_{/\equiv} \to M_{/\equiv} : f'([x]) = [f(x)]$ wohldefiniert.

Zum Abschluss werfen wir einen letzten Blick auf die Nerode-Äquivalenzen. Sei nun L eine formale Sprache, für die $\equiv_L$ nur endlich viele Äquivalenzklassen hat. Wir schreiben zur Abkürzung $Z = A^*_{/\equiv_L}$ und definieren

$$f : Z \times A \to Z : f([w], x) = [wx]$$

Diese Abbildung ist nach dem oben Gesagten wohldefiniert. Und sie erinnert Sie hoffentlich an endliche Automaten. Das ist Absicht. Legt man nämlich noch fest

- $z_0 = [\varepsilon]$ und
- $F = \{[w] \mid w \in L\}$

dann hat man einen endlichen Akzeptor, der genau die formale Sprache L erkennt. Überlegen Sie sich das!

Ohne Beweis teilen wir Ihnen noch die folgenden schönen Tatsachen mit: Für jede formale Sprache, die von einem endlchen Akzeptor erkannt wird, hat $\equiv_L$ nur endlich viele Äquivalenzklassen. Und der gerade konstruierte Akzeptor ist unter allen, die L erkennen, einer mit minimaler Zustandszahl. Und dieser endliche Akzeptor ist bis auf Isomorphie (also Umbenenung von Zuständen) sogar eindeutig.

17.3 HALBORDNUNGEN

Eine Ihnen wohlvertraute Halbordnung ist die Mengeninklusion $\subseteq$. Entsprechende Beispiele tauchen daher im folgenden immer wieder auf, zumal Sie am Ende dieses Abschnittes sehen werden, dass man zum Beispiel durch den Fixpunktsatz von Knaster-Tarski für sogenannte vollständige Halbordnungen noch einmal einen neuen Blick auf kontextfreie Grammatiken bekommt.

17.3.1 Grundlegende Definitionen

Eine Relation $R \subseteq M \times M$ heißt *antisymmetrisch*, wenn für alle $x, y \in M$ gilt: *Antisymmetrie*

$$xRy \land yRx \Longrightarrow x = y$$

Eine Relation $R \subseteq M \times M$ heißt *Halbordnung*, wenn sie *Halbordnung*

- reflexiv,
- antisymmetrisch und
- transitiv

ist. Wenn R eine Halbordnung auf einer Menge M ist, sagt man auch, die Menge sei *halbgeordnet*. *halbgeordnete Menge*

Auf der Menge aller Wörter über einem Alphabet A ist die Relation $\sqsubseteq_p$ ein einfaches Beispiel, die definiert sei vermöge der Festlegung $w_1 \sqsubseteq_p w_2 \iff \exists u \in A^* : w_1 u = w_2$. Machen Sie sich zur Übung klar, dass sie tatsächlich die drei definierenden Eigenschaften einer Halbordnung hat.

Es sei M' eine Menge und $M = 2^{M'}$ die Potenzmenge von M'. Dann ist die Mengeninklusion $\subseteq$ eine Halbordnung auf M. Auch hier sollten Sie noch einmal aufschreiben, was die Aussagen der drei definierenden Eigenschaften einer Halbordnung sind. Sie werden merken, dass die Antisymmetrie eine Möglichkeit an die Hand gibt, die Gleichheit zweier Mengen zu beweisen (wir haben das auch schon ausgenutzt).

Wenn R Halbordnung auf einer *endlichen* Menge M ist, dann stellt man sie manchmal graphisch dar. Wir hatten schon in Unterabschnitt 11.1.4 darauf hingewiesen, dass Relationen und gerichtete Graphen sich formal nicht unterscheiden. Betrachten wir als Beispiel die halbgeordnete Menge $(2^{\{a,b,c\}}, \subseteq)$. Im zugehörigen Graphen führt eine Kante von M_1 zu M_2, wenn $M_1 \subseteq M_2$ ist. Es ergibt sich also die Darstellung aus Abbildung 17.1. Wie man sieht wird das ganze recht schnell

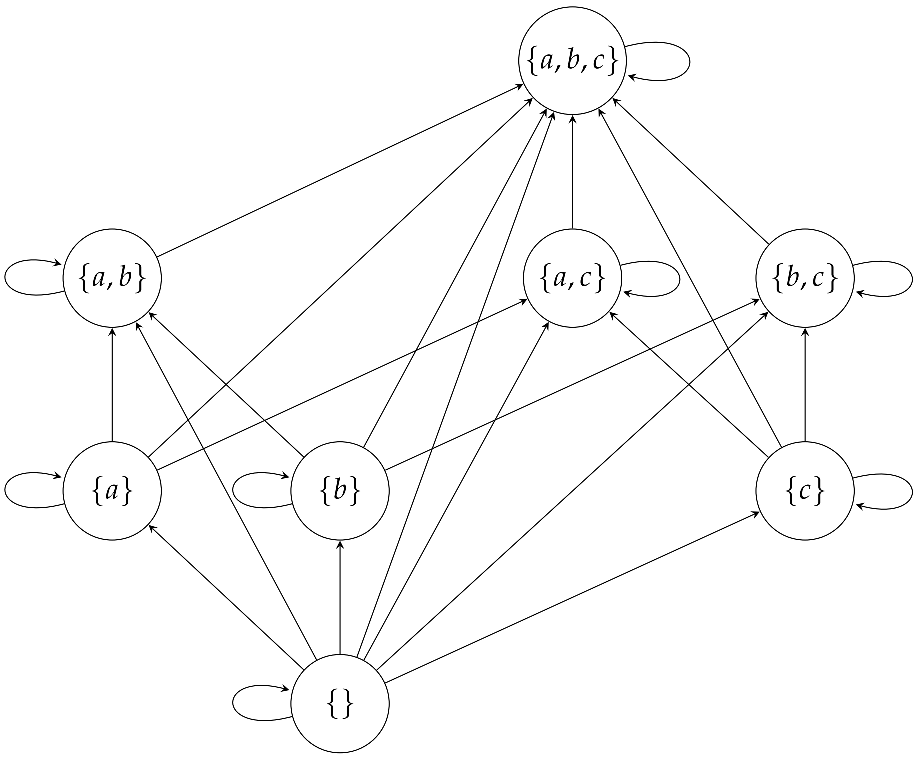


Abbildung 17.1: Die Halbordnung $(2^{\{a,b,c\}}, \subseteq)$ als Graph

relativ unübersichtlich. Dabei ist ein Teil der Kanten nicht ganz so wichtig, weil deren Existenz ohnehin klar ist (wegen der Reflexivität) oder aus anderen Kanten gefolgert werden kann (wegen der Transitivität). Deswegen wählt man meist die Darstellung als sogenanntes *Hasse-Diagramm* dar. Das ist eine Art „Skelett" der Halbordnung, bei dem die eben angesprochenen Kanten fehlen. Genauer gesagt ist es der Graph der Relation $H_R = (R \setminus \mathrm{I}) \setminus (R \setminus \mathrm{I})^2$. In unserem Beispiel ergibt sich aus Abbildung 17.1 durch Weglassen der Kanten Abbildung 17.2. *Hasse-Diagramm*

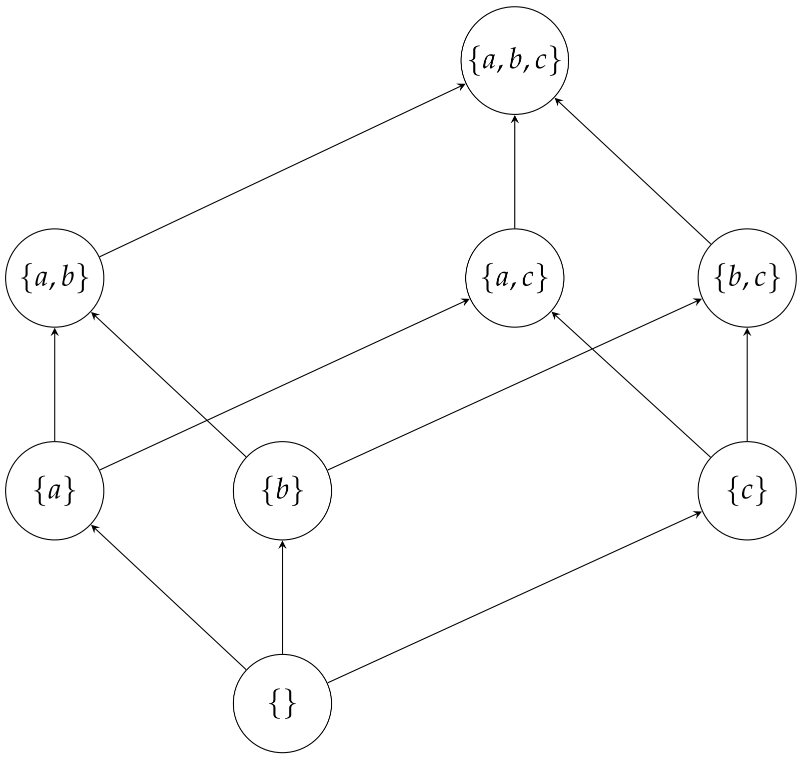


Abbildung 17.2: Hassediagramm der Halbordnung $(2^{\{a,b,c\}}, \subseteq)$

Vom Hassediagramm kommt man „ganz leicht" wieder zur ursprünglichen Halbordnung: Man muss nur die reflexiv-transitive Hülle bilden.

17.3 Lemma. Wenn R eine Halbordnung auf einer endlichen Menge M ist und H_R das zugehörige Hassediagramm, dann ist $H_R^* = R$.

17.4 Beweis. R und H_R sind beides Relationen über der gleichen Grundmenge M (also ist $R^0 = H_R^0$) und offensichtlich ist $H_R \subseteq R$. Eine ganz leichte Induktion (machen Sie sie) zeigt, dass für alle $i \in \mathbb{N}_0$ gilt: $H_R^i \subseteq R^i$ und folglich $H_R^* \subseteq R^* = R$.

Nun wollen wir zeigen, dass umgekehrt auch gilt: $R \subseteq H_R^*$. Sei dazu $(x,y) \in R$. Falls $x = y$ ist, ist auch $(x,y) \in \mathrm{I} \subseteq H_R^*$.

Sei daher im folgenden $x \neq y$, also $(x,y) \in R \setminus \mathrm{I}$, und sein $(x_0, x_1, \dots, x_m)$ eine Folge von Elementen mit folgenden Eigenschaften:

- $x_0 = x$ und $x_m = y$
- für alle $0 \leq i < m$ ist $x_i \sqsubseteq x_{i+1}$
- für alle $0 \leq i < m$ ist $x_i \neq x_{i+1}$

In einer solchen Folge kann kein Element $z \in M$ zweimal auftauchen. Wäre nämlich $x_i = z$ und $x_k = z$ mit $k > i$, dann wäre jedenfalls $k \geq i+2$ und $x_{i+1} \neq z$. Folglich wäre einerseits $z = x_i \sqsubseteq x_{i+1}$ und andererseits $x_{i+1} \sqsubseteq \cdots \sqsubseteq x_k$, also wegen Transitivität $x_{i+1} \sqsubseteq x_k = z$. Aus der Antisymmetrie von $\sqsubseteq$ würde $x_{i+1} = z$ folgen im Widerspruch zum eben Festgehaltenen.

Da in einer Folge $(x_0, x_1, \dots, x_m)$ der beschriebenen Art kein Element zweimal vorkommen kann und M endlich ist, gibt es auch maximal lange solche Folgen. Sei im folgenden $(x_0, x_1, \dots, x_m)$ maximal lang. Dann gilt also für alle $0 \leq i < m$, dass man zwischen zwei Elemente x_i und x_{i+1} kein weiteres Element einfügen kann, dass also gilt: $\neg \exists z \in M : x_i \sqsubseteq z \sqsubseteq x_{i+1} \land x_i \neq z \land x_{i+1} \neq z$.

Dafür kann man auch schreiben: $\neg \exists z \in M : (x_i, z) \in R \setminus \mathrm{I} \land (z, x_{i+1}) \in R \setminus \mathrm{I}$, d. h. $(x_i, x_{i+1}) \notin (R \setminus \mathrm{I})^2$.

Also gilt für alle $0 \leq i < m$: $(x_i, x_{i+1}) \in (R \setminus \mathrm{I}) \setminus (R \setminus \mathrm{I})^2 = H_R$. Daher ist $(x,y) = (x_0, x_m) \in H_R^m \subseteq H_R^*$. ■

gerichtete azyklische Graphen

Dag

Graphen, die das Hassediagramm einer endlichen Halbordnung sind, heißen auch *gerichtete azyklische Graphen* (im Englischen *directed acyclic graph* oder kurz *Dag*), weil sie keine Zyklen mit mindestens einer Kante enthalten. Denn andernfalls hätte man eine Schlinge oder (fast die gleiche Argumentation wie eben im Beweis 17.4) des Lemmas verschiedene Elemente x und y mit $x \sqsubseteq y$ und $y \sqsubseteq x$.

Gerichtete azyklische Graphen tauchen an vielen Stellen in der Informatik auf, nicht nru natürlich bei Problemstellungen im Zusammenhang mit Graphen, sondern z. B. auch bei der Darstellung von Datenflüssen, im Compilerbau, bei sogenannten *binary decision diagrams* zur Darstellung logischer Funktionen usw.

17.3.2 „Extreme“ Elemente

Es sei $(M, \sqsubseteq)$ eine halbgeordnete Menge und T eine beliebige Teilmenge von M.

minimales Element

maximales Element

Ein Element $x \in T$ heißt *minimales Element* von T, wenn es kein $y \in T$ gibt mit $y \sqsubseteq x$ und $y \neq x$. Ein Element $x \in T$ heißt *maximales Element* von T, wenn es kein $y \in T$ gibt mit $x \sqsubseteq y$ und $x \neq y$.

größtes Element

Ein Element $x \in T$ heißt *größtes Element* von T, wenn für alle $y \in T$ gilt: $y \sqsubseteq x$.

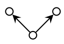
ement

Ein Element $x \in T$ heißt *kleinstes Element* von T, wenn für alle $y \in T$ gilt: $x \sqsubseteq y$.

Eine Teilmenge T kann mehrere minimale (bzw. maximale) Elemente besitzen, aber nur ein kleinstes (bzw. größtes). Als Beispiel betrachte man die Teilmenge $T \subseteq 2^{\{a,b,c\}}$ aller Teilmengen von $\{a,b,c\}$, die nichtleer sind. Diese Teilmenge besitzt die drei minimalen Elemente $\{a\}$, $\{b\}$ und $\{c\}$. Und sie besitzt ein größtes Element, nämlich $\{a,b,c\}$.

obere Schranke
untere Schranke

Ein Element $x \in M$ heißt *obere Schranke* von T, wenn für alle $y \in T$ gilt: $y \sqsubseteq x$. Ein $x \in M$ heißt *untere Schranke* von T, wenn für alle $y \in T$ gilt: $x \sqsubseteq y$. In der Halbordnung $(2^{\{a,b,c\}}, \subseteq)$ besitzt zum Beispiel $T = \{\{\}, \{a\}, \{b\}\}$ zwei obere Schranken: $\{a,b\}$ und $\{a,b,c\}$. Die Teilmenge $T = \{\{\}, \{a\}, \{b\}, \{a,b\}\}$ besitzt die gleichen oberen Schranken.

In einer Halbordnung muss nicht jede Teilmenge eine obere Schranke besitzen. Zum Beispiel besitzt die Teilmenge aller Elemente der Halbordnung mit dem Hassediagramm  *keine* obere Schranke.

Supremum
$\bigsqcup T$, $\sup(T)$

Besitzt die Menge der oberen Schranken einer Teilmenge T ein kleinstes Element, so heißt dies das *Supremum* von T und wir schreiben dafür $\bigsqcup T$ (oder $\sup(T)$).

Infimum

Besitzt die Menge der unteren Schranken einer Teilmenge T ein größtes Element, so heißt dies das *Infimum* von T. Das werden wir in dieser Vorlesung aber nicht benötigen.

Wenn eine Teilmenge kein Supremum besitzt, dann kann das daran liegen, dass sie gar keine oberen Schranken besitzt, oder daran, dass die Menge der oberen Schranken kein kleinstes Element hat. Liegt eine Halbordnung der Form $(2^M, \subseteq)$, dann besitzt aber jede Teilmenge $T \subseteq 2^M$ ein Supremum. $\bigsqcup T$ ist dann nämlich die Vereinigung aller Elemente von T (die Teilmengen von M sind).

17.3.3 Vollständige Halbordnungen

aufsteigende Kette

Eine *aufsteigende Kette* ist eine abzählbar unendliche Folge $(x_0, x_1, x_2, \dots)$ von Elementen einer Halbordnung mit der Eigenschaft: $\forall i \in \mathbb{N}_0 : x_i \sqsubseteq x_{i+1}$.

vollständige Halbordnung

Eine Halbordnung heißt *vollständig*, wenn sie ein kleinstes Element besitzt und jede aufsteigende Kette ein Supremum besitzt. Für das kleinste Element schreiben wir im folgenden $\bot$. Für das das Supremum einer aufsteigenden Kette $x_0 \sqsubseteq x_1 \sqsubseteq x_2 \sqsubseteq \cdots$ schreiben wir $\bigsqcup_i x_i$.

Ein ganz wichtiges Beispiel für eine vollständige Halbordnung ist die schon mehrfach erwähnte Potenzmenge $2^{M'}$ einer Menge M' mit Mengeninklusion $\subseteq$ als Relation. Das kleinste Element ist die leere Menge $\emptyset$. Und das Supremum einer aufsteigenden Kette $T_0 \subseteq T_1 \subseteq T_2 \subseteq \cdots$ ist $\bigsqcup_i T_i = \bigcup T_i$.

Andererseits ist $(\mathbb{N}_0, \leq)$ *keine* vollständige Halbordung, denn unbeschränkt wachsende aufsteigende Ketten wie z. B. $0 \leq 1 \leq 2 \leq \cdots$ besitzen kein Supremum in $\mathbb{N}_0$. Wenn man aber noch ein weiteres Element u „über“ allen Zahlen hinzufügt, dann ist die Ordnung vollständig. Man setzt also $N = \mathbb{N}_0 \cup \{u\}$ und definiert

$$x \sqsubseteq y \iff (x, y \in \mathbb{N}_0 \land x \leq y) \lor (y = u)$$

Weil wir es später noch brauchen können, definieren wir auch noch $N' = \mathbb{N}_0 \cup \{u_1, u_2\}$ mit der totalen Ordnung

$$x \sqsubseteq y \iff (x, y \in \mathbb{N}_0 \land x \leq y) \lor (x \in \mathbb{N}_0 \cup \{u_1\} \land y = u_1) \lor y = u_2$$

also sozusagen

$$0 \sqsubseteq 1 \sqsubseteq 2 \sqsubseteq 3 \sqsubseteq \cdots \sqsubseteq u_1 \sqsubseteq u_2$$

Ein anderes Beispiel einer (sogar totalen) Ordnung, die *nicht* vollständig ist, werden wir am Ende von Abschnitt 17.4 sehen.

17.3.4 Stetige Abbildungen auf vollständigen Halbordnungen

monotone Abbildung

Es sei $\sqsubseteq$ eine Halbordnung auf einer Menge M. Eine Abbildung $f : M \to M$ heißt *monoton*, wenn für alle $x, y \in M$ gilt: $x \sqsubseteq y \Longrightarrow f(x) \sqsubseteq f(y)$.

Zum Beispiel ist die Abbildung $f(x) = x + 1$ auf der Halbordnung $(\mathbb{N}_0, \leq)$ monoton. Die Abbildung $f(x) = x \bmod 5$ ist auf der gleichen Halbordnung dagegen nicht monoton, denn es ist zwar $3 \leq 10$, aber $f(3) = 3 \not\leq 0 = f(10)$.

stetige Abbildung

Eine Abbildung $f : D \to D$ auf einer vollständigen Halbordnung $(D, \sqsubseteq)$ heißt *stetig*, wenn für jede aufsteigende Kette $x_0 \sqsubseteq x_1 \sqsubseteq x_2 \sqsubseteq \cdots$ gilt: $f(\bigsqcup_i x_i) = \bigsqcup_i f(x_i)$.

Betrachten wir als erstes Beispiel noch einmal die vollständige Halbordnung $N' = \mathbb{N}_0 \cup \{u_1, u_2\}$ von eben. Die Abbildung $f : N' \to N'$ mit

$$f(x) = \begin{cases} x + 1 & \text{falls } x \in \mathbb{N}_0 \\ u_1 & \text{falls } x = u_1 \\ u_2 & \text{falls } x = u_2 \end{cases}$$

ist stetig. Denn für jede aufsteigende Kette $x_0 \sqsubseteq x_1 \sqsubseteq x_2 \sqsubseteq \cdots$ gibt es nur zwei Möglichkeiten:

- Die Kette wird konstant. Es gibt also ein $n' \in N'$ und ein $i \in \mathbb{N}_0$ so dass gilt: $x_0 \sqsubseteq x_1 \sqsubseteq x_2 \sqsubseteq \cdots \sqsubseteq x_i = x_{i+1} = x_{i+2} = \cdots = n'$. Dann ist jedenfalls $\bigsqcup_i x_i = n'$. Es gibt nun drei Unterfälle zu betrachten:
  - Wenn $n' = u_2$ ist, dann ist wegen $f(u_2) = u_2$ ist auch $\bigsqcup_i f(x_i) = u_2$, also ist $f(\bigsqcup_i x_i) = \bigsqcup_i f(x_i)$.

– Wenn $n' = u_1$, gilt eine analoge Überlegung.
– Wenn $n' \in \mathbb{N}_0$ ist, dann ist $f(\bigsqcup_i x_i) = f(n') = n' + 1$. Andererseits ist die Kette der Funktionswerte $f(x_0) \sqsubseteq f(x_1) \sqsubseteq f(x_2) \sqsubseteq \cdots \sqsubseteq f(x_i) = f(x_{i+1}) = f(x_{i+2}) = \cdots = f(n') = n' + 1$. Also ist $f(\bigsqcup_i x_i) = \bigsqcup_i f(x_i)$.

- Der einzige andere Fall ist: die Kette wird nicht konstant. Dann müssen alle $x_i \in \mathbb{N}_0$ sein, und die Kette wächst unbeschränkt. Das gleiche gilt dann auch für die Kette der Funktionswerte. Also haben beide als Supremum u_1 und wegen $f(u_1) = u_1$ ist $f(\bigsqcup_i x_i) = \bigsqcup_i f(x_i)$.

Der letzte Fall zeigt einem auch gleich schon, dass dagegen die folgende Funktion $g : N' \to N'$ *nicht* stetig ist:

$$g(x) = \begin{cases} x+1 & \text{falls } x \in \mathbb{N}_0 \\ u_2 & \text{falls } x = u_1 \\ u_2 & \text{falls } x = u_2 \end{cases}$$

Der einzige Unterschied zu f ist, dass nun $g(u_1) = u_2$. Eine unbeschränkt wachsende Kette $x_0 \sqsubseteq x_1 \sqsubseteq x_2 \sqsubseteq \cdots$ nichtnegativer ganzer Zahlen hat Supremem u_1, so dass $g(\bigsqcup_i x_i) = u_2$ ist. Aber die Kette der Funktionswerte $g(x_0) \sqsubseteq g(x_1) \sqsubseteq g(x_2) \sqsubseteq \cdots$ hat Supremem $\bigsqcup_i g(x_i) = u_1 \neq g(\bigsqcup_i x_i)$.

Der folgende Satz ist eine abgeschwächte Version des sogenannten Fixpunktsatzes von Knaster und Tarski.

17.5 **Satz.** *Es sei $f : D \to D$ eine monotone und stetige Abbildung auf einer vollständigen Halbordnung $(D, \sqsubseteq)$ mit kleinstem Element $\perp$. Elemente $x_i \in D$ seien wie folgt definiert:*

$$\begin{gathered} x_0 = \perp \\ \forall i \in \mathbb{N}_0 : x_{i+1} = f(x_i) \end{gathered}$$

Dann gilt:

1. *Die x_i bilden eine Kette: $x_0 \sqsubseteq x_1 \sqsubseteq x_2 \sqsubseteq \cdots$.*
2. *Das Supremum $x_f = \bigsqcup_i x_i$ dieser Kette ist Fixpunkt von f, also $f(x_f) = x_f$.*
3. *x_f ist der kleinste Fixpunkt von f: Wenn $f(y_f) = y_f$ ist, dann ist $x_f \sqsubseteq y_f$.*

17.6 **Beweis.** Mit den Bezeichnungen wie im Satz gilt:

1. Dass für alle $i \in \mathbb{N}_0$ gilt $x_i \sqsubseteq x_{i+1}$, sieht man durch vollständige Induktion: $x_0 \sqsubseteq x_1$ gilt, weil $x_0 = \perp$ das kleinste Element der Halbordnung ist. Und wenn man schon weiß, dass $x_i \sqsubseteq x_{i+1}$ ist, dann folgt wegen der Monotonie von f sofort $f(x_i) \sqsubseteq f(x_{i+1})$, also $x_{i+1} \sqsubseteq x_{i+2}$.

2. Wegen der Stetigkeit von f ist $f(x_f) = f(\bigsqcup_i x_i) = \bigsqcup_i f(x_i) = \bigsqcup_i x_{i+1}$. Die Folge der x_{i+1} unterscheidet sich von der Folge der x_i nur durch das fehlende erste Element $\perp$. Also haben „natürlich“ beide Folgen das gleiche Supremum x_f; also ist $f(x_f) = x_f$.

   Falls Sie das nicht ganz „natürlich“ fanden, hier eine ganz genaue Begründung:

   - Einerseits ist für alle $i \geq 1$: $x_i \sqsubseteq \bigsqcup_i x_{i+1}$. Außerdem ist $\perp = x_0 \sqsubseteq \bigsqcup_i x_{i+1}$. Also ist $\bigsqcup_i x_{i+1}$ eine obere Schranke für alle x_i, $i \in \mathbb{N}_0$, also ist $\bigsqcup_i x_i \sqsubseteq \bigsqcup_i x_{i+1}$.
   - Andererseits ist für alle $i \geq 1$: $x_i \sqsubseteq \bigsqcup_i x_i$. Also ist $\bigsqcup_i x_i$ eine obere Schranke für alle x_{i+1}, $i \in \mathbb{N}_0$, also ist $\bigsqcup_i x_{i+1} \sqsubseteq \bigsqcup_i x_i$.
   - Aus $\bigsqcup_i x_i \sqsubseteq \bigsqcup_i x_{i+1}$ und $\bigsqcup_i x_{i+1} \sqsubseteq \bigsqcup_i x_i$ folgt mit der Antisymmetrie von $\sqsubseteq$ sofort die Gleichheit der beiden Ausdrücke.

3. Durch Induktion sieht man zunächst einmal: $\forall i \in \mathbb{N}_0 : x_i \sqsubseteq y_f$. Denn $x_0 \sqsubseteq y_f$ gilt, weil $x_0 = \perp$ das kleinste Element der Halbordnung ist. Und wenn man schon weiß, dass $x_i \sqsubseteq y_f$ ist, dann folgt wegen der Monotonie von f sofort $f(x_i) \sqsubseteq f(y_f)$, also $x_{i+1} \sqsubseteq y_f$. Also ist y_f eine obere Schranke der Kette, also ist gilt für die kleinste obere Schranke: $x_f = \bigsqcup_i x_i \sqsubseteq y_f$.

■

Dieser Fixpunktsatz (und ähnliche) finden in der Informatik an mehreren Stellen Anwendung. Zum Beispiel kann er Ihnen in Vorlesungen über Sematik von Programmiersprachen wieder begegnen.

Hier können wir Ihnen schon andeuten, wie er im Zusammenhang mit kontextfreien Grammatiken nützlich sein kann. Betrachten wir als Terminalzeichenalphabet $T = \{\mathtt{a}, \mathtt{b}\}$ und die kontextfreie Grammatik $G = (\{X\}, T, X, P)$ mit Produktionenmenge $P = \{X \to \mathtt{a}X\mathtt{b} \mid \varepsilon\}$. Als halbgeordnete Menge D verwenden wir die Potenzmenge $D = 2^{T^*}$ der Menge aller Wörter mit Inklusion als Halbordnungsrelation. Die Elemente der Halbordnung sind also Mengen von Wörtern, d. h. formale Sprachen. Kleinstes Element der Halbordnung ist die leere Menge $\emptyset$. Wie erwähnt, ist diese Halbordnung vollständig.

Es sei nun $f : D \to D$ die Abbildung mit $f(L) = \{\mathtt{a}\}L\{\mathtt{b}\} \cup \{\varepsilon\}$. Der Bequemlichkeit halber wollen wir nun einfach glauben, dass f stetig ist. (Wenn Ihnen das nicht passt, prüfen Sie es nach. Es ist nicht schwer.) Der Fixpunktsatz besagt, dass man den kleinsten Fixpunkt dieser Abbildung erhält als Supremum, hier also

Vereinigung, aller der folgenden Mengen:

$$\begin{aligned}
L_0 &= \emptyset \\
L_1 &= f(L_0) = \{\mathtt{a}\}L_0\{\mathtt{b}\} \cup \{\varepsilon\} \\
&= \{\varepsilon\} \\
L_2 &= f(L_1) = \{\mathtt{a}\}L_1\{\mathtt{b}\} \cup \{\varepsilon\} \\
&= \{\mathtt{ab}, \varepsilon\} \\
L_3 &= f(L_2) = \{\mathtt{a}\}L_2\{\mathtt{b}\} \cup \{\varepsilon\} \\
&= \{\mathtt{aabb}, \mathtt{ab}, \varepsilon\}
\end{aligned}$$

Sie sehen, wie der Hase läuft. Der kleinste Fixpunkt ist $L = \{\mathtt{a}^k\mathtt{b}^k \mid k \in \mathbb{N}_0\}$. Das ist auch genau die Sprache, die die Grammatik erzeugt. Und L ist Fixpunkt von f, also

$$L = \{\mathtt{a}\}L\{\mathtt{b}\} \cup \{\varepsilon\}$$

Es ist also sozusagen die kleinste Lösung der Gleichung $X = \{\mathtt{a}\}X\{\mathtt{b}\} \cup \{\varepsilon\}$. Was das mit den Produktionen der Grammatik zu tun, sehen Sie vermutlich.

17.4 ORDNUNGEN

Eine Relation $R \subseteq M \times M$ ist eine *Ordnung*, oder auch genauer *totale Ordnung*, wenn R Halbordnung ist und außerdem gilt:

Ordnung
totale Ordnung

$$\forall x, y \in M : xRy \vee yRx$$

Wie kann man aus der weiter vorne definierten Halbordnung $\sqsubseteq_p$ auf A^* eine totale Ordnung machen? Dafür gibt es natürlich verschiedene Möglichkeiten. Auf jeden Fall muss aber z. B. festgelegt werden, ob $\mathtt{a} \sqsubseteq \mathtt{b}$ oder $\mathtt{b} \sqsubseteq \mathtt{a}$.

Es ist also auf jeden Fall eine totale Ordnung $\sqsubseteq_A$ auf den Symbolen des Alphabetes erforderlich. Nehmen wir an, wir haben das: also z. B. $\mathtt{a} \sqsubseteq_A \mathtt{b}$.

Dann betrachtet man des öfteren zwei sogenannte *lexikographische Ordnungen*. Die eine ist die naheliegende Verallgemeinerung dessen, was man aus Wörterbüchern kennt. Die andere ist für algorithmische Zwecke besser geeignet.

lexikographische Ordnung

- Die lexikographische Ordnung $\sqsubseteq_1$, nach der Wörter im Lexika usw. sortiert sind, kann man wie folgt definieren. Seien $w_1, w_2 \in A^*$. Dann gibt es das eindeutig bestimmte maximal lange gemeinsame Präfix von w_1 und w_2, also das maximal lange Wort $v \in A^*$, so dass es $u_1, u_2 \in A^*$ gibt mit $w_1 = v\,u_1$ und $w_2 = v\,u_2$. Drei Fälle sind möglich:

1. Falls $v = w_1$ ist, gilt $w_1 \sqsubseteq_1 w_2$.
2. Falls $v = w_2$ ist, gilt $w_2 \sqsubseteq_1 w_1$.
3. Falls $w_1 \neq v \neq w_2$, gibt es $x, y \in A$ und $u_1', u_2' \in A^*$ mit
   - $x \neq y$ und
   - $w_1 = v\,x\,u_1'$ und $w_2 = v\,y\,u_2'$.

   Dann gilt $w_1 \sqsubseteq_1 w_2 \iff x \sqsubseteq_A y$.

Muss man wie bei einem Wörterbuch nur endlich viele Wörter ordnen, dann ergibt sich zum Beispiel

$$\begin{aligned} \mathtt{a} \sqsubseteq_1 \mathtt{aa} \sqsubseteq_1 \mathtt{aaa} \sqsubseteq_1 \mathtt{aaaa} \\ \sqsubseteq_1 \mathtt{ab} \sqsubseteq_1 \mathtt{aba} \sqsubseteq_1 \mathtt{abbb} \\ \sqsubseteq_1 \mathtt{b} \sqsubseteq_1 \mathtt{baaaaaa} \sqsubseteq_1 \mathtt{baab} \\ \sqsubseteq_1 \mathtt{bbbbb} \end{aligned}$$

Allgemein auf der Menge aller Wörter ist diese Ordnung aber nicht ganz so „harmlos“. Wir gehen gleich noch darauf ein.

- Eine andere lexikographische Ordnung $\sqsubseteq_2$ auf A^* kann man definieren vermöge der Festlegungen: $w_1 \sqsubseteq_2 w_2$ gilt genau dann, wenn
  - entweder $|w_1| < |w_2|$
  - oder $|w_1| = |w_2|$ und $w_1 \sqsubseteq_1 w_2$ gilt.

  Diese Ordnung beginnt also z. B. im Falle $A = \{\mathtt{a}, \mathtt{b}\}$ mit der naheliegenden Ordnung $\sqsubseteq_A$ so:

$$\begin{aligned} &\varepsilon \sqsubseteq_2 \mathtt{a} \sqsubseteq_2 \mathtt{b} \\ &\sqsubseteq_2 \mathtt{aa} \sqsubseteq_2 \mathtt{ab} \sqsubseteq_2 \mathtt{ba} \sqsubseteq_2 \mathtt{bb} \\ &\sqsubseteq_2 \mathtt{aaa} \sqsubseteq_2 \cdots \sqsubseteq_2 \mathtt{bbb} \\ &\sqsubseteq_2 \mathtt{aaaa} \quad \sqsubseteq_2 \cdots \sqsubseteq_2 \mathtt{bbbb} \\ &\cdots \end{aligned}$$

Wir wollen noch darauf hinweisen, dass die lexikographische Ordnung $\sqsubseteq_1$ als Relation auf der Menge aller Wörter einige Eigenschaften hat, an die man als Anfänger vermutlich nicht gewöhnt ist. Zunächst einmal merkt man, dass die Ordnung nicht vollständig ist. Die aufsteigende Kette

$$\varepsilon \sqsubseteq_1 \mathtt{a} \sqsubseteq_1 \mathtt{aa} \sqsubseteq_1 \mathtt{aaa} \sqsubseteq_1 \mathtt{aaaa} \sqsubseteq_1 \cdots$$

besitzt kein Supremum. Zwar ist jedes Wort, das mindestens ein $\mathtt{b}$ enthält, obere Schranke, aber es gibt keine kleinste. Das merkt man, wenn man die absteigende Kette

$$\mathtt{b} \sqsupseteq_1 \mathtt{ab} \sqsupseteq_1 \mathtt{aab} \sqsupseteq_1 \mathtt{aaab} \sqsupseteq_1 \mathtt{aaaab} \sqsupseteq_1 \cdots$$

betrachtet. Jede obere Schranke der aufsteigenden Kette muss ein b enthalten. Aber gleich, welche obere Schranke w man betrachtet, das Wort $\mathtt{a}^{|w|}\mathtt{b}$ ist eine echt kleine obere Schranke. Also gibt es keine kleinste.

Dass es sich bei obigen Relationen überhaupt um totale Ordnungen handelt, ist auch unterschiedlich schwer zu sehen. Als erstes sollte man sich klar machen, dass $\sqsubseteq_1$ auf der Menge A^n aller Wörter einer festen Länge n eine totale Ordnung ist. Das liegt daran, dass für verschiedene Wörter gleicher Länge niemals Punkt 1 oder Punkt 2 zutrifft. Und da $\sqsubseteq_A$ als totale Ordnung vorausgesetzt wird, ist in Punkt 3 stets $x \sqsubseteq_A y$ oder $y \sqsubseteq_A x$ und folglich $w_1 \sqsubseteq_1 w_2$ oder $w_2 \sqsubseteq_1 w_1$.

Daraus folgt schon einmal das auch $\sqsubseteq_2$ auf der Menge A^n aller Wörter einer festen Länge n eine totale Ordnung ist, und damit überhaupt eine totale Ordnung.

Für $\sqsubseteq_1$ muss man dafür noch einmal genauer Wörter unterschiedlicher Länge in Betracht ziehen. Wie bei der Formulierung der Definition schon suggeriert, decken die drei Punkte alle Möglichkeiten ab.

17.5 AUSBLICK

Vollständige Halbordnungen spielen zum Beispiel eine wichtige Rolle, wenn man sich mit sogenannter denotationaler Semantik von Programmiersprachen beschäftigt und die Bedeutung von while-Schleifen und Programmen mit rekursiven Funktionsaufrufen präzisieren will. Den erwähnten Fixpunktsatz (oder verwandte Ergebnisse) kann man auch zum Beispiel bei der automatischen statischen Datenflussanalyse von Programmen ausnutzen. Diese und andere Anwendungen werden ihnen in weiteren Vorlesungen begegnen.