Vorkurs Mathematik 2007

Vorlesung 4

Tilman Bauer

Universität Münster

13. September 2007

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Kartesische Produkte

Wir befassen uns in dieser Vorlesung noch einmal mit Mengen.

Definition

Seien M und N zwei Mengen. Dann bezeichnen wir mit $M \times N$ das (kartesische) Produkt dieser Mengen. Die Elemente von $M \times N$ sind Paare (m, n), wobei $m \in M$ und $n \in N$ liegt.

- Anders als bei Mengen ist bei Paaren die Reihenfolge wichtig: $(m, n) \neq (n, m)$.
- Ebenso kann man das Produkt von drei Mengen $M \times N \times K$ bilden, dessen Elemente die Tripel (m, n, k) sind.
- Ebenso die Menge der Quadrupel, Quintupel usw. Allgemein sagt man auch n-Tupel für $n \in \mathbb{N}$.

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Kartesische Produkte

- Das Wort kartesisch leitet sich von Cartesius ab, der lateinischen Version des Nachnamens des Philosophen und Mathematikers René DESCARTES (1596–1650).
- Descartes hat das kartesische Koordinatensystem erfunden (x-, y- und ggf. z-Koordinaten), das man als kartesisches Produkt $\mathbb{R} \times \mathbb{R}(\times\mathbb{R})$ auffassen kann.
- Haben M und N nur endlich viele Elemente (sagen wir m und n), so hat $M \times N$ genau $m \cdot n$ Elemente:
- Ist $M = \{\mu_1, \ldots, \mu_m\}$, $N = \{\nu_1, \ldots, \nu_n\}$, so ist

$$M \times N = \left\{ \begin{matrix} (\mu_1, \nu_1), & (\mu_1, \nu_2), & \ldots, & (\mu_1, \nu_n) \\ (\mu_2, \nu_1), & (\mu_2, \nu_2), & \ldots, & (\mu_2, \nu_n) \\ \vdots & \vdots & \ddots & \vdots \\ (\mu_m, \nu_1), & (\mu_m, \nu_2), & \ldots, & (\mu_m, \nu_n) \end{matrix} \right\}.$$

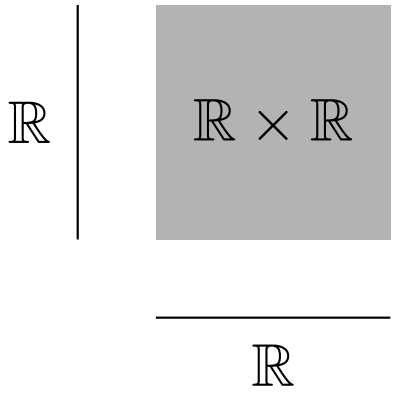
Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Kartesische Produkte

Beispiel

- Die Punktmenge einer Ebene lässt sich mit $\mathbb{R} \times \mathbb{R}$ identifizieren: Einen Punkt kann man eindeutig beschreiben durch $(x, y) \in \mathbb{R} \times \mathbb{R}$, wobei x und y Koordinaten sind.



Dazu muss man natürlich ein Koordinatensystem gewählt haben.

- Die Punktmenge eines Rechtecks mit den Kantenlängen a und b lässt sich als $[0, a] \times [0, b]$ beschreiben: wiederum gibt man zwei Koordinaten x und y an, wobei aber $0 \leq x \leq a$ und $0 \leq y \leq b$ gelten muss.

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Relationen

Definition

Eine Relation auf einer Menge M ist eine Teilmenge R von $M \times M$.
Ist $(m, m') \in R$, so schreiben wir: mRm'.

- Eine Relation bestimmt zu je zwei Elementen m und m', ob sie zueinander in einer gewissen Beziehung stehen (mRm') oder nicht.
- Eine sehr stupide, aber gültige Relation ist die leere Menge $\emptyset \subseteq M \times M$: hier gilt mRm' nie.
- Am anderen Ende der Skala ist $R = M \times M$: hier gilt mRm' immer.

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

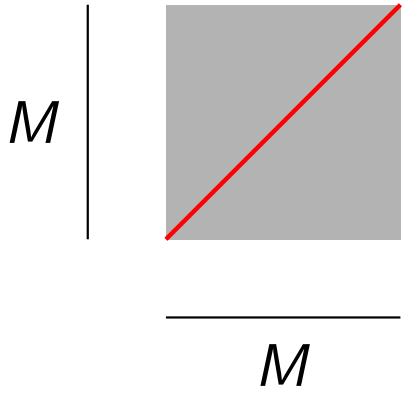
Mengen und Relationen

Relationen

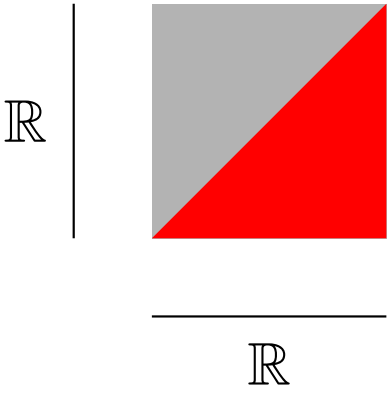
Beispiel

Weitere Beispiele von Relationen:

- die Gleichheitsrelation $R = \{(m, m) \mid m \in M\}$. Hier gilt mRm' genau dann, wenn $m = m'$ ist.



- die Relation „$\leq$“ auf $\mathbb{R} \times \mathbb{R}$.

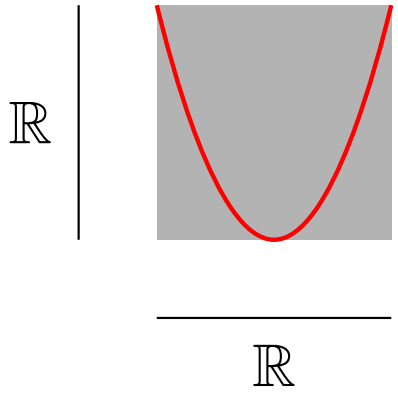


Mengen und Relationen

Abbildungen als Relationen

Abbildungen $f\colon M \to M$ lassen sich als Spezialfall von Relationen verstehen:

- Zu f definieren wir eine Relation R_f mit $xR_fy \iff f(x) = y$.
- Wir nennen R_f auch den Graph von f.
- Zum Beispiel: Ist $f\colon \mathbb{R} \to \mathbb{R}$ gegeben durch $f(x) = x^2$, so ist R_f in folgendem Bild dargestellt:



- Nicht jede Relation R kommt von einer Funktion. Sie muss erfüllen:

Zu jedem $x \in M$ existiert genau ein $y \in M$ mit xRy.

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Eigenschaften von Relationen

Zu Relationen gibt es wiederum einige Vokabeln:
$R \subseteq M \times M$ heißt

symmetrisch, falls $xRy \iff yRx$

reflexiv, falls xRx für alle x

antisymmetrisch, falls $(xRy \wedge yRx) \Rightarrow x = y$

total, falls $xRy \vee yRx$ für alle x, y

transitiv, falls $(xRy \wedge yRz) \Rightarrow xRz$.

Beispiel (auf $\mathbb{R} \times \mathbb{R}$:)

Relation	symm.	refl.	antis.	total	trans.
$=$	✓	✓	✓	–	✓
$\neq$	✓	–	–	–	–
$\leq$	–	✓	✓	✓	✓
$<$	–	–	✓(!)	–	✓

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Äquivalenzrelationen

Definition

Eine Relation R heißt Äquivalenzrelation, falls sie symmetrisch, reflexiv und transitiv ist.

Äquivalenzrelationen bezeichnet man meist mit Symbolen wie $\sim$, $\simeq$, $\equiv$, $\cong$, $\approx$ etc.

Also muss für eine Äquivalenzrelation $\sim$ gelten:

$$\begin{aligned} \text{(SYM)} \quad & x \sim y \iff y \sim x \\ \text{(REF)} \quad & x \sim x \\ \text{(TRANS)} \quad & x \sim y \wedge y \sim z \Rightarrow x \sim z. \end{aligned}$$

Beispiel

- „$=$“ ist eine Äquivalenzrelation.
- „ist gleich alt wie“ ist eine Äquivalenzrelation auf der Menge der Menschen.

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Äquivalenzrelationen

Beispiel (Fortsetzung)

$M \cong N \iff$ „es gibt eine bijektive Abbildung zwischen den Mengen M und N“ ist eine Äquivalenzrelation. Denn:

(SYM) $M \cong N \iff N \cong M$: Gibt es eine Bijektion $M \to N$, so ist die Umkehrabbildung $N \to M$ auch eine Bijektion.

(REF) $M \cong M$: Die Identitätsabbildung $f(m) = m$ ist eine Bijektion $M \to M$.

(TRANS) $M \cong N \wedge N \cong K \Rightarrow M \cong K$: Sind $f\colon M \to N$ und $g\colon N \to K$ Bijektionen, so auch $g \circ f\colon M \to K$. Hierbei ist $(g \circ f)(m)$ definiert als $g(f(m))$.

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Äquivalenzrelationen

Beispiel (noch eines)

Sei $n > 0$ eine natürliche Zahl. Auf der Menge $\mathbb{Z}$ der ganzen Zahlen sei $x \equiv y \iff n$ teilt $x - y$.

(SYM) $x \equiv y \iff y \equiv x$: Gilt $n \mid (x - y)$, so auch $n \mid (y - x)$.
Ist $n \cdot k = x - y$, so ist $n \cdot (-k) = y - x$.

(REF) $x \equiv x$: Einfach. Jede Zahl n teilt $x - x = 0$.

(TRANS) $x \equiv y \wedge y \equiv z \Rightarrow x \cong z$: ist $n \cdot k = x - y$ und $n \cdot l = y - z$, so ist
$n \cdot (k + l) = (x - y) + (y - z) = x - z$.

Um die Abhängigkeit von n anzudeuten, schreibt man gern

$$x \equiv y \pmod{n}$$

(„x ist kongruent zu y modulo n“)

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Äquivalenzklassen

Sei $\sim$ eine Äquivalenzrelation auf einer Menge M.

Definition

Eine Äquivalenzklasse von $\sim$ ist eine nicht leere Teilmenge K von M, die folgende zwei Bedingungen erfüllt:

- Je zwei Elemente $x, y \in K$ sind äquivalent, d.h. $x \sim y$.
- Ist $x \in K$ und $y \sim x$, so ist auch $y \in K$.

Beispiel

- Die Menge $\{\ldots, -8, -4, 0, 4, 8, \ldots\}$ ist eine Äquivalenzklasse für $\equiv \pmod 4$.
- Die Menge $\{\ldots, -9, -5, -1, 3, 7, \ldots\}$ ist ebenfalls eine Äquivalenzklasse!
- Insgesamt hat $\equiv \pmod 4$ genau vier Äquivalenzklassen.

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Äquivalenzklassen

Sei $\sim$ eine Äquivalenzrelation auf einer Menge M.

Definition

Eine Äquivalenzklasse von $\sim$ ist eine nicht leere Teilmenge K von M, so dass

- Je zwei Elemente $x, y \in K$ sind äquivalent, d.h. $x \sim y$.
- Ist $x \in K$ und $y \sim x$, so ist auch $y \in K$.

Lemma

1. *Je zwei verschiedene Äquivalenzklassen sind disjunkt.*
2. *Die Vereinigung aller Äquivalenzklassen ist ganz M.*
3. *Jedes Element $x \in M$ gehört zu genau einer Äquivalenzklasse.*

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Äquivalenzklassen

1. Je zwei verschiedene Äquivalenzklassen sind disjunkt.

Beweis.

- Seien K und L zwei Äquivalenzklassen von $\sim$.
- Angenommen, K und L wären nicht disjunkt, es gäbe also ein Element $x \in K \cap L$.
- Wir zeigen: $K \subseteq L$. Sei also $y \in K$.
- Also gilt $x \sim y$ (und $y \sim x$).
- Da außerdem $x \in L$, muss auch $y \in L$ sein.
- Umgekehrt sieht man genauso: $L \subseteq K$.
- Also ist $K = L$.

$\square$

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Äquivalenzklassen

3. Jedes Element $x \in M$ gehört zu genau einer Äquivalenzklasse.

Beweis.

- „höchstens eine“ ist schon gezeigt: wäre x in zwei verschiedenen Klassen, so wären diese nicht disjunkt!
- Behauptung: $K = \{y \in M \mid y \sim x\}$ ist eine Äquivalenzklasse.
- Sind y, $y' \in K$, so gilt $y \sim x$ und $y' \sim x$.
- Transitivität $\Rightarrow y \sim y'$
- Ist umgekehrt $y \in K$ und $y' \sim y$, so folgt $y' \sim y \sim x$, also $y' \sim x$, also $y' \in K$.
- K enthält x, denn $x \sim x$.

$\square$

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Äquivalenzklassen

2. Die Vereinigung aller Äquivalenzklassen ist ganz M.

Beweis.

Da jedes Element $x \in M$ zu einer Äquivalenzklasse gehört, gehört x automatisch zur Vereinigung aller Äquivalenzklassen. □

Definition

Wir bezeichnen die Äquivalenzklasse K, die x enthält, mit $[x]$.

Wir bezeichnen die Menge aller Äquivalenzklassen einer Relation $\sim$ auf M mit $M/\sim$ („M modulo $\sim$“).

- $M/\sim$ ist eine Menge von Mengen!
- Ist $x \sim y$, so gilt $[x] = [y]$, obwohl $x \neq y$ gelten kann.

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Äquivalenzklassen

Beispiel

Ist $\equiv$ die Äquivalenzrelation „Kongruenz modulo 4“ auf $\mathbb{Z}$, so ist

$$(\mathbb{Z}/\equiv) = \left\{ \begin{array}{c} \{\ldots,-8,-4,0,4,8,\ldots\}, \\ \{\ldots,-7,-3,1,5,9,\ldots\}, \\ \{\ldots,-6,-2,2,6,10,\ldots\}, \\ \{\ldots,-5,-1,3,7,11,\ldots\} \end{array} \right\}$$

$$= \{[0],[1],[2],[3]\} = \{[4],[-3],[2],[11]\} = \ldots$$

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Mengen und Relationen

Äquivalenzklassen

Beispiel

- Betrachten wir die Äquivalenzrelation $\sim$: „ist im gleichen Lebensjahr wie“ auf der Menge M aller lebenden Menschen.
- Was ist $M/\sim$? Eine Äquivalenzklasse ist z.B. die Menge aller Menschen, die zur Zeit 19 Jahre alt sind.
- Wir könnten diese Äquivalenzklasse mit $[\mathit{Astrid}]$ bezeichnen.
- Oder z.B. schlicht mit $\underline{19}$.
- Derzeit sähe $M/\sim$ so aus:

$$(M/\sim) = \{\underline{0}, \underline{1}, \ldots, \underline{113}, \underline{114}\},$$

- wobei $\underline{114} = [\text{Edna Parker}]$.

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Modulo-Arithmetik

Rechenregeln für $\mathbb{Z}/n$

Wir wollen uns eingehender mit den Äquivalenzklassen zur Kongruenz modulo n befassen.
Statt $\mathbb{Z}/\equiv$ schreiben wir $\mathbb{Z}/n$ für die Menge aller Äquivalenzklassen:

$$\mathbb{Z}/n = \{[0], [1], \ldots, [n-1]\}.$$

Definition

Wir erklären eine Addition auf $\mathbb{Z}/n$ durch $[x] + [y] = [x + y]$.

- Zwei Klassen werden addiert, indem wir Repräsentanten wählen (x und y), diese addieren ($x + y$) und die dazugehörige Äquivalenzklasse betrachten.
- Wohldefiniertheit: Wir müssen zeigen, dass es egal ist, welche Repräsentanten x und y wir wählen!

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Modulo-Arithmetik

Rechenregeln für $\mathbb{Z}/n$

Definition

Wir erklären eine Addition auf $\mathbb{Z}/n$ durch $[x] + [y] = [x + y]$.

Wohldefiniertheit: Seien x' und y' andere Repräsentanten, also $[x'] = [x]$ und $[y'] = [y]$.

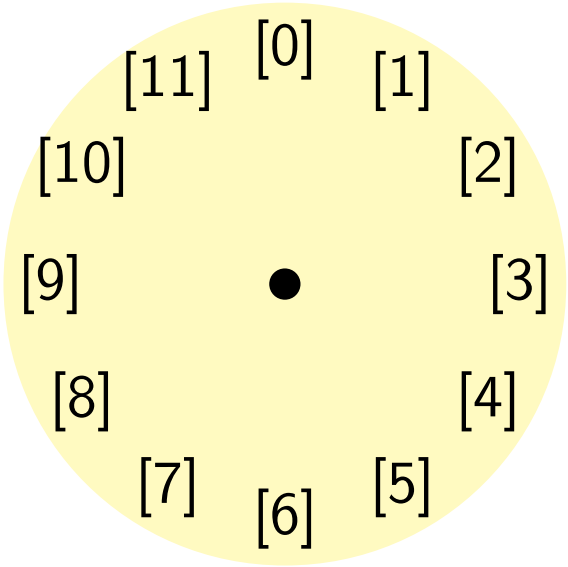
- Wir müssen zeigen, dass $[x' + y'] = [x + y]$ gilt.
- Da $[x'] = [x]$ gilt, muss $n \mid (x - x')$ gelten.
- Ebenso $n \mid (y - y')$.
- Also auch $n \mid (x - x') + (y - y') = (x + y) - (x' + y')$.
- Also gilt $[x' + y'] = [x + y]$. $\square$

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Modulo-Arithmetik

Rechenregeln für $\mathbb{Z}/n$

Wir können uns diese Arithmetik wie das Rechnen mit Stunden auf einer Uhr vorstellen:



$$[2] + [6] = [8]$$
$$[9] + [4] = [1] \quad (= [13])$$
$$[11] + [11] = [10] \quad (= [22])$$

Definition

Genauso erklären wir eine Negation $-[x] = [-x]$ und eine Multiplikation $[x] \cdot [y] = [x \cdot y]$.

Wohldefiniertheit: Übungsaufgabe!

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Modulo-Arithmetik

Inverse

Beispiel

In $\mathbb{Z}/12$ gilt:

- $[2] \cdot [-1] = [-2] = [10]$
- $[10] \cdot [4] = [40] = [4]$

Frage: Können wir in $\mathbb{Z}/n$ auch dividieren?

- Es ergibt keinen Sinn, $[\frac{1}{3}]$ hinzuschreiben, weil $\frac{1}{3} \notin \mathbb{Z}$.
- Da wir multiplizieren können, müssen wir uns fragen, ob es Kehrwerte $\frac{1}{x}$ gibt.
- Das heißt: $\forall x \in \mathbb{Z}\ \exists y \in \mathbb{Z}\colon [x] \cdot [y] = [1]$?
- Zum Beispiel in $\mathbb{Z}/12$: $[5] \cdot [5] = [25] = [1]$. Also $\frac{1}{[5]} = [5]$.
- Andererseits gibt es kein y, so dass $[4] \cdot [y] = [1]$, denn $4 \cdot y$ ist immer gerade!

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Modulo-Arithmetik

Inverse

Satz

In $\mathbb{Z}/n$ gibt es genau zu den Äquivalenzklasse $[m]$ einen Kehrwert, für die m und n teilerfremd sind.

Beweis.

Wir verwenden:

Lemma

m und n sind genau dann teilerfremde Zahlen, wenn es ganze Zahlen k und l gibt, so dass $km + ln = 1$.

Also: m und n sind teilerfremd $\iff \exists k,\ l\colon 1 = km + ln$

$$\begin{aligned} &\iff \exists k,\ l\colon [1] = [km] + [ln] = [k][m] + [l][n] \\ &\qquad\qquad = [k][m] + [l][0] = [k][m] \\ &\iff m \text{ hat einen Kehrwert.} \end{aligned}$$

$\square$

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Modulo-Arithmetik

Der euklidische Algorithmus

Das Lemma lässt sich allgemeiner formulieren. Dazu benötigen wir den Begriff des größten gemeinsamen Teilers (ggT).

Definition

Seien $m,\ n \in \mathbb{Z}$ nicht beide Null. Dann nennt man $g \in \mathbb{N}$ den ggT von m und n, falls

- $g \mid m$ und $g \mid n$
- Ist l ein anderer Teiler von sowohl m als auch n, dann gilt $l \mid g$.

Lemma

Sei $g = \mathrm{ggT}(m, n)$. Dann gibt es ganze Zahlen k und l, so dass $km + ln = g$.

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Modulo-Arithmetik

Der euklidische Algorithmus

Lemma

Sei $g = \mathrm{ggT}(m, n)$. Dann gibt es ganze Zahlen k und l, so dass $km + ln = g$.

Beweis durch Induktion in n; o.B.d.A. $n \geq m$.

- Induktionsanfang: Ist $n = 1$, so ist $g = 1$. Somit ist $0 \cdot m + 1 \cdot n = 1$.
- Induktionsschritt: Das Lemma gelte für alle $n' < n$.
- Ist $m = 0$, so ist $g = n = 0 \cdot m + 1 \cdot n$.
- Ansonsten können wir n mit Rest durch m teilen: $n = am + b$ mit $0 \leq b < m$.
- $\mathrm{ggT}(m, n) = \mathrm{ggT}(b, m)$, denn jede Zahl, die m und n teilt, teilt auch $b = n - am$.
- Induktionsannahme $\Rightarrow g = \mathrm{ggT}(m, n) = \mathrm{ggT}(b, m) = km + lb = km + l(n - am) = (k - la)m + ln$. $\square$

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben

Aufgaben

Bitte bis Dienstag, den 18. 9., bearbeiten!

1. Welche der Eigenschaften symmetrisch, reflexiv, antisymmetrisch, total, transitiv besitzen die folgenden Relationen auf $\mathbb{R}$? Begründen Sie Ihre Antworten.
   - $x \sim y \iff |x| = |y|$ ($|x|$ ist der Betrag von x, also x ohne Vorzeichen)
   - $x \sim y \iff x - y \geq 1$
2. Welche Klassen von $\mathbb{Z}/13$ besitzen Kehrwerte? Berechnen Sie alle.

Mengen und Relationen
Produkte
Relationen
Graphen
Äquivalenzen
Kongruenz
Äquivalenzklassen
Modulo-Arithmetik
Rechenregeln
Inverse
Der euklidische Algorithmus
Aufgaben