

# Diskrete Mathematik 2

Jan Fässler

2. Semester (FS 2012)

# 1 Quantifizierung

## 1.1 Einleitung

Die bekannten mathematischen Quantifizierungen haben den Nachteil, dass sie nicht immer 100% genau sind. Deshalb definieren wir eine neue, genauere Schreibweise:

$$\sum_{i=1}^n i^2 = (+i : \mathbb{Z} | 1 \leq i \leq n : i^2)$$

## 1.2 Definition

$(\oplus v_1 : T_1, \dots, v_n : T_n | R : P)$

- Anwendung von  $\oplus$  auf die Werte von P für alle Kombinationen von Werten für  $v_1, \dots, v_n$  für die R wahr ist.
- Ist R für keine Kombination wahr, dann die Identität u von  $\oplus$
- Datentyp der gesamten Quantifizierung: T

$\oplus$  - binärer Operator

- Typ:  $T \times T \rightarrow T$
- Eigenschaften:
  - $a \oplus b = b \oplus a$  (kommutativ)
  - $a \oplus (b \oplus c) = (a \oplus b) \oplus c$  (assoziativ)
  - $a \oplus u = a$

$T_1, \dots, T_n$  - Datentypen

$v_1, \dots, v_n$  - gebundene Variablen

- Englisch: bound variables, dummies
- $n \geq 1$
- alle paarweise verschieden
- $v_i$  hat den Typ  $T_i$

$R$  - boolescher Ausdruck

- ist der Bereich (Range)
- kann  $v_1, \dots, v_n$  enthalten

$P$  - beliebiger Ausdruck von Typ T

- ist der Körper (Body)
- kann  $v_1, \dots, v_n$  enthalten

	Quantor	Operator	Idendität	Typ
	$\sum$	+	0	$\mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$
•	$\prod$	*	1	$\mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$
	$\forall$	$\wedge$	<i>true</i>	$\mathbb{B} \times \mathbb{B} \rightarrow \mathbb{B}$
	$\exists$	$\vee$	<i>false</i>	$\mathbb{B} \times \mathbb{B} \rightarrow \mathbb{B}$

### 1.3 Beispiele

- $(+i | 0 \leq i < 4 : i * 8) = 0 * 8 + 1 * 8 + 2 * 8$
- $(*i | 0 \leq i < 3 : i + (i + 1)) = (0 + 0 + 1) * (1 + 1 + 1) * (2 + 2 + 1)$
- $(\wedge i | 0 \leq i < 2 : i * d \neq 6) = 0 * d \neq 6 \wedge 1 * d \neq 6 = d \neq 6$
- $(\vee i | \leq i < 21 : b[i] = 0) = b[0] = 0 \vee b[1] = 0 \vee \dots \vee b[20] = 0$
- $(+k : \mathbb{N} | k^2 = 4 : k^2) = 2^2$
- $(+k : \mathbb{Z} | k^2 = 4 : k) = 2^2 + (-2)^2$

### 1.4 Variablen

#### 1.4.1 Definition

**Def.** Eine Variable  $v$  heisst frei in einem Ausdruck  $E$ , falls  $v$  in  $E$  frei auftritt.

$FV(E)$  = Menge der freien Variablen in  $E$

**Bem.** Werte der freien Variablen stehen nicht im Ausdruck selbst, diese Info muss aus anderer Quelle kommen. (Background Info)

**Def.** Ein Ausdruck  $E$  ohne freie Variable heisst geschlossen.

**Bem.** Die erste gebundene Variable nennt man bindend und alle folgenden angewandt.

#### 1.4.2 Beispiele

$$E_1 : (\underbrace{\sum \underbrace{i : \mathbb{Z}}_{\text{gebunden}} | 0 \leq \underbrace{i}_{\text{frei}} < \underbrace{n}_{\text{frei}} : \underbrace{i^2}_{\text{frei}}}_{\text{gebunden}})$$

$$E_2 : (\overbrace{(\sum \underbrace{i : \mathbb{Z}}_{\text{gebunden}} | 0 \leq \underbrace{i}_{\text{frei}} < \underbrace{n}_{\text{frei}} : \underbrace{i^2 + i}_{\text{frei}})}^{n \text{ muss gleich sein (frei)}} + (\sum \underbrace{i : \mathbb{Z}}_{\text{gebunden}} | 0 \leq \underbrace{i}_{\text{frei}} < \underbrace{n}_{\text{frei}} : \underbrace{i^3}_{\text{frei}})_{\text{gebunden}})$$

$$E_3 : (\overbrace{(\pi \underbrace{u}_{\text{frei}} | \underbrace{k}_{\text{frei}} \leq \underbrace{u}_{\text{frei}} \leq \underbrace{b}_{\text{frei}} : (\sum \underbrace{i}_{\text{frei}} | 0 \leq \underbrace{i}_{\text{frei}} \leq \underbrace{u}_{\text{frei}} : \underbrace{i^2 + i}_{\text{frei}}) * (\sum \underbrace{i}_{\text{frei}} | 0 \leq \underbrace{i}_{\text{frei}} < \underbrace{u}_{\text{frei}} : \underbrace{i^3}_{\text{frei}}))}_{\text{gebunden}})$$

#### 1.4.3 Umbenennung (dummy renaming)

**Bed.**  $w \notin FV(R) \cup FV(P)$  ( $w$  darf nicht als FV im Ausdruck vorkommen)

**Regel:**  $(\oplus v | R : P) = (\oplus w | R[v \leftarrow w] : P[v \leftarrow w])$

**Def.**  $E[v \leftarrow F]$  bezeichnet den selben Ausdruck wie  $E$ , aber alle freien Auftreten von  $v$  ersetzt durch  $(F)$ .

**Bsp.**  $(i^2)[i \leftarrow (z + 3)] = (z + 3)^2$

**Bsp.**  $(\sum i | 0 \leq i < n : i^2) = (\sum j | (0 \leq j < n)[i \leftarrow j] : (i^2)[i \leftarrow j]) = (\sum j | 0 \leq j < n : j^2)$

## 1.5 Rechenregeln

### Empty-Range

Bei einer leeren Range, ist das Resultat die Identität:

$$(\oplus v | \text{false} : P) = \text{neutral} \oplus (\text{Identität})$$

### One-Point

Eine gebundene Variabel wird durch eine freie ersetzt:

$$(\oplus v | v = E : P) = P[v \leftarrow E] \text{ wenn: } v \notin FV(E)$$

$$\text{Bsp } (\sum i | i = j + 3 : i^2) = (i^2)[i \leftarrow (j + 3)] = (j + 3)^2$$

### Split-Off Term

Die Range wird gekürzt. Weggenommenes Element anschliessend angefügt:

$$(\oplus i | 0 \leq i < n + 1 : P) = (\oplus i | 0 \leq i < n : P) \oplus P[i \leftarrow n]$$

$$\text{Bsp: } \underbrace{(\sum i | 0 \leq i < n + 1 : i^2)}_{0^2 + 1^2 + \dots + (n-1)^2 + n^2} = \underbrace{(\sum i | 0 \leq i < n : i^2)}_{0^2 + 1^2 + \dots + (n-1)^2} + \underbrace{(i^2)[i \leftarrow n]}_{n^2}$$

### Trading

Bei mehreren Bedingungen in der Range, kann eine in den Body genommen werden:

$$(\oplus | \underbrace{R_1 \wedge R_2}_{\text{Bool}} : \underbrace{P}_{\text{irgendeinDatentyp}}) = (\oplus v | R_1 : \text{if } R_2 \text{ then } P \text{ else } v_{\oplus} \text{ endif})$$

$$\text{Bsp: } \underbrace{(\sum i | 0 \leq i < 10 \wedge \text{odd}(i) : i)}_{1+3+5+7+9} = \underbrace{(\sum i | 0 \leq i < 10 : \text{if odd}(i) \text{ then } i \text{ else } 0 \text{ endf } i)}_{0+1+0+3+0+5+0+7+0+9}$$

## 1.6 Prädikate

### 1.6.1 Definition

- $\oplus = \forall(\forall x | R : P) \implies$  für alle x im Bereich R gilt P
- $\oplus = \exists(\exists x | R : P) \implies$  es gibt ein x im Bereich R, das P erfüllt

### 1.6.2 Beispiele

- $\oplus = \forall(\forall x | R_1 \wedge R_2 : P) = (\forall u | R_1 : \text{if } R_2 \text{ then } P \text{ else true endf}) = (\forall u | R_1 : R_2 \rightarrow P)$

$R_2$	P	if	$\rightarrow$
0	0	1	1
0	1	1	1
1	0	0	0
1	1	1	1

$$\text{Sei } R_1 = \text{true} \iff (\forall u | R_2 : P) = (\forall u | \text{true} : R_2 \rightarrow P) = (\forall u | : R_2 \rightarrow P)$$

- $\oplus = \exists(\exists u | R_1 \wedge R_2 : P) = (\exists u | R_1 : \text{if } R_2 \text{ then } P \text{ else false endf}) = (\exists u | R_1 : R_2 \wedge P)$

$R_2$	P	if	$\wedge$
0	0	0	0
0	1	0	0
1	0	0	0
1	1	1	1

### 1.6.3 Übungen

**b enthält eine -1:**  $(\exists i | 0 \leq i < n : b[i] = -1)$

**b enthält genau eine -1:**  $(\exists i | 0 \leq i < n : b[i] = -1 \wedge (\forall j | 0 \leq j < n \wedge j = i : b[j] \neq -1))$

**b enthält keine -1:**  $(\forall i : \mathbb{Z} | 0 \leq i < n : b[i] \neq -1)$

## 2 Induktion/Rekursion

### 2.1 Induktion

$$(P(0) \wedge (\forall n : \mathbb{N} : P(n) \rightarrow P(n+1))) \rightarrow (\forall n : \mathbb{N} : P(n))$$

Für alle  $n : \mathbb{N}$  gilt:  $n^3 + 5n$  ist Vielfaches von 6.

**1.) Induktionsanfang** zu zeigen:  $P(0)$ , das heisst es gilt  $r : \mathbb{Z}$  mit  $0^3 + 5 * 0 = 6 * z$  klar, wähle  $z=0$

**2.) Induktionsschritt** zu zeigen:  $P(n) \rightarrow P(n+1)$  für alle  $n \geq 0$ . Sei  $n$  eine **beliebige** natürliche Zahl

**Annahme:** es gelte  $P(n)$ , das heisst es gibt  $r : \mathbb{Z}$  mit  $n^3 + 5n = 6r$

**zu Zeigen:** es gibt ein  $s : \mathbb{Z}$  mit  $(n+1)^3 + 5(n+1) = 6s$

$$\begin{aligned} & (n+1)^3 + 5(n+1) \\ &= < Arith > \text{ (arithmetische Veränderungen)} \\ & (n^3 + 3n^2 + 3n + 1) + (5n + 5) \\ &= < Arith > \\ & (n^3 + 5n) + (3n^2 + 3n + 6) \\ &= < Annahme > \\ & 6r + 3n(n+1) + 6 \\ & < n(n+1) \text{ immer durch 2 teilbar, d.h. } n(n+1) = 2t, t = \mathbb{Z} > \\ & 6r + 3 * 2t + 6 \\ &= < Arith > \\ & 6(r + t + 1) \\ &= < \text{wähle } s : \mathbb{Z} = r + t + 1 > \\ & \underline{\underline{6s}} \end{aligned}$$

#### 2.1.1 Schema

**Induktionsanfang** zu zeigen:  $P(0)$

**Induktionsschritt** zu zeigen:  $P(n) \rightarrow P(n+1)$  für alle  $n : \mathbb{N}$

Sei  $n$  eine beliebige natürliche Zahl

**Annahme:** Es gelte  $P(n)$

**zu zeigen:** Es gilt  $P(n+1)$

#### 2.1.2 Beispiel 1

Für alle  $n : \mathbb{N}$  gilt:  $(+i | 1 \leq i < n : i) = \frac{n(n+1)}{2}$

**1. Induktionsanfang:**

zu zeigen:  $< neutral+ > 0 = \frac{0(0+1)}{2} = 0 \checkmark$

**2. Induktionsschritt:**

zu zeigen:  $P(n) \rightarrow P(n+1)$  für alle  $n : \mathbb{N}$

Sei  $n$  eine beliebige Zahl

**Annahme:** es gelte  $(i + | 1 \leq i \leq n : i) = \frac{n(n+1)}{2}$

**zu zeigen:** es gilt  $(i + | 1 \leq i \leq n+1 : i) = \frac{(n+1)(n+2)}{2}$

$$\begin{aligned} & (i + | 1 \leq i \leq n+1 : i) \\ & < splitof > \\ &= (i + | 1 \leq i \leq n : i) + n+1 \end{aligned}$$

$$\begin{aligned}
&< \text{Annahme} > \\
&= \frac{n(n+1)}{2} + n + 1 \\
&< \text{Arith} > \\
&= \frac{n^2+n}{2} + \frac{2n+2}{2} \\
&< \text{Arith} > \\
&= \frac{(n+1)(n+2)}{2}
\end{aligned}$$

### 2.1.3 Beispiel 2

Für alle  $n : \mathbb{N}$  gilt:  $(+i : \mathbb{N} | 1 \leq i \leq n : 2i - 1) = n^2$

#### 1. Induktionsanfang:

$$\text{zu zeigen: } \underbrace{(+i : \mathbb{N} | 1 \leq i \leq n : 2i - 1)}_{0(<\text{neutral}+>)} = \underbrace{n^2}_{=0}$$

#### 2. Induktionsschritt:

zu zeigen:  $P(n) \rightarrow P(n+1)$  für alle  $n : \mathbb{N}$

Sei  $n$  eine beliebige Zahl

**Annahme:** es gelte  $(+i : \mathbb{N} | 1 \leq i \leq n : 2i - 1) = n^2$

**zu zeigen:** es gilt  $(+i : \mathbb{N} | 1 \leq i \leq (n+1) : 2i - 1) = (n+1)^2$

$$\begin{aligned}
&(+i : \mathbb{N} | 1 \leq i \leq (n+1) : 2i - 1) \\
&< \text{splitof} > \\
&= (+i : \mathbb{N} | 1 \leq i \leq n : 2i - 1 + 2(n+1 - 1)) \\
&< \text{Annahme} > \\
&= n^2 + 2(n+1) - 1 \\
&< \text{Arith} > \\
&= n^2 + 2n + 1 \\
&< \text{Arith} > \\
&= (n+1)^2
\end{aligned}$$

### 2.1.4 Induktion mit einem anderer Anfangswert

Sei  $k : \mathbb{N}$

$$\underbrace{P(k)}_{IA} \wedge \underbrace{(\forall n : \mathbb{N} | n \geq k : P(n) \Rightarrow P(n+1))}_{IS} \Rightarrow (\forall n : \mathbb{N} | n \geq k : P(n))$$

#### Theorem

Für alle  $n : \mathbb{N}$  mit  $n \geq 3$  gilt:  $2n + 1 < 2^n$

#### Beweis

*IA:* zu zeigen:  $P(3) \quad 2 * 3 + 1 < 2^3$

*IS:* zu zeigen:  $P(n) \Rightarrow P(n+1)$  für alle  $n \geq 3$

Sei  $n$  eine beliebige nat. Zahl  $\geq 3$ .

**Annahme:** Es gelte:  $2n + 1 < 2^n$

**zu zeigen:** Es gilt:  $2(n+1) + 1 < 2^{n+1}$

$$\begin{aligned}
&2(n+1) + 1 \\
&= < \text{Arith} > \\
&2n + 2 + 1 \\
&= < \text{Arith} > \\
&(2n+1) + 2 \\
&<< \text{Annahme} > \\
&2^n + 2 \\
&<< \text{für } n+1 \text{ ist } 2^n > 2 > \\
&2^n + 2^n \\
&= < \text{Arith} > \\
&2^{n+1}
\end{aligned}$$

## 2.2 Rekursion

$$fact : \mathbb{N} \rightarrow \mathbb{N}$$

$$fact(0) = 1$$

$$fact(n) = n + fact(n - 1), \text{ für alle } n > 0.$$

*Rekursion*

### 2.2.1 Factorial Beispiel

Für alle  $n : \mathbb{N}$  gilt:  $fact(n) = (*i : \mathbb{N} | 1 \leq i \leq n : i)$

#### 1. Induktionsanfang:

$$fact(0) = (*i : \mathbb{N} | 1 \leq i \leq 0 : i)$$

#### 2. Induktionsschritt:

Zeigen:  $P(n) \rightarrow P(n + 1)$ , für alle  $n \geq 0$

Sei  $n$  eine beliebige natürliche Zahl.

$$fact(n) = (*i : \mathbb{N} | 1 \leq i \leq n : i)$$

$$fact(n) = (*i : \mathbb{N} | 1 \leq i \leq (n + 1) : i)$$

$$(*i : \mathbb{N} | 1 \leq i \leq (n + 1) : i)$$

$$< splitoff >$$

$$= (*i : \mathbb{N} | 1 \leq i \leq n : i) * (n + 1)$$

$$< Annahme >$$

$$= fact(n) * (n + 1)$$

$$< Rekursion >$$

$$= fact(n + 1)$$

### 2.2.2 Fibonacci Beispiel

Für alle  $n : \mathbb{N}$  gilt:  $(+i : \mathbb{N} | 1 \leq i \leq n : fib(i)) = fib(n + 2) - 1$

#### 1. Induktionsanfang:

$$(+i : \mathbb{N} | 1 \leq i \leq 0 : fib(i)) = fib(0 + 2) - 1$$

#### 2. Induktionsschritt:

Zeigen:  $P(n) \rightarrow P(n + 1)$ , für alle  $n \geq 0$

Sei  $n$  eine beliebige natürliche Zahl.

$$(+i : \mathbb{N} | 1 \leq i \leq n : fib(i)) = fib(n + 2) - 1$$

$$(+i : \mathbb{N} | 1 \leq i \leq (n + 1) : fib(i)) = fib(n + 3) - 1$$

$$(+i : \mathbb{N} | 1 \leq i \leq (n + 1) : fib(i))$$

$$< splitoff >$$

$$(+i : \mathbb{N} | 1 \leq i \leq n : fib(i)) + fib(n + 1)$$

$$< Annahme >$$

$$fib(n + 2) - 1 + fib(n + 1)$$

$$< Rekursion >$$

$$= fib(n + 3) - 1$$

## 3 Zahlentheorie

### 3.1 Teilbarkeit

#### 3.1.1 Definition

$c \setminus b$

- $c$  teilt  $b$
- $b$  ist teilbar durch  $c$
- $c$  ist ein Teiler von  $b$
- $b$  ist Vielfaches von  $c$

**Beispiel**

$$7 \setminus 13 \equiv \text{false}$$
$$(-7) \setminus 14 \equiv \text{true}$$

**oft**  $c|b$  statt  $c \setminus b$

**Formal**  $c|b \equiv (\exists k : \mathbb{Z} | b = k * c)$

**Sätze**

- (1)  $c \setminus c$  (reflexiv)
- (2)  $c \setminus 0$
- (3)  $1 \setminus b$
- (4)  $c \setminus 1 \Rightarrow c = 1 \vee c = -1$
- (5)  $d \setminus c \wedge c \setminus b \Rightarrow d \setminus b$  (transitiv)
- (6)  $b \setminus c \wedge c \setminus b \Rightarrow (b = c) \vee (b = -c)$  (Antisymmetrie für Zahlen  $> 0$ )
- (7)  $b \setminus c \Rightarrow b \setminus (c * b)$
- (8)  $b \setminus c \Rightarrow (b * d) \setminus (c * d)$
- (9)  $1 < b \wedge b \setminus c \Rightarrow \neg(b \setminus (c + 1))$

**Der Fall Null**

- $0 \setminus 0 \equiv \text{true}$
- $0 \setminus 7 \equiv \text{false}$

#### 3.1.2 Beweise

**Beweis von Regel 5**

$$d \setminus c \wedge c|b \Rightarrow d \setminus b$$

Annahmen:  $d \setminus c, c \setminus b$

zu zeigen:  $d \setminus b$

wegen  $d \setminus c$  gibt es  $k_1 : \mathbb{Z}$  mit  $c = k_1 * d$

wegen  $c \setminus b$  gibt es  $k_2 : \mathbb{Z}$  mit  $b = k_2 * c$

also  $b = k_2 * c = k_2 * (k_1 * d) = (k_2 * k_1) * d = k_3 * d$

mit  $k_3 = k_2 * k_1 : \mathbb{Z}$

also  $d \setminus b$

**Beweis von  $a \setminus b \wedge a \setminus c \Rightarrow a \setminus (b + c)$**

Annahmen:  $a \setminus b, a \setminus c$

zu zeigen:  $a \setminus (b + c)$

$a \setminus b$ , also existiert ein  $k_1 : \mathbb{Z}$  mit  $b = k_1 * a$

$a \setminus c$ , also existiert ein  $k_2 : \mathbb{Z}$  mit  $c = k_2 * a$

also  $b + c = k_1 * a + k_2 * a = (k_1 + k_2) * a = k_3 * a$

mit  $k_3 = k_2 + k_1$

also  $a \setminus (b + c)$



### 3.2 Division

#### Theorem

$b, c : \mathbb{Z}, c \neq 0$ . Dann gibt es eindeutig bestimmte  $q, r : \mathbb{Z}$  mit  $b = q * c + r \wedge 0 \leq r < c$ .  
Wir nennen  $q$  den **Quotienten** und  $r$  den **Rest** von  $c$  geteilt durch  $b$ .

#### Beispiel

$$17 = 5 * 3 + 2 \wedge 0 \leq 2 < 3$$

$$b = q * c + r \wedge 0 \leq r < |c|$$

#### Eindeutigkeit

Seien $b, c : \mathbb{Z}, c > 0$ Ausserdem $q, q', r, r' : \mathbb{Z}$ mit	
$b = q * c + r$	$\wedge \quad 0 \leq r < c$
$b = q' * c + r'$	$\wedge \quad 0 \leq r' < c \mid * (-1)$
$b - b = (q - q') * c + (r - r')$	$-0 \geq -r' > -c \mid \text{umdrehen}$
$0 \Rightarrow r - r' = (q' - q) * c$	$-c < -r' \leq -0$
	$0 \leq r < c$
$r = r' \equiv q = q'$	$-c < r - r' < c$
	$-c < (q' - q) * c < c$
	$-1 < q' - q < 1$
$\Rightarrow q' - q = 0$ , also $q' = q$ , also $r = r'$	

#### Definition

Sei  $b, c, q, r : \mathbb{Z}, c \neq 0$ ,  
 $b = q * c + r, 0 \leq r < |c|$

Nach Satz  $q, r$  existent und eindeutig.  
Definieren die Funktionen **div** und **mod**.

$$\text{div}(b, c) = b \text{ div } c = \text{def } q,$$

$$\text{mod}(b, c) = b \text{ mod } c = \text{def } r \text{ div, mod: } \mathbb{Z} \times \mathbb{Z}^{\neq 0} \rightarrow \mathbb{Z}$$

### 3.3 Greatest Common Divisor (GCD)

$$x \uparrow y \Rightarrow \text{if } x \geq y \text{ then } x \text{ else } y \text{ (max)}$$

$$x \downarrow y \Rightarrow \text{if } x \leq y \text{ then } x \text{ else } y \text{ (min)}$$

Sei  $M \subseteq \mathbb{Z}, M \neq \emptyset, M$  endlich  
Dann  $\max(M) = (\uparrow b : \mathbb{Z} \mid b \in M : b)$

#### Definition

$$D_m = \{d : \mathbb{Z} \mid d \mid m\} \text{ (Menge aller Teiler von } m)$$

#### Beispiele

$$D_{12} = \{-12, -6, -4, -3, -2, -1, 1, 2, 3, 4, 6, 12\}$$

$$D_0 = \{\mathbb{Z}\}$$

$$1 \in D_m, \text{ also } D_m \neq \emptyset$$

#### Definition

$$D_{m,n} = D_m \cap D_n \text{ (Menge der gemeinsamen Teiler von } m \text{ und } n)$$

#### Beispiele

$$D_{4,6} = \{-4, -2, -1, 1, 2, 4\} \cap \{-6, -3, -2, -1, 1, 2, 3, 6\} = \{-2, -1, 1, 2\}$$

$$D_{2,0} = \{-3, -1, 1, 3\} \cap \mathbb{Z} = \{-3, -1, 1, 3\}$$

$$D_{0,0} = \mathbb{Z}$$

$$1 \in D_{m,n}, \text{ also } D_{m,n} \neq \emptyset$$

Sei  $b, c : \mathbb{Z}, b \neq 0 \vee c \neq 0$ .

Dann ist  $D_{b,c} \neq \emptyset$  und  $D_{b,c}$  hat ein grösstes Element.

$$\gcd(b,c) = b \gcd c = \max(D_{b,c})$$

$$\gcd(0,0) = 0 \quad \gcd 0 = 0$$

### Satz

- a)  $b \gcd b = |b|$
- b)  $0 \gcd b = |b|$
- c)  $b \gcd c = |b| \gcd |c|$  (ged - Algo für N genug)
- d)  $b \gcd c = c \gcd b$
- e)  $b = a * c + d \Rightarrow b \gcd c = c \gcd d$

### Beweis

Annahme: Sei  $b = a * c + d$

zu zeigen: Es gilt  $b \gcd = c \gcd d$

#### 1. Fall:

$b = c = 0$  Dann  $d = 0$

Dann  $b \gcd c = c \gcd d$

#### 2. Fall:

$b \neq 0 \vee c \neq 0$

Wir zeigen  $D_{b,c} = D_{c,d}$ . Damit folgt  $b \gcd = \max(D_{b,c}) = \max(D_{c,d}) = c \gcd d$ .

zu zeigen:  $D_{b,c} = D_{c,d}$  also  $t \in D_{b,c} \Leftrightarrow t \in D_{c,d}$

**Teil 1** ( $t \in D_{b,c} \Rightarrow t \in D_{c,d}$ )

$t \in D_{b,c}$

$\Rightarrow \langle Def D_{m,n} \rangle$

$t \in D_b \cap D_c$

$\Rightarrow \langle Def \cap \rangle$

$t \in D_b \wedge t \in D_c$

$\Rightarrow \langle Def D_m \rangle$

$t \setminus b \wedge t \setminus c$

$\Rightarrow \langle Def \setminus \text{ mit } k_1, k_2 : \mathbb{Z} \rangle$

$b = k_1 * t \wedge c = k_2 * t$

$\Rightarrow \langle d = b - a * c, \text{ nach Annahme} \rangle$

$d = k_1 * t - a * (k_2 * t)$

$\Rightarrow \langle Arith \rangle$

$d = t * (k_1 - a * k_2)$

$\Rightarrow \langle Def \setminus \text{ mit } k_1 - a * k_2 : \mathbb{Z} \rangle$

$t \setminus d \wedge t \setminus c$

$\Rightarrow \langle Def D_m \rangle$

$t \in D_d \wedge t \in D_c$

$\Rightarrow \langle Def \cap \rangle$

$t \in D_d \cap D_c$

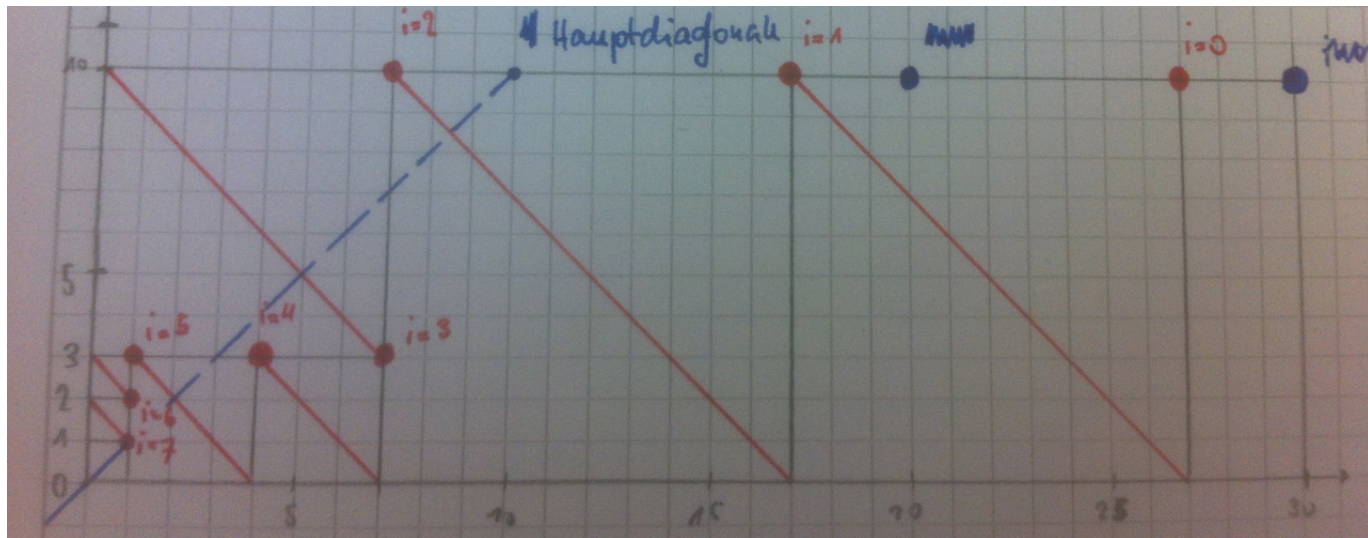
$\Rightarrow \langle Def D_{d,c} \rangle$

$t \in D_{d,c}$

**Teil 2** ( $t \in D_{b,c} \Leftarrow t \in D_{c,d}$ )

tbd

### 3.4 Euklid ged(27,10)



**Regel e:**  $b = a * c + d \Rightarrow b \text{ gcd } c = c \text{ gcd } d$

$b \text{ gcd } c = c \text{ gcd } (b - ac)$  Setze  $a = 1$

$x \text{ gcd } y = y \text{ gcd } (x - y)$

$\text{gcd}(x, y) = \text{gcd}(y, x - y)$

$\text{gcd}(x, y) = \text{gcd}(x - y, y)$  : Korollar zu Regel e

**Algo:**

Algo terminiert. Sei  $n$  Anzahl Iterationen. Dann  $\text{gcd}(bc) = x_n$  oder  $y_n$  (beides richtig)

$(x_0, y_0) = b, c$

$i := 0$

**while**  $x_i \neq y_i$  **do**

**if**  $x_i < y_i$  **then**  $(x_{i+1}, y_{i+1}) = (x_i - y_i, y_i)$

**else**

$(x_{i+1}, y_{i+1}) = (x_i, y_i - x_i)$

**endif**

$i := i + 1$

**end**

### 3.5 Erweiterter Euklid

**Satz:**

$$\underbrace{r_{-2}}_A = a \quad \underbrace{r_{-1}}_B = b$$

$$\underbrace{p_{-2}}_C = 0 \quad \text{X} \quad \underbrace{p_{-1}}_D = 1$$

$$\underbrace{q_{-2}}_E = 1 \quad \underbrace{q_{-1}}_F = 0$$

$i := 0$

**while**  $\underbrace{r_{i-1}}_G > 0$  **do**

**berechne**  $c_i$  und  $r_i$  mit  $\underbrace{r_{i-2}}_H = c_i * \underbrace{r_{i-1}}_I + \underbrace{r_i}_J \wedge 0 \leq \underbrace{r_i}_J < \underbrace{r_{i-1}}_I$

$$\underbrace{p_i}_K = c_i * \underbrace{p_{i-1}}_I + \underbrace{p_{i-2}}_J$$

```


$$q_i \stackrel{L}{=} c_i * q_{i-1} + 1_{i-2}$$


$$i := i + 1$$

end

```

Seien  $a, b : \mathbb{N}$ .

Dann gibt es Zahlen  $x, y : \mathbb{Z}$  mit  $x * a + y * b = \gcd(a, b)$ .

Wir konstruieren diese Zahlen durch erweiterten Euklidischen Algorithmus.

- 1)  $r_{n-1} = \gcd(a, b)$
- 2)  $p_n \gcd(a, b) = a$
- 3)  $q_n \gcd(a, b) = b$
- 4)  $a * \frac{q_{n-1}}{(-1)^{n-1}} + b * \frac{p_{n-1}}{(-1)^n} = \gcd(a, b)$

i	$r_i$	$c_i$	$p_i$	$q_i$	
-2	654		0	1	
-1	444		1	0	
0	210	1	1	1	=b
1	24	2	3	2	= 210
2	18	8	25	17	= 24
3	6	1	28	19	= 18
4	0	3	109	74	= 0

### Beweis

Es gibt offensichtlich  $n : \mathbb{N}$  mit  $r_n = 0 \wedge (\forall i | 0 \leq i < n : r_i > 0)$

Invarianten für alle  $i : 0 \leq i \leq n + 1$

$$I_1: \gcd(r_{i-2}, r_{i-1}) = \gcd(a, b)$$

$$I_2: p_{i-2} * r_{i-1} + p_{i-1} * r_{i-2} = a$$

$$I_3: q_{i-2} * r_{i-1} + q_{i-1} * r_{i-2} = b$$

$$I_4: q_{i-2} * p_{i-1} - q_{i-1} * p_{i-2} = (-1)^{i-1}$$

### Beispiel $I_2$ :

#### Induktionsanfang

$$\begin{aligned}
& i = 0, p_{-2} * r_{-1} + p_{-1} * r_{-2} \\
& = < C, B, D, A > \\
& 0 * b + 1 * a \\
& = < Arith > \\
& a
\end{aligned}$$

#### Induktionsschritt

zu zeigen:  $I_2(i) \wedge r_{i-1} > 0 \Rightarrow I_2(i+1)$  für alle  $i$  mit  $0 \leq i \leq n$ .

Sei  $n$  eine  $\mathbb{N}$  mit  $0 \leq i \leq n$

Annahme: es gibt  $I_2(i) \wedge r_{i-1} > 0$

zu zeigen: es gibt  $I_2(i+1)$

$$\begin{aligned}
& p_{(i+1)-2} * r_{(i+1)-1} + p_{(i+1)-1} * r_{(i+1)-2} \\
& = < Arith > \\
& p_{i-1} * r_i + p_i * r_{i-1} \\
& = < H, < > \\
& p_{i-1} * (r_{i-2} - c_i * r_{i-1}) + (c_i * p_{i-1} + p_{i-2}) * r_{i-1} \\
& = < Arith > \\
& p_{i-1} * r_{i-2} + p_{i-2} * r_{i-1}
\end{aligned}$$

=< Annahme >

a

Es gilt insbesondere:  $p_n * r_{n-1} + p_{n-1} * \overbrace{r_n}^0 = a$   
 $p_n * \gcd(a, b) = a \rightarrow (2)$

### 3.6 Primzahlen

**Def.** Eine Zahl  $n : \mathbb{N}$  heisst prim, wenn ihre einzigen positiven Teiler 1 und n sind, mit  $1 \neq p$ :

p ist prim =  $p \geq 2 \wedge (\forall i : \mathbb{N} | 1 < i < p : i \nmid p)$   
 $|D_p^+| = 2$

Sonst heisst, p zusammengesetzt.

**Def.** Zahlen  $b, c : \mathbb{N}$  heisst relativ prim ( $b \perp c$ ) falls der  $\gcd(b, c) = 1$

**Satz** Jede natürliche Zahl  $n : \mathbb{N}, n \geq 1$  kann als Produkt von PRIMzahlen geschrieben werden.  
D.h. es existiert eine Zahl  $m : \mathbb{N}$  und m Primzahlen  $(p_0, p_1, \dots, p_{m-1})$

$n := (*i | 0 \leq i < m : p^i)$

**Satz** Sei  $n : \mathbb{N}, n \geq 1$ . Die Zerlegung von n in Primzahlen ist eindeutig.

**Bsp.**  $328 = 2 * 164 = 2 * 2 * 82 = 2 * 2 * 2 * 41$

$328 = 2^3 * 41^1$

$328 = p_0^{\alpha_0}, p_1^{\alpha_1}, \dots, p_{m-1}^{\alpha_{m-1}}$

**Satz** Es gibt unendlich viele Primzahlen.

### 3.7 Modulare Arithmetik

#### 3.7.1 Kongruenz

**Def**  $a, b : \mathbb{Z}, m : \mathbb{N}^{>0}$ . a heisst kongruent b modulo n, wenn  $n \mid (a - b)$

$a \equiv_n b = n \mid (a - b)$

$a \equiv b \pmod{n}$

**Bsp.**  $5 \equiv_3 7 = false$

$5 \equiv_3 8 = true$

**Satz** Sei  $n : \mathbb{N}^{>0} \equiv_n$  eine Äquivalenzrelation.

**Beweis**

**Symmetrisch**  $n \mid (a - b) \rightarrow a - b = k * n \rightarrow b - a = (-k) * n \rightarrow n \mid (b - a)$

**Reflexiv**  $a \equiv_n a \rightarrow n \mid (a - a) \rightarrow n \mid 0$

**Transitiv**  $a \equiv_n b \wedge b \equiv_n c \rightarrow a \equiv_n c$

#### 1. Annahme

$a \equiv_n b$

< def  $k : \mathbb{Z}$  >

$a - b = k_n * n$

#### 2. Annahme

$b \equiv_n c$

< def  $k_2 : \mathbb{Z}$  >

$(b - c) = k_2 * n$

**zu zeigen**

$$\begin{aligned}
 a &\equiv_n c \\
 a - b &= k_1 * n \\
 b - c &= k_2 * n \\
 a - c &= (k_1 + k_2) * n \\
 &< \text{def} \setminus \text{mit } (k_1 + k_2) : \mathbb{Z} > \\
 n &\setminus (a - c)
 \end{aligned}$$

### 3.7.2 Restklasse modulo n

**Def.**  $[a]_n = \{n : \mathbb{Z} \mid a + k * n\}$   
 Jedes  $a' \in [a]_n$  ist ein Repräsentant von  $[a]_n$

**Satz** Sei  $a \equiv_n a'$  Dann  $[a] = [a']$

**Bsp.**  $[3]_7 = \{k : \mathbb{Z} \mid 3 + k * 7 : \} = \{\dots, -11, -4, 3, 10, 17, \dots\}$   
 $[-4]_7 = \{k : \mathbb{Z} \mid -4 + k * 7 : \} = \{\dots, -11, -4, 3, 10, 17, \dots\}$   
 $[2]_7 = \{k : \mathbb{Z} \mid 2 + k * 7 : \} = \{\dots, -12, -5, 2, 9, 16, \dots\}$

### 3.7.3 Zerlegung, Partition

**Def.**  $Z_n = \{a : \mathbb{N} \mid 0 \leq a < n : [a]_n\}$

**Bsp.**  $Z_3 = \{[0]_3, [1]_3, [2]_3\}$   
 $Z_3 = \{\{\dots, -3, 0, 3, 6, \dots\},$   
 $\quad \{\dots, -2, 1, 4, 7, \dots\},$   
 $\quad \{\dots, -1, 2, 5, 8, \dots\}\}$

**Satz**  $a, a', b, b' : \mathbb{Z}, n : \mathbb{N}^{>0}$   
 $a \equiv_n a'$  und  $b \equiv_n b'$   
 $a + b \equiv_n a' + b'$   
 $a * b \equiv_n a' * b'$

**Def**  $a, b : \mathbb{Z}, n : \mathbb{N}^{>0}$   
 $[a]_n +_n [b]_n = [a + b]_n$   
 $[a]_n *_n [b]_n = [a * b]_n$

**Bsp.**  $[5]_n +_n [4]_n = [5 + 4]_n = [9]_7 = [2]_7$

### 3.7.4 Gruppe

Eine Gruppe  $(S, \oplus)$  ist eine Menge S mit einer Operation  $\oplus$  aus S mit:  $\oplus : S \times S \rightarrow a'$

#### 1. Abgeschlossenheit

$$\begin{aligned}
 a \oplus b &\in S' \\
 \forall a, b &\in S'
 \end{aligned}$$

#### 2. Identität, neutrales Element

$$\begin{aligned}
 \exists e \in S' &\text{ mit } e \oplus a = a = a \oplus e \\
 \forall a &
 \end{aligned}$$

#### 3. Assoziativität

$$\begin{aligned}
 a \oplus (b \oplus c) &= (a \oplus b) \oplus c \\
 \forall a, b, c &\in S
 \end{aligned}$$

#### 4. Inverse

für jedes  $a \in S$  gibt es ein  $b \in S$  mit  $a \oplus b = b \oplus a = e$

#### 5. Symmetrie

$$\begin{aligned}
 a \oplus b &= b \oplus a \\
 \forall a, b &
 \end{aligned}$$

dan heisst  $(S', \oplus)$  abelsche Gruppe.

### 3.7.5 endliche Gruppe

Ist  $S$  endlich, dann ist  $(S, \oplus)$  eine endliche Gruppe.

**Def.**  $(Z_n, +_n)$  heisst additive Gruppe modulo  $n$ .

**Satz**  $(Z_n, +_n)$  ist endliche abelsche Gruppe.

**Beispiel:**

$+_4$	$[0]_4$	$[1]_4$	$[2]_4$	$[3]_4$
$[0]_4$	$[0]_4$	$[1]_4$	$[2]_4$	$[3]_4$
$[1]_4$	$[1]_4$	$[2]_4$	$[3]_4$	$[0]_4$
$[2]_4$	$[2]_4$	$[3]_4$	$[0]_4$	$[1]_4$
$[3]_4$	$[3]_4$	$[0]_4$	$[1]_4$	$[2]_4$

**Beweis:**

1. Abgeschlossen: klar aus Def
2. Identität  $[0]_n$
3. Asso: aus Def
4. Inv:  $[a]_n + [n - a]_n = [a + (n - a)]_n = [n]_n = [0]_n$
5. Sym: aus Def
6.  $|Z_n| = n$  endlich

### 3.7.6 reduzierte Menge

**Def.**  $Z_n^* = \{a : \mathbb{Z} | \gcd(a, n) = 1 : [a]_n\}$   
heisst reduzierte Menge Des Rest modulo  $n$

**Bsp.**  $Z_{10}^* = \{[1]_{10}, [3]_{10}, [7]_{10}, [9]_{10}\}$

$*_{10}$	$[1]_{10}$	$[3]_{10}$	$[7]_{10}$	$[9]_{10}$
$[1]_{10}$	$[1]_{10}$	$[3]_{10}$	$[7]_{10}$	$[9]_{10}$
$[3]_{10}$	$[3]_{10}$	$[9]_{10}$	$[1]_{10}$	$[7]_{10}$
$[7]_{10}$	$[7]_{10}$	$[1]_{10}$	$[9]_{10}$	$[3]_{10}$
$[9]_{10}$	$[9]_{10}$	$[7]_{10}$	$[3]_{10}$	$[1]_{10}$

**Beweis:**

1. Abgeschlossen:  

$$\underbrace{\gcd(a, n) = 1}_{a \in Z_n^*} \wedge \underbrace{\gcd(b, n) = 1}_{b \in Z_n^*} \Rightarrow \underbrace{\gcd(a * b, n) = 1}_{a * b \in Z_n^*}$$
2. Identität:  
 $[1]_n$  weil  $[a]_n * [1]_n = [a * 1]_n = [a]_n$
3. Assoziativität:  

$$\begin{aligned} & [a]_n * ([b]_n * [c]_n) \\ &= \langle \text{Def } *_n \rangle \\ & [a]_n * [b * c]_n \\ &= \langle \text{Def } *_n \rangle \\ & [a * (b * c)]_n \\ &= \langle \text{Asso } * \rangle \\ & [(a * b) * c]_n \\ &= \langle \text{Def } *_n \rangle \\ & [a * b]_n * [c]_n \\ &= \langle \text{Def } *_n \rangle \\ & ([a]_n * [b]_n) * [c]_n \end{aligned}$$

4. Inv: (erweiterter Euklid)

$$\exists x, y$$

$$x * b + y * c = \gcd(b, c)$$

$$x * a + y * n = \gcd(a, n) = 1$$

$$\text{also } x * a \equiv_n 1$$

$[x]_n$  ist also multiplikativ Inverses

$$[x]_n * [a]_n = [x * a]_n = [1]_n$$

x ausrechenbar mit erweitertem Euklid

5. Sym: wie Asso

6.  $|Z_n^*| < |Z_n|'$  endlich

### Berechnung multiplikatives Inverses, am Beispiel 105mod31

1. Überprüfen ob die beiden Werte koPrim sind.

$$105 *_{\text{x}} -31 *_{\text{y}} = 1$$

$$105 = 3 * 31 + 12$$

$$31 = 2 * 12 + 7$$

$$12 = 1 * 7 + 5$$

$$7 = 1 * 5 + 2$$

$$5 = 2 * 2 + 1$$

$$2 = 2 * 1 + 0$$

- Da diese Abfolge erstellen lässt ohne Rest, sind die beiden Zahlen koPrim

2. Rücksubstitution

$$1 = 5 - (2 * 2)$$

$$1 = 5 - 2 * (7 - 5) = 3 * 5 - 2 * 7 - 1$$

$$1 = 5 - 2 * (7 - 5) = 3 * 5 - 2 * 7 - 1$$

$$1 = 3 * (12 - 7) - 5 * (31 - 2 * 12) = -5 * 31 + 13 * 12$$

$$1 = 13 * 12 - 5 * 31$$

$$1 = 13 * (105 - 3 * 31) - 5 * 31$$

$$1 = 13 * 105 - 39 * 31 - 5 * 31$$

$$1 = \underbrace{13}_{\text{gesuchter Wert}} * 105 - 44 * 31$$

gesuchter Wert für  $105^{-1} =_{31} 13$