2.1.5 Definition (Lex-, Grlex- und Grevlex-Ordnungen).

Auf $\mathbb{Z}_{\geq 0}^n$ führen wir die folgenden drei Ordnungen ein:

$$\alpha \succ_{lex} \beta \;:\Longleftrightarrow\; \alpha_1 = \beta_1, \dots, \alpha_{i-1} = \beta_{i-1}, \alpha_i > \beta_i \text{ für ein } i \in \{1, \dots, n\}$$

(a, b, f, a, h, r, e, n) $\succ_{lex}$
(a, b, e, e, h, n, e, n)
a = 0, b = 1, c = 2, d = 3 ...

$$\alpha \succ_{grlex} \beta \;:\Longleftrightarrow\; |\alpha| > |\beta| \text{ oder } |\alpha| = |\beta| \text{ und } \alpha \succ_{lex} \beta.$$

$$|\alpha| = \sum_{i=1}^{n} \alpha_i$$

$$\alpha \succ_{grevlex} \beta \;:\Longleftrightarrow\; |\alpha| > |\beta| \text{ oder } |\alpha| = |\beta| \text{ und } \alpha_i < \beta_i, \alpha_{i+1} = \beta_{i+1}, \dots, \alpha_n = \beta_n \text{ für ein } i \in \{1, \dots, n\}$$

$\succ_{lex}$ ist die lexikographische Ordnung

$\succ_{grlex}$ ist die graduierte lexikographische Ordnung

$\succ_{grevlex}$ ist die sogenannte umgekehrte graduierte lexikographische Ordnung

Was wollen wir für Eigenschaften für die Ordnungen? Welche ist besser für die Praxis?

Lex (3. ?)

$$
\begin{aligned}
&>_{lex} (3,1,10000)\\
&>_{lex} (3,1,9999)\\
&>_{lex} (3,1,9998)\\
&>_{lex} \quad \vdots\\
&>_{lex} (3,1,1)\\
&>_{lex} (3,1,0)\\
&>_{lex} (3,0,1000000)\\
&>_{lex} (3,0,999999)\\
&\quad \vdots\\
&>_{lex} \quad \vdots\\
&>_{lex} (3,0,0)\\
&>_{lex} (2,100000000,100000000)\\
&\quad \vdots\\
&\quad \vdots\\
&\quad (0,0,0)
\end{aligned}
$$

Bei $<_{grlex}$ kann für ein gegebenes $\alpha \in \mathbb{Z}_{\geq 0}^n$ abschätzen, wieviele β's die Bedingung $\beta <_{grlex} \alpha$ erfüllen, denn

$$\beta <_{grlex} \alpha \implies |\beta| \leq |\alpha|.$$

Bei $<_{grevlex}$ ist die Situation ähnlich, diese Ordnung ist aber für die Praxis noch besser.

2.1.6 Proposition. $<_{lex}$, $<_{grlex}$ und $<_{grevlex}$ sind Monomordnungen auf $\mathbb{Z}^n_{\geq 0}$.

Beweis: Wir beweisen die Behauptung nur für

$\succ_{lex}$.

<u>$\succeq_{lex}$ ist eine totale Ordnung:</u>

$\alpha \succeq_{lex} \beta \iff \alpha = \beta$ oder $\alpha \succ_{lex} \beta$.

- $\alpha \succeq_{lex} \alpha$ für alle $\alpha \in \mathbb{Z}^n_{\geq 0}$ ist klar.
- $\alpha \succeq_{lex} \beta, \beta \succeq_{lex} \gamma \Rightarrow \alpha \succeq_{lex} \gamma$:

  Wenn $\alpha = \beta$ oder $\beta = \gamma$ ist, dann ist diese Implikation trivial. Es bleibt zu zeigen, dass

  $$\alpha \succ_{lex} \beta, \beta \succ_{lex} \gamma \Rightarrow \alpha \succ_{lex} \gamma$$

  gilt.

  $\alpha \succ_{lex} \beta$ heißt $\alpha_1 = \beta_1, \dots, \alpha_{i-1} = \beta_{i-1}, \alpha_i > \beta_i$ für ein $i \in \{1, \dots, n\}$

  $\beta \succ_{lex} \gamma$ heißt $\beta_1 = \gamma_1, \dots, \beta_{j-1} = \gamma_{j-1}, \beta_j > \gamma_j$ für ein $j \in \{1, \dots, n\}$.

  Für $m = \min\{i, j\}$ gilt:

  $\alpha_1 = \beta_1 = \gamma_1, \dots, \alpha_{m-1} = \beta_{m-1} = \gamma_{m-1}$

  und $\alpha_m \leq \beta_m \leq \gamma_m$

  mit $\alpha_m < \beta_m$ im Fall $m = i$

  und $\beta_m < \gamma_m$ im Fall $m = j$

  Das ergibt: $\alpha_m < \gamma_m$.

  Das heißt: $\alpha \succ_{lex} \gamma$.

- Für alle $\alpha, \beta \in \mathbb{Z}^n_{\geq 0}$ gilt $\alpha \succeq_{lex} \beta$ oder $\beta \succeq_{lex} \alpha$:

  Für $\alpha = \beta$ ist das klar. Für $\alpha \neq \beta$

wähle $i \in \{1, \dots, n\}$ mit

$\alpha_1 = \beta_1, \dots, \alpha_{i-1} = \beta_{i-1}$ und $\alpha_i \neq \beta_i$

Ob $\alpha \preccurlyeq_{lex} \beta$ oder $\alpha \succ_{lex} \beta$ gilt, wird durch den Vergleich von α_i und β_i entschieden.

$\alpha, \beta, \gamma \in \mathbb{Z}_{\geq 0}^n, \alpha \preccurlyeq_{lex} \beta \Rightarrow \alpha + \gamma \preccurlyeq_{lex} \beta + \gamma.$

$\alpha_1 = \beta_1, \dots, \alpha_{i-1} = \beta_{i-1}, \alpha_i > \beta_i$

für ein $i \in \{1, \dots, n\}$.

Daraus folgt.

$\alpha_1 + \gamma_1 = \beta_1 + \gamma_1, \dots, \alpha_{i-1} + \gamma_{i-1} = \beta_{i-1} + \gamma_{i-1}$

und $\alpha_i + \gamma_i > \beta_i + \gamma_i$.

$\Rightarrow \alpha + \gamma \succ_{lex} \beta + \gamma.$

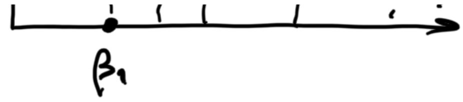
Jede Teilmenge $A \subseteq \mathbb{Z}_{\geq 0}^n$ mit $A \neq \emptyset$ besitzt das kleinste Element bzgl. $\succ_{lex}$.

Sei

$A_1 = \{\alpha_1 : (\alpha_1, \dots, \alpha_n) \in A\} \subseteq \mathbb{Z}_{\geq 0}$

Wegen $A_1 \subseteq \mathbb{Z}_{\geq 0}$ wird in A_1 das Minimum erreicht, d.h. es gibt $\beta_1 := \min(A_1)$ und dieses Minimum wird ~~das~~ an einem Element

$(\beta_1, \alpha_2, \dots, \alpha_n) \in A$ erreicht.



Wir betrachten die Menge

$$\{(\beta_1, \alpha_2, \dots, \alpha_n) : (\beta_1, \alpha_2, \dots, \alpha_n) \in A\}$$

Sei $A_2 = \{\alpha_2 : \alpha_2, \dots, \alpha_n \in \mathbb{Z}_{\geq 0}, (\beta_1, \alpha_2, \dots, \alpha_n) \in A\}$

Wir setzen $\beta_2 = \min(A_2)$ usw.

Wir definieren

$$\beta_i = \min(A_i) \text{ mit}$$

$$A_i = \min\{\alpha_i : \alpha_i, \dots, \alpha_n \in \mathbb{Z}_{\geq 0}, (\beta_1, \dots, \beta_{i-1}, \alpha_i, \dots, \alpha_n) \in A\}$$

Auf diese Weise definieren wir

$$\beta = (\beta_1, \dots, \beta_n) \in A.$$

Wir behaupten, dass β in A bzgl. $>_{lex}$ das Maximum ist.

Sei $\alpha \in A \setminus \{\beta\}$ beliebig.

Wir fixieren ein $i \in \{1, \dots, n\}$ mit

$$\alpha_1 = \beta_1, \dots, \alpha_{i-1} = \beta_{i-1} \text{ und } \alpha_i \neq \beta_i$$

$$\beta_i = \min\{\alpha_i : \alpha_i, \dots, \alpha_n \in \mathbb{Z}_{\geq 0}, (\beta_1, \dots, \beta_{i-1}, \alpha_i, \dots, \alpha_n) \in A\}$$

Nach der Wahl von β_i ist $\beta_i < \alpha_i$.

$\Rightarrow \beta >_{lex} \alpha$. $\square$

<u>Bemerkung.</u> Durch das Umnummerieren der Variablen $x_1, \dots, x_n$ entstehen $n!$

Varianten der Ordnung $\leq_{lex}$, $\leq_{grlex}$ und $\leq_{grevlex}$.

2.1.7. Definition. Sei $f = \sum_\alpha a_\alpha x^\alpha \in k[x_1, \dots, x_n] \setminus \{0\}$ und sei $\leq$ eine Monomordnung auf $\mathbb{Z}^n_{\geq 0}$.

(i) Der Multigrad $\mathrm{mdeg}(f)$ ist das maximale $\alpha \in \mathbb{Z}^n_{\geq 0}$ bzgl. $\leq$ mit $a_\alpha \neq 0$.

(ii) Das Leitmonom ist $LM(f) = x^{\mathrm{mdeg}(f)}$

(iii) Der Leitterm ist $LT(f) = a_{\mathrm{mdeg}(f)} \cdot x^{\mathrm{mdeg}(f)}$

Für das Nullpolynom setzen wir
$$\mathrm{mdeg}(0) = -\infty \text{ und}$$
$$-\infty \leq \alpha \quad \text{für alle } \alpha \in \mathbb{Z}^n_{\geq 0}.$$

2.1.8. Lemma Seien $f, g \in k[x_1, \dots, x_n] \setminus \{0\}$. Dann gilt:

(i) $\mathrm{mdeg}(fg) = \mathrm{mdeg}(f) + \mathrm{mdeg}(g)$

(ii) $\mathrm{mdeg}(f+g) \leq \max\{\mathrm{mdeg}(f), \mathrm{mdeg}(g)\}$ wobei wir hier die max-Operation bzgl. $\leq$ meinen.

(iii) $\mathrm{mdeg}(f) \neq \mathrm{mdeg}(g) \Rightarrow$
$$\mathrm{mdeg}(f+g) = \max\{\mathrm{mdeg}(f), \mathrm{mdeg}(g)\}$$

Beweis: Aufgabe.

2.2 Ein Divisionsalgorithmus für multivariate Polynome

Die Eingabe für die Polynomdivision besteht aus $f \in k[x_1, \dots, x_n]$ und $g_1, \dots, g_s \in k[x_1, \dots, x_n] \setminus \{0\}$. Gesucht wird eine Darstellung der Form

$$f = a_1 f_1 + \dots + a_s f_s + r$$

$a_1, \dots, a_s \in k[x_1, \dots, x_n]$ sind die Quotienten

$r \in k[x_1, \dots, x_n]$ ist der Rest.

Man beachte:

- Wir teilen durch s Polynome (nicht unbedingt durch ein Polynom).
- Beim Teilen wird der Prozess etwas anders organisiert als im univariaten Fall wegen des folgenden Problems:

$$x_1 \cdot x_2^{20} x_3^3 x_4^4 \succ_{lex} x_1 x_2^{20} x_3^2 x_4^7$$

aber beim Teilen erhält man

$$\frac{x_1 \cdot x_2^{20} \cdot x_3^3 \cdot x_4^4}{x_1 \cdot x_2^{20} \cdot x_3^2 \cdot x_4^7} = \underbrace{x_3 \cdot x_4^{-3}}_{\text{kein Monom!}}$$

Daher werden beim Teilen manchmal Terme innerhalb der Iteration direkt in den Rest aufgenommen.

Die Idee des Algorithmus: Man benutzt die Darstellung

$$a_1 f_1 + \dots + a_s f_s + p + r = f$$

$a_1, \dots, a_s$ – Quotienten im Aufbau

r – Rest im Aufbau

p – was noch zu teilen ist.

(beim Iterieren wird p zwischen den Quotienten und dem Rest verteilen)

Sobald $p = 0$ ist terminiert der Algorithmus

Im Fall $p \neq 0$, werden $i = 1, \dots, s$ durchprobiert, bis man ein i findet bei dem $LT(p)$ durch $LT(f_i)$ teilbar ist.

Wenn man kein solches i findet, wird der Leitterm von p direkt in den Rest aufgenommen:

$$r := r + LT(p)$$
$$p := p - LT(p)$$

Wenn man ein solches i findet, dann benutzen

$$a_i f_i + p = \left(a_i + \frac{LT(p)}{LT(f_i)}\right) f_i + p - \frac{LT(p)}{LT(f_i)} f_i$$

Man setzt:

$$\left.\begin{aligned} T &:= \frac{LT(p)}{LT(f_i)} \\ a_i &:= a_i + T \\ p &:= p - T \cdot f_i \end{aligned}\right\} \text{ wie im univariaten Fall.}$$

Der vollständige Algorithmus:

<u>Multivariate-Division</u>$(f, f_1, \dots, f_s)$

Annahme: $f \in k[x_1, \dots, x_n]$, $f_1, \dots, f_s \in k[x_1, \dots, x_n] \setminus \{0\}$

Ergebnis: Rückgabe der Quotienten $a_1, \dots, a_s$ und des Rests r der Division von f durch $f_1, \dots, f_s$.

```
r := 0            ▷ Rest im Aufbau
p := f            ▷ Was noch zu teilen ist
a_i := 0 (i=1,..,n)  ▷ Quotienten im Aufbau
▷ Invariante: a_1 f_1 + ... + a_s f_s + p + r = f
while p ≠ 0:
    if not Quotients-Updated():
```

```
            r := r + LT(p)
            p := p - LT(p)
        end
    end
    return a_1, ..., a_s, r
```

Quotients-Update()

Annahme: wir benutzen alles aus der Funktion ~~oben~~

Ergebnis: wir versuchen einen der Quotienten zu aktualisieren, und berichten, ob es geklappt hat.

```
for i := 1, ..., s:
    if LT(p) durch LT(f_i) teilbar:
        T := LT(p) / LT(f_i)
        a_i := a_i + T
        p := p - T · f_i
        ▷ Multigrad von p wird geringer
        return True
    end
end
return False
```

2.2.1 Beispiel. Wir fixieren $>_{lex}$. Wir teilen $f = x^2y + xy^2 + y^2$ durch $f_1 = \underline{xy} - 1$,
$f_2 = \underline{y^2} - 1$.

Eingabe:	Rückgabe
$f = x^2y + xy^2 + y^2$	$r = x + y + 1$
$f_1 = xy - 1$	$a_1 = x + y$
$f_2 = y^2 - 1$	$a_2 = 1$

	Rechenschritte
$\underline{x^2y} + xy^2 + y^2$	$x f_1 = x^2y - x$
$\underline{xy^2} + x + y^2$	$y f_1 = xy^2 - y$
$\underline{x} + y^2 + y$	x in der Rest.
$\underline{y^2} + y$	$1 \cdot f_2 = y^2 - 1$
$\underline{y} + 1$	y in der Rest
1	1 in den Rest
0	

$$f = \underbrace{(x+y)}_{\uparrow} f_1 + \underbrace{1}_{\uparrow} \cdot f_2 + \underbrace{(x+y+1)}_{\text{Rest.}}$$

Quotienten