SS 2010 2010/04/29

Exercises to the Lecture Computer Algebra
Sheet 3

Prof. Dr. Klaus Madlener Delivery until 2010/05/07

Exercise 1: [Ringe]

a) Ist für Integritätsbereiche R, S der Ring $R+S$ (die direkte Summe) ebenfalls immer ein Integritätsbereich? Zeigen oder widerlegen Sie.

b) Sei $I = \{f \in \mathbb{R}[x] \mid f(5) = 0\}$ die Menge der reellen Polynome, die 5 als eine Nullstelle haben. Zeigen Sie, dass I ein Ideal in $\mathbb{R}[x]$ ist. Geben Sie einen Isomorphismus $\mathbb{R}[x]/I \to \mathbb{R}$ an.

Exercise 2:

a) Bestimmen Sie in $\mathbb{Z}[[x]]$ das Inverse zur formalen Potenzreihe

$$a(x) = \sum_{k=0}^{\infty} a_k x^k = 1 + x + 2x^2 + 3x^3 + 5x^4 + \cdots$$

mit $a_k = a_{k-1} + a_{k-2}$ für $k \geq 2$.

Exercise 3: [Simplifikation]

a) Betrachten Sie die folgende "'Definition"' einer *expandierten kanonischen Form* für multivariate Polynom-Ausdrücke über einem Integritätsbereich D:

1. Ausmultiplizieren aller Produkte von Polynomen
2. Terme gleichen Grades zusammenfassen
3. Terme nach fallenden Grad ordnen

Ist dies ein kanonischer Simplifikator gemäß der Definition aus der Vorlesung? Welche Angaben brauchen Sie, um diese Frage beantworten zu können? Ist die obige Definition eindeutig?

Wenden Sie die Vorschrift auf folgenden Ausdruck an:

$$\begin{aligned} a(x,y) &= ((x^2 - xy + x) + (x^2 + 3) \cdot (x - y + 1)) \cdot \\ &\quad ((y^3 - 3y^2 - 9y - 5) + x^4 \cdot (y^2 + 2y + 1)) \end{aligned}$$

b) Finden Sie "'einfachste"' Ausdrücke, die äquivalent zu den folgenden Ausdrücken sind:

- $a(x,y) = \dfrac{1}{x^9 + x^8y + x^7y^2 + x^6y^3 + x^5y^4 + x^4y^5 + x^3y^6 + x^2y^7 + xy^8 + y^9}$
- $b(x,y) = \dfrac{x-4}{x^5 + x^4y + x^3y^2 + x^2y^3 + xy^4 + y^5} - \dfrac{x^2 - xy + y^2}{x^6 - y^6}$

c) Wir betrachten ein freies Monoid $M = (\Sigma, \circ)$, wobei die Monoid-Elemente die Wörter über dem Alphabet Σ sind, das neutrale Element ϵ das leere Wort über dem Alphabet Σ ist und $\circ$ eine assoziative Abbildung von Paaren auf Wörtern auf ein Wort ist mit $a \circ b = ab, a, b \in \Sigma^*$. Für ein $w \in M$ sei $(w)^n, n \in \mathbb{N}_+$ definiert durch $(w)^1 = w$ und $(w)^n = w(w)^{n-1}$. überlegen Sie sich, wie hier Normalformen minimaler Länge von Worten aus M aussehen, d.h. es sollen Terme mit möglichst wenige Symbolen gefunden werden, die dasselbe Wort darstellen. Können Sie einen effizienten Algorithmus angeben, der diese Normalformen berechnet?

Exercise 4: [Potenzen]

Sei $n \in \mathbb{N}^+$. Eine Additionskette für n ist eine endliche Folge positiver ganzer Zahlen $a_0, \ldots, a_r$ mit $a_0 = 1$, $a_r = n$, und für alle $i = 1, \ldots, r$ gibt es j und k mit $k \leq j < i$, so dass $a_i = a_j + a_k$; r bezeichnen wir als die Länge der Additionskette. Es bezeichne weiter $l(n)$ die minimale Länge einer Additionskette für n.

a) Welcher Zusammenhang besteht zwischen $l(n)$ und der Berechnung von x^n? Ist die Anzahl der Multiplikationen der binären Potenzierungsmethode zur Berechnung von x^n immer minimal?

b) Sei $s(n)$ die Summe der Bits in der Binärdarstellung von n (also die Anzahl der Einsen). Zeigen Sie: $l(n) \leq \lfloor \log_2 n \rfloor + s(n) - 1$ und $l(mn) \leq l(m) + l(n)$.

c) Seien $a > b \geq 0$ ganze Zahlen. Zeigen Sie, dass $l(2^a) = a$ und $l(2^a + 2^b) = a + 1$.

Exercise 5: [Fibonacci]

a) Wir wollen die Fibonacci-Zahlen F_k modulo $n \in \mathbb{N}$ berechnen. Geben Sie eine asymptotische obere Schranke für die Anzahl der Bitoperationen eines naiven Verfahrens zur Berechnung von F_k modulo n im schlechtesten Fall an.

Ein anderes Verfahren benutzt die Tatsache, dass F_{k+1} sich wie folgt gewinnen lässt:

$$\begin{pmatrix} F_{k+1} & F_k \\ F_k & F_{k-1} \end{pmatrix} = \begin{pmatrix} F_k & F_{k-1} \\ F_{k-1} & F_{k-2} \end{pmatrix} \cdot X,$$

wobei X von Ihnen anzugeben ist. Konkretisieren Sie nun dieses Verfahren zur Berechnung von F_k modulo n und geben Sie ebenfalls eine asymptotische obere Schranke für die Anzahl der Bitoperationen im schlechtesten Fall an.

b) Zeigen Sie, wie man $1 + x + \cdots + x^{n-1} \bmod m$ für $x, n, m \in \mathbb{N}^+$ mit $O(\lambda(n)\lambda(m)^2)$ Bitoperationen im schlechtesten Fall berechnet. Beachten Sie besonders den Fall, dass m keine Primzahl ist.