Stellenwertsysteme

1 Einleitung

Es geht um die Problematik der Zahldarstellung in verschiedenen Stellenwertsystemen. Wie bekannt ist, arbeiten die Computer mit dem Binärsystem. Daneben findet noch in der Technik das Hexadezimalsystem und auch das Oktalsystem Anwendung. Somit ist die Umwandlung der einzelnen Stellenwertsysteme ein Problem der Informatik. Mathematisch stellt sich die Frage, ob eine Zahldarstellung in jedem Stellenwertsystem möglich ist und ob diese Darstellung eindeutig ist? Welche Algorithmen zur Umrechnung gibt es?

2 Das Horner-Schema

Aufgabe: Wandle die Binärzahl $z = 10111_2$ effizient in das Dezimalsystem um.

Lösung: Allgemein ist die Bedeutung der einzelnen Ziffern $a_i \in \mathbb{B} = \{0, 1\}$. $\mathbb{B}$ ist die Zahlenmenge der Binärziffern.

Hinweis: Man kann zeigen (Zusatzübung für Interessierte!), dass $\mathbb{B}$ mit den Verknüpfungen $\wedge$, welches dem $\cdot$ entspricht und $\oplus$, welches dem $+ \, mod \, 2$ entspricht, ein Körper bildet. Da die Anzahl der Ziffern in $\mathbb{B}$ (Mächtigkeit) 2 beträgt, ist $\mathbb{B}$ sogar ein endlicher Körper. Man nennt diesen endlichen Körper $\mathbb{F}_2 = GF(2)$, den Galoiskörper.

Die Elemente auf den einzelnen Stellen nennt man auch die Ziffern oder auch Zeichen aus dem Alphabet $\mathcal{A}$ sind. Die Kombination vieler Zeichen nennt man *Wort*.

Wir notieren die Binärdarstellung der Zahl z als Summe von Zweierpotenzen entsprechend der allgemeinen Darstellung $z = \sum_{i=0}^{n-1} a_i \cdot 2^i$; n - Anzahl der Binärstellen.

$$
\begin{aligned}
10111_2 &= \underbrace{1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1}\, + 1 \cdot 2^0 = 23_{10} \quad | \, 2 \text{ ausklammern} \\
&= \underbrace{(1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1}\, + 1) \cdot 2 + 1 = 23_{10} \\
&= \underbrace{((1 \cdot 2^2 + 0 \cdot 2^1}\, + 1) \cdot 2 + 1) \cdot 2 + 1 = 23_{10} \\
&= \underbrace{(((1 \cdot 2^1}\, + 0) \cdot 2 + 1) \cdot 2 + 1) \cdot 2 + 1 = 23_{10} \\
&= ((((1) \cdot 2 + 0) \cdot 2 + 1) \cdot 2 + 1) \cdot 2 + 1) \cdot 2 + 1 = 23_{10}
\end{aligned}
$$

Somit beschränkt sich die Umwandlung einer Zahl in einem beliebigen Stellenwertsystem zur Basis b (b-adisches System) auf die Berechnung des Wertes des Polynoms $p(b) = \sum_{i=0}^{n-1} a_i \cdot b^i$; b - Basis.

Jetzt betrachten wir die Komplexität der Berechnung von $p(b)$. Die Analyse der Anzahl der Operationen lässt erkennen, dass für die Berechnung von $p(b) = \sum_{i=0}^{n-1} a_i \cdot b^i$; b - Basis für die Multiplikation $\mathcal{O}(n^2)$ und für die Addition $\mathcal{O}(n)$ gilt. Eine erste Verbesserung ergibt sich schon bei Speicherung der berechneten Potenzen, womit die Zahl der Multiplikationen für die Potenzen $n-1$ beträgt (dynamisches Programmieren). Hinzu kommen n Multiplikationen der Potenzen mit ihren Koeffizienten. Macht insgesamt $2n-1$.
Beim Horner-Schema reduziert sich jedoch die Anzahl der Multiplikationen, so dass hier $\mathcal{O}(n)$ gilt. (Hinweis: Um ein Berechnung der Komplexität durchzuführen, bestimme man die Anzahl der Operationen einer konkreten Binärdarstellung durch auszählen. Dann verallgemeinere man den Ausdruck in Abhängigkeit der Stellenzahl n.) Die Komplexität der Addition bleibt unverändert bei $\mathcal{O}(n)$.

3 Divison mit Rest

Bei der Darstellung des Polynoms im Hornerschema sieht man zudem, dass bei ganzzahliger Division durch 2 jeweils ein Restpolynom mit Grad $n-1$ und als Rest

eben eine 0 oder 1 übrigbleibt.

$$\begin{aligned} p(b) &= 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 \\ &= (1 \cdot 2^3 + 0 \cdot 2^2 + \cdot 2^1 + 1) \cdot 2 + 1 \\ \Rightarrow \frac{p(b)}{2} &= q(b), \quad \text{Rest 1, mit grad } q(b) = \text{grad } p(b) - 1 \end{aligned}$$

Es gibt für die Berechnung der Division mit Rest nach dem Horner-Schema eine Übersicht unter `https://de.wikipedia.org/wiki/Horner-Schema#Polynomdivision`

Aufgabe:

1. Wandle die Zahl 1433_5 mit Hilfe des Horner-Schemas in das Dezimalsystem um!
2. Entwickle ein Programm `horner` in Haskell (Python, C#), welches dir die Berechnung des Hornerschemas abnimmt.