4. Übungsblatt zu Mathematik für Informatiker I, WS 2003/04

JOACHIM VON ZUR GATHEN, OLAF MÜLLER, MICHAEL NÜSKEN

Abgabe bis Freitag, 14. November 2003, 11^{11}
in den jeweils richtigen grünen oder roten Kasten auf dem D1-Flur.

Einige Aufgaben werden von den Ziffern Deiner Matrikelnummer abhängen. Dazu bezeichne die Ziffern Deiner Matrikelnummer mit m_6, m_5, m_4, m_3, m_2, m_1, m_0, sodass also $\sum_{i=0}^{6} m_j 10^j$ Deine Matrikelnummer ist. Trage Deine Matrikelnummer hier ein, um ganz einfach Deine persönlichen Ziffern abzulesen.

m_6	m_5	m_4	m_3	m_2	m_1	m_0

Aufgabe 4.1 (Logik). (2 Punkte)

Sei a eine (unbekannte) reelle Zahl. Bestimme den Wahrheitswert:

(i) $\forall x \in \mathbb{R}\colon\ \mathrm{e}^{m_1+ax} - (x - m_4)^2 > 0 \iff \forall y \in \mathbb{R}\colon\ \mathrm{e}^{m_1+ay} - (y - m_4)^2 > 0.$

(ii) $\forall x \in \mathbb{R}\colon\ \mathrm{e}^{a(x-m_2)} - (x - am_6)^2 > 0 \iff \forall y \in \mathbb{R}\colon\ \mathrm{e}^{a(y-m_2)} - (y - am_6)^2 \leq 0.$

(iii) $\exists x \in \mathbb{R}\colon\ \mathrm{e}^{a \sin x} - (m_0 x - a)^2 < 0 \iff \exists y \in \mathbb{R}\colon\ \mathrm{e}^{a \sin y} - (m_0 y - a)^2 < 0.$

(iv) $\exists x \in \mathbb{R}\colon\ \mathrm{e}^{a \sin x} - (m_0 x - a)^2 < 0 \iff \exists y \in \mathbb{R}\colon\ \mathrm{e}^{a \sin x} - (m_0 x - a)^2 < 0.$

Aufgabe 4.2 (Binäraddierer). (12 Punkte)

Ziel dieser Aufgabe ist ein Algorithmus für die binäre Addition. Die binäre Addition läuft genauso ab, wie eine schriftliche Addition in der Schule nur eben zur Basis 2 anstatt zur Basis 10. Man beginnt mit den beiden Bits ganz rechts und addiert diese. Das niederwertige Bit des Ergebnisses schreiben wir auf, das höherwertige nehmen wir als Übertrag mit in die nächste Spalte (zur Linken). Dort müssen dann drei Bits addiert werden. Wieder wird das niederwertige Bit der Summe als Ergebnis aufgeschrieben und das höherwertige als Übertrag …

(i) Für $k = 1$ erhalten wir einen Halbaddierer, der aus zwei einzelnen Bits a_0 und b_0 zwei Bits c_1 und c_0 berechnet mit $c_1 \cdot 2 + c_0 = a_0 + b_0$. Entwirf einen Algorithmus, der diese zwei Bits berechnet. Verwende dazu nur die Operationen $\wedge$, $\vee$, $\neg$ und $\oplus$.

(ii) Entwirf einen Algorithmus, der drei Bits a_0, b_0 und u addiert und zwei Bits c_1 und c_0 berechnet mit $c_1 \cdot 2 + c_0 = a_0 + b_0 + u$. Verwende dazu nur die Operationen $\wedge$, $\vee$, $\neg$ und $\oplus$ und Halbaddierer. (Dies bzw. einen entsprechenden Schaltkreis nennt man dann einen Volladdierer.)

(iii) Entwirf einen vollwertigen Algorithmus zur Addition zweier Binärzahlen:

Algorithmus. Binäre Addition.

Eingabe: Eine natürliche Zahl k und Listen $a = (a_{k-1}, \ldots, a_0)$, $b = (b_{k-1}, \ldots, b_0)$ mit $a_i, b_i \in \{0, 1\}$.

Ausgabe: Eine Liste $c = (c_k, c_{k-1}, \ldots, c_0)$ mit $c_i \in \{0, 1\}$ und $\sum_{0 \le i \le k} c_i 2^i = \sum_{0 \le i \le k-1} a_i 2^i + \sum_{0 \le i \le k-1} b_i 2^i$.

1. …
2. For $i = 0$ to $k - 1$ do 3–6
3.   …

⋮   …

6.   …
7. …

⋮ …

10. Return $c = (c_k, c_{k-1}, \ldots, c_0)$.

Verwende dazu nur die Operationen $\wedge$, $\vee$, $\neg$ und $\oplus$ und Volladdierer. In einer Schleife soll ein Zwischenergebnis $u 2^{i+1} + \sum_{0 \le j \le i} c_j 2^j$ berechnet werden, dass die Summe der unteren k Bits von a und b darstellt; formuliere die genaue Bestimmungsgleichung.

(iv) Zeige, dass Dein Algorithmus korrekt arbeitet. Verwende dabei die Bestimmungsgleichung als Schleifeninvariante.

(v) Bestimme die Laufzeit Deines Algorithmus. [Es soll die Anzahl der eigentlichen logischen Operationen gezählt werden, die Schleifenkontrollen vernachlässigen wir.]

(vi) Zeichne einen entsprechenden Schaltkreis für $k = 3$.

Aufgabe 4.3 (Quantoren). (2 Punkte)

Sei $\varphi(x)$ ein logischer Ausdruck, der von x abhängt. Drücke die Aussage „Es gibt genau ein x mit der Eigenschaft $\varphi(x)$." durch den Allquantor $\forall$ und die logischen Operationen $\neg$, $\wedge$, $\vee$, $\Rightarrow$ sowie das Gleichheitszeichen $=$ aus.

Aufgabe 4.4 (Logarithmen). (2 Punkte)

(i) Stelle für $a > 0$ die Lösung y der Gleichung $7^a = 2^y$ durch Logarithmen dar.

(ii) Drücke den Logarithmus von x zur Basis 7 durch Logarithmen zur Basis 2 aus.

Aufgabe 4.5 (Komplexe Zahlen). (3 Punkte)

(i) Berechne $(m_3 + 1 + im_1)(m_6 - im_1)$.

(ii) Berechne $(m_3 + 1 + im_1)/(m_6 + im_1)$.

(iii) Berechne den Betrag von $(m_5 + 2) - 3i(m_1 + 1)$.

Aufgabe 4.6 (Stellenwertsysteme). (5 Punkte)

Besonders in der Informatik sind Stellensysteme zu anderen Basen als 10 gebräuchlich und sehr wichtig. Für die Basis 16 verwenden wir die Buchstaben $a, \ldots, f$ für die hexadezimalen Ziffern(!) $10, \ldots, 15$. Somit ist die hexadezimale Zahl $(ff)_{16}$ die Dezimalzahl 255, nämlich $15 \cdot 16 + 15$.

(i) Stelle Deine Matrikelnummer dezimal, hexadezimal, oktal, 3-adisch und binär dar.

(ii) Berechne (schriftlich und ohne Taschenrechner) hexadezimal $(92a782)_{16} + (b7a10f)_{16}$.

(iii) Berechne (schriftlich und ohne Taschenrechner) oktal $(626053)_8 \cdot (26745)_8$.

(iv) Wandle die Binärzahl $(1001\,0110\,1001\,0110\,0110\,1001)_2$ ins Dezimalsystem um.

(v) Wandle die Binärzahl $(1001\,0110, 1001\,0110\,0110\,1001)_2$ ins Dezimalsystem um. (Achtung, da ist ein Komma drin!)

4. Übungsblatt zu Mathematik für Informatiker I, WS 2003/04, Mündlicher Teil

JOACHIM VON ZUR GATHEN, OLAF MÜLLER, MICHAEL NÜSKEN

Mündliche Aufgabe 4.7 (Quantoren).

Drücke die Aussage „Es gibt höchstens zwei x mit der Eigenschaft $\varphi(x)$." durch den Allquantor $\forall$ und die logischen Operationen $\neg$, $\wedge$, $\vee$, $\Rightarrow$ sowie das Gleichheitszeichen $=$ aus.

Mündliche Aufgabe 4.8 (Komplexe Zahlen).

(i) Berechne $(m_2 + i(1 + m_5))(m_3 - im_6)$.

(ii) Berechne $(m_2 + i(1 + m_5))/(m_3 + im_6)$.

(iii) Berechne den Betrag von $4m_2 + 2im_5$.

Mündliche Aufgabe 4.9 (Stellenwertsysteme).

Besonders in der Informatik sind Stellensysteme zu anderen Basen als 10 gebräuchlich und sehr wichtig. Für die Basis 16 verwenden wir die Buchstaben $a, \ldots, f$ für die hexadezimalen Ziffern(!) $10, \ldots, 15$. Somit ist die hexadezimale Zahl $(ff)_{16}$ die Dezimalzahl 255, nämlich $15 \cdot 16 + 15$.

(i) Stelle Deine Matrikelnummer hexadezimal dar.

(ii) Berechne (schriftlich und ohne Taschenrechner) hexadezimal $(726382)_{16} + (f0913e)_{16}$.

(iii) Berechne (schriftlich und ohne Taschenrechner) oktal $(2472)_8 \cdot (1206)_8$.

(iv) Wandle die Binärzahl $(1001\,1110\,0101\,0010\,0001\,1111\,0001)_2$ ins Dezimalsystem um.

(v) Wandle die Binärzahl $(1001\,1110\,0101\,0010\,0001, 1111\,0001)_2$ ins Dezimalsystem um. (Achtung, da ist ein Komma drin!)

Mündliche Aufgabe 4.10 (Algorithmus b-adisch).

Ziel dieser Aufgabe ist ein Algorithmus, der eine b-adische Zahl in eine ganze (rechnerintern dargestellte) Zahl umwandelt. Die Eingabe sind also die Basis b und eine Liste $a = (a_{k-1}, \ldots, a_0)$ von Ziffern $a_i \in \{0, 1, \ldots, b-1\}$ und die Ausgabe soll die dadurch dargestellte Zahl x sein.

Variante Aufwärts: Berechne in einer Schleife die durch die rechten i Ziffern dargestellte Zahl.

Variante Abwärts: Berechne in einer Schleife, die durch die linken i Ziffern dargestellte Zahl.

Für jede der beiden Varianten bearbeite die folgenden Schritte:

(i) Entwickle den Algorithmus.

(ii) Finde eine Schleifenvariante und beweise, dass sie gültig ist.

(iii) Schliesse, dass der Algorithmus tatsächlich das gewünschte Ergebnis liefert.

(iv) Bestimme die Laufzeit des Algorithmus.

(v) Bestimme den Speicherbedarf des Algorithmus.

Vergleiche die Laufzeiten und entscheide, welche Variante wohl günstiger ist.