Algorithmische Mathematik I

Wintersemester 2011 / 2012
Prof. Dr. Sven Beuchler
Peter Zaspel

universitätbonn

Übungsblatt 1.

Abgabe am **26.10.2011**.

> Am 4. Juni 1996 startete die ESA eine unbemannte Rakete mit vier Satelliten an Bord von Französisch Guyana aus. 40 Sekunden nach dem Start explodierte die Ariane 5. Verlust ca. 500 Millionen Dollar für Rakete und Satelliten. Entwicklungskosten ca. 7 Milliarden Dollar.
> Ursache für den Absturz: Der Bordcomputer stürzte 36.7 Sek. nach dem Start ab als er versuchte, den Wert der horizont. Geschwindigkeit von 64 Bit Gleitkommadarstellung in 16 Bit signed Integer umzuwandeln: -+ b1 b2 ... b15. Die entsprechende Zahl war grösser als $2^{15} = 32768$ und erzeugte einen Overflow. Das Lenksystem brach zusammen und gab die Kontrolle an eine zweite, identische Einheit ab. Selbstzerstörung wurde ausgelöst, da die Triebwerke abzubrechen drohten.

Quelle: Kleine BUGs, große GAUs – Softwarefehler und ihre Folgen, Prof. Thomas Huckle, Institut für Informatik, TU München: http://www5.in.tum.de/~huckle/bugs.html

Aufgabe 1. (Umrechnung Zahlendarstellungen)

Aus der Vorlesung sind bereits Binärzahlen, also B-adische Zahlen für $B = 2$ bekannt. Daneben sind in der Informatik und der numerischen Mathematik auch B-adische Zahlen für die Basis 8 und 16 sehr gebräuchlich. B-adische Zahlen zur Basis 8 werden Oktalzahlen genannt. In der Zifferndarstellung werden also stets nur die Ziffern $0 - 7$ verwendet. Hexadezimalzahlen sind Zahlen zur Basis 16. Bei diesen werden neben den gebräuchlichen Ziffern $0 - 9$ für die gleichlautenden Zahlenwerte auch die Buchstaben A bis F für die Zahlenwerte 10 bis 15 verwendet.

a) Schreiben Sie die Binärzahl 101010 als Dezimalzahl.

b) Schreiben Sie die Hexadezimalzahl $1A8$ als Dezimalzahl.

c) Schreiben Sie die Dezimalzahl 2210 als Oktalzahl.

d) Seien z_1 und z_2 zwei natürliche Zahlen mit identischer Ziffernfolge $d_{n-1}d_{n-2}\dots d_0$ bezüglich unterschiedlicher Basen b_1 und b_2, also

$$z_1 = (d_{n-1}d_{n-2}\dots d_0)_{b_1} \quad \text{und} \quad z_2 = (d_{n-1}d_{n-2}\dots d_0)_{b_2}.$$

Welche der folgenden Aussagen sind wahr?

1) Falls $b_1 > b_2$, so ist $z_1 > z_2$.
2) Falls $z_1 > z_2$, so ist $b_1 > b_2$.

(10 Punkte)

Aufgabe 2. (Gleitkommazahlen)

Wir statten einen Rechner mit einer Zwei-Byte (16 Bit) Gleitkommaarithmetik

$$\pm(1.m_1\, m_2, \dots)_2 \cdot 2^e$$

aus. Die Zahldarstellung hat ein Vorzeichenbit, ist normalisiert und die 1 vor dem Komma wird nicht gespeichert (hidden Bit). Für die Zahl 0 wird das Bitmuster $000\ldots0$ speziell verwendet. Der Exponent soll im Bereich $-7 \leq e \leq 7$ möglich sein und wird – wie in der Vorlesung – in Exzess- (Bias-) Darstellung gespeichert[1].Außer der Spezialbehandlung für die 0 soll nur noch die spezielle Exponentenbitfolge $\tilde{e} = 0$ zur Kennzeichnung von „Underflow“ reserviert werden — sie steht also nicht für die Zahldarstellung zur Verfügung.

a) Welche Darstellung haben die Zahlen 13, 42.125 und 0.8? Für den Fall, daß eine Zahl nicht exakt darstellbar ist, werden die überzähligen Stellen einfach abgeschnitten.

b) Ermitteln Sie die zu der Bitfolge

$$1\,|\,01101011100\,|\,1101$$

gehörige Dezimalzahl.

c) Wie viele Zahlen können in diesem Gleitkomma–Format dargestellt werden? Die Bitkombinationen für Underflow seien zu vernachlässigen.

d) Geben Sie die größte darstellbare Zahl $z_{\max}$, die kleinste darstellbare Zahl $z_{\min}$ sowie die betragsmäßig kleinste Zahl $\bar{z} \neq 0$ an. Geben Sie jeweils auch die zugehörigen Bitmuster an.

(10 Punkte)

Programmieraufgabe 1. (Polynom-Auswertung)

Es sei a_i $i = 0, \ldots, n$ und $x \in \mathbb{R}$ reelle Zahlen sowie

$$p(x) = \sum_{i=0}^{n} a_i x^i$$

ein sog. Polynom n-ten Grades. Man bestimme $p(x)$ durch direktes Auswerten der Summe, d.h. bestimme `p = POLYNOM_WERT(a,n,x)`. Dabei ist a ein *double*-Array der Länge $n + 1$.

Schließlich soll die Polynomauswertung anhand des Beispiels

$$p(x) = 3x^5 - 4x^4 + 3x^2 - x + 9, \quad x = -0.625$$

ausprobiert werden.

(4 Punkte)

Programmieraufgabe 2. (Lineare Algebra - Operationen)

Es sollen typische Operationen auf *double*-Arrays bzw. Vektoren der Länge n von reelen Zahlen implementiert werden. Es seien Vektoren $x, y, z \in \mathbb{R}^n$ der Form $x = (x_1, x_2, \ldots, x_n)^t = (x_i)_{i=1}^n$, sowie der *double*-Werte / die reelle Zahl $\alpha \in \mathbb{R}$ gegeben. Sofern nicht anders angegeben, wird aus den beiden Eingabe-Arrays x und y ein Array z berechnet. Implementieren Sie nun:

a) komponentenweise Addition `VDPLUS(n,x,y,z)`, d.h.

$$z_i = x_i + y_i, \quad i = 1 \ldots n$$

b) komponentenweise Subtraktion `VDMINUS(n,x,y,z)`, d.h.

$$z_i = x_i - y_i, \quad i = 1 \ldots n$$

[1]Erinnerung: Bei der Exzessdarstellung speichert man die Zahl $\tilde{e} := e + |e_{\min}| + 1$.

c) komponentenweise Multiplikation `VDMULT_VEKTOR(n,x,y,z)`, d.h.

$$z_i = x_i y_i, \quad i = 1 \dots n$$

d) Skalarprodukt `sigma = DSCAPR(n,x,y)` mit dem Ergebnis $\sigma \in \mathbb{R}$,

$$< x, y >:= \sigma = \sum_{i=1}^{n} x_i y_i$$

e) Multiplikation des Arrays/Vektors x mit Zahl $\alpha \in \mathbb{R}$ `VDMULT_ZAHL(n,x,alpha,z)`, d.h.

$$z = \alpha x \quad \text{mit } z_i = \alpha x_i.$$

f) kombinierte Addition `VDAXPY(n,x,alpha,y)`, d.h.

$$z = y + \alpha x \quad \text{mit } z_i = \alpha x_i + y_i$$

Man speichere dabei das Ergebnis z auf dem Wert von y, d.h. $y := y + \alpha x$.

g) komponentenweises Maximum `VDMAX(n,x,y,z)`, d.h.

$$z_i = \max\{x_i, y_i\}.$$

h) Maximum von n Zahlen `maximum = MAX(n,x)`:

$$maximum = \max_{i=1\dots n} x_i,$$

Man teste alle implementierten Funktionen für $n = 4$, $x = (1, 8, -2, 7)^\top$, $y = (4, 1, 2, 5)^\top$ und $\alpha = -1.25$.

(8 Punkte)

Programmieraufgabe 3. (Euklidscher Algorithmus)

Man bestimme mit dem Euklidschen Algorithmus den größten gemeinsamen Teiler $t = ggT(m, n)$ zweier ganzer Zahlen m und n. Dies implementiere man

1. iterativ, d.h. das Unterprogramm `t = EUKILD_ITERATIV(m,n)`
2. rekursiv. d.h. das Unterprogramm `t = EUKILD_REKURSIV(m,n)`

und bestimme $ggT(7, 11)$, $ggT(12, 16)$ und $ggT(4536, -5472)$.

(8 Punkte)

Die Abgabe der Programmieraufgaben erfolgt in den CIP-Pools in der Woche vom 31.10. bis 04.11.2011. Die Listen für die Anmeldung zu den Abgabe- Terminen hängt in der Woche vom 24.10. bis 28.10.2011 aus. Bitte beachten Sie, dass Sie lauffähige HRZ-Login-Accounts benötigen.