Übungen zu Informatik I
Wintersemester 03/04

Übungsleiter: Dipl.-Inform. Tom Gelhausen

Vorlesung? Übung? Tutorium? Rechnerübung? Wohin?

- Vorlesung
  - Betreuer: Prof. Goos
  - Zeit: Mo. und Mi. 14.00
  - Ort: Forum
- Übung
  - Betreuer: Tom Gelhausen
  - Zeit: Fr. 14.00
  - Ort: Forum
- Tutorium
  - Betreuer: variabel
  - Zeit: variabel
  - Ort: Info-Bau, AVG, (RZ)
- Rechnerübung
  - Betreuer: der gleiche wie beim Tutorium
  - Zeit: variabel
  - Ort: RZ -110

Organisatorisches

- Tutorieneinteilung
  - Ergebnis unter: http://www.ira.uka.de/~thgries/wis/Einteilung1.html
  - Tutorien starten am Montag 20.10. (ggf. mit Rechnerübung)
  - Tutoren unter http://www.infoeins.de/tutorien.php
- Programme/CD
  - „Raub“-kopieren der offiziellen CD hat letztes Jahr hervorragend funktioniert
  - Hohes finanzielles Risiko für die Produktion einer CD
  - Daher dieses Jahr nur unter: http://www.infoeins.de/software.php
- Schreibmaschinenkurs
  - http://www.infoeins.de/software.php#Schreibmaschinenkurs
- Login auf der Webseite
  - Punktestand
  - Kontaktdaten für Tutor

Anmeldung zur Klausur

- Anmeldung erfordert Zulassungsbescheinigung („Blauer Zettel“ auch in grün oder rot …)
  - Ausnahme Physiker
  - und Schülerstudenten

- Für Physiker gilt also:
  - Info I ist eine Scheinklausur
  - Info I-Schein ist Voraussetzung für die Teilnahme an der Info II Klausur

- Wichtig für alle: die Anmeldung muss rechtzeitig (bis Fr, 06.02.2004, 18:00 Uhr, Einwurfkasten im AVG Süd, 2. OG) erfolgen!

Probeklausur

- Termin: Fr, 19.12.2003 14:00 Uhr
- Sinn
  - Klausurablauf proben
  - Klausuratmosphäre kennen lernen
  - Klausurvorbereitung üben
  - Klausurumfang und -schwierigkeitsgrad begreifen lernen

Übungsaufgaben

- Klausurbonus:
  - 0,3 bei Bestehen der Klausur
  - Wenn mindestens 2/3 (also $66{,}\overline{6}$%)
    - der Theoriepunkte,
    - der Praxispunkte,
    - im Zeitraum bis Weihnachten
    - <u>und</u> im Zeitraum nach Weihnachten erreicht wurden.
- Weitere Regeln:
  - Keine Teamabgabe (z.B. 1 Blatt, 5 Namen)
  - Keine getippten Theorieaufgaben
  - Übungsblattkopf (s. nächste Folie) ist verbindlich!
  - Abgabezeitplan:

Ausgabe Übungsblatt X ↓ (F, 1. Woche)

Späteste Abgabe Übungsblatt X (Theorie) ↓ (F, 2. Woche)

Späteste Abgabe Übungsblatt X (Praxis) ↓ (F, 3. Woche)

M D M D F S S | M D M D F S S | M D M D F S S

Übungsblattkopf

[Tacker]

Matrikelnummer

Übungsblatt X

Tut.nr.

Name, Vorname

Informatik I

Bla blabla blubber rababer gelaber. Blablabla blub bla. Blablabla! Bluber blub blubbla. Blabbberblubber laber. Rababer blaber blub laber rababer. Bla blub Rababera baberaba bla. Blubber bla rababer gelaber. Blubber blubber blubber blubber blubber blubber. Bla blabla blubber rababer gelaber. Blablabla blub bla. Blablabla! Bluber blub blubbla. Blabbberblubber blubber blubber blubber blubber blubber. Bla blabla blubber rababer gelaber. Blablabla blub bla. Blablabla! Bluber blub blubbla. Blabbberblubber laber. Rababer blaber blub laber rababer. Bla blub Rababera baberaba bla. Blubber bla! Bluber blub blubbla. Blabbberblubber laber. Rababer blaber blub laber rababer. Bla blub Rababera baberaba bla. Blubber bla rababer gelaber. Blubber blubber blubber blubber blubber blubber….

Fachschaft: Fachbereichinformation

- Mittwoch, 22.10.2003
- 17.30 Uhr
- Hörsaal –101, Info Bau
- Für alle, die die Fachbereichinfo in der O-Phase verpasst haben
- Inhaltlich identisch zu der aus der O-Phase
- Folien ab Ende nächster Woche im Netz
- Für die Mathematiker ist eine solche Veranstaltung nicht geplant.

Anmerkungen zum 0. Übungsblatt

- Aufgabe 1.2
  - eMail-Adressen von Schulen/T-Online/… gelten auch nicht!
- Aufgabe 2.4
  - Packen Sie die Datei wieder genau an die Stelle im ZIP-Archiv, an der Sie sie gefunden haben
  - Schicken Sie die Aufgabe an Ihren Tutor.
- Bewertung Aufgabe 1.2 bis 2.4
  - Wenn Sie eine zufrieden stellende Lösung (ZIP-Datei) einsenden, gehen wir davon aus, dass sie die Aufgaben 1.2 bis 2.4 gelöst haben und jetzt mit Rechnern, Dateien, Verzeichnissen, Text-Editoren, Web und eMail umgehen können.
- Aufgabe 2.5
  - Tackern Sie das Ergebnis an Ihre Lösungen von Übungsblatt 1.
- Aufgabe 2.6
  - Tackern Sie Ihre Ausführungen an die Lösungen von Übungsblatt 1.

1.1 Codierung ganzer Zahlen

Binär codierte Dezimalzahlen: (BCD: *binary coded decimal*)
Codieren einzelner Dezimalziffern
☛ Rechnen im Dezimalsystem

Beispiel: Darstellung von z=295: z = OOLO LOOL OLOL

Binärdarstellung ganzer Zahlen:

$$\boxed{z = \sum_{i=0}^{n-1} z_i \cdot 2^i}$$

in Rechnern und Programmiersprachen ist n fest
Langzahlarithmetik: n unbeschränkt
negative Zahlen?

☛ Rechnen im Binärsystem

Darstellung im Oktalsystem: Basis 8

oder im Sedezimalsystem: Basis 16, Darstellung der Zahlen 10-15:
A≙10, B≙11, C≙12, D≙13, E≙14, F≙15

Einerkomplement: invertiert jedes Bit:

$$-z = 2^n - 1 - z = \sum_{i=0}^{n-1} (1 - z_i) \cdot 2^i$$

$$+0 = \sum_{i=0}^{n-1} 0 \cdot 2^i$$

$$-0 = \sum_{i=0}^{n-1} 1 \cdot 2^i$$

Zweierkomplement:

$$-z = 2^n - z = 1 + \sum_{i=0}^{n-1} (1 - z_i) \cdot 2^i$$

darstellbarer Zahlenbereich mit n+1 Bits: $-2^n \le z \le 2^n - 1$
n und daher darstellbarer Zahlenbereich beliebig

Vorzeichen und Betrag:

$$-z = (L, |z|)$$

Beispiel:

Binärzahl:	0100110	(dezimal : 38) —
Einerkomplement:	1011001	
Zweierkomplement:	1011010	(dezimal : -38)
Vorzeichen und Betrag:	1100110	(dezimal : -38)

Beispiel:

$$z_2 = 1010011010_2 = \sum_{i=0}^{9} z_i 2^i = 1\cdot 2^9 + 1\cdot 2^7 + 1\cdot 2^4 + 1\cdot 2^3 + 1\cdot 2^1 = 666_{10}$$

Konvertierung ganzer Zahlen:

Rechnen im Quellsystem: (Zielbasis B_Z und Zahl Z im Quellsystem)

Beispiel: 1020_{10} (dezimal) im Oktalsystem darstellen:

1020	: 8 = 127	Rest 4
127	: 8 = 15	Rest 7
15	: 8 = 1	Rest 7
1	: 8 = 0	Rest 1

1774

☛ $1020_{10} = 1774_8$

Konvertierung ganzer Zahlen:

Rechnen im Zielsystem: (Zielsystem ist Dezimalsystem)

Beispiel: 1100011_2 dezimal darstellen:

Quellziffer	1	1	0	0	0	1	1
zu addieren	-	2	6	12	24	48	98
Quellbasis	· 2	· 2	· 2	· 2	· 2	· 2	
Stellensumme	1	3	6	12	24	49	99

(nach Hornerschema)

Hornerschema: neues Ergebnis := altes Ergebnis · Basis + Ziffer

hier: $\underbrace{(((((1 \cdot 2 + 1) \cdot 2 + 0) \cdot 2 + 0) \cdot 2 + 0) \cdot 2 + 1) \cdot 2 + 1}$

die Lösung ist 99_{10}

Konvertierung von Dezimalbrüchen:

- Umwandlung nicht immer exakt möglich
- Rechnen im Quellsystem:

  Beispiel: $0{,}35_{10}$ im Dualsystem darstellen:

Schritt	Operation	Zwischenergebnis	notiere
1	$0{,}35 \cdot 2$	0,70	0
2	$0{,}70 \cdot 2$	1,40	1
3	$0{,}40 \cdot 2$	0,80	0
4	$0{,}80 \cdot 2$	1,60	1
5	$0{,}60 \cdot 2$	1,20	1
6	$0{,}20 \cdot 2$	0,40	0
-	Abbruch	-	-

☛ das Ergebnis ist $0{,}010110\ldots_2$. Es setzt sich aus den notierten Ziffern von oben nach unten zusammen

Konvertierung von Dualbrüchen:

Rechnen im Zielsystem: (Zielsystem ist Dezimalsystem)

Beispiel: $0{,}100101_2$ dezimal darstellen:

In umgekehrter Ziffernreihenfolge ergibt sich:

Quellziffer	1	0	1	0	0	1	0
zu addieren	-	0,5	0,25	0,625	0,3125	0,15625	0,578125
Quellbasis	÷2	÷2	÷2	÷2	÷2	÷2	
Stellensumme	1	0,5	1,25	0,625	0,3125	1,15625	0,578125

☛ Ergebnis: 0,578125

1.1 Rechnen im Dualsystem

Rechenregeln des Dezimalsystems lassen sich prinzipiell unverändert auf Dualsystem übertragen:

Operation	Ergebnis	Übertrag auf nächsthöhere Stelle
0+0	0	0
0+1	1	0
1+0	1	0
1+1	0	1 = Übertragsbit
0-0	0	0
0-1	1	1 = „Borgbit“
1-0	1	0
1-1	0	0
0*0	0	0
0*1	0	0
1*0	0	0
1*1	1	0

Beispiele:

1. Addition

$$1010_2 = 10_{10}$$
$$+1011_2 = +11_{10}$$
(Übertrag)1 01
$$10101_2 = 21_{10}$$

2. Subtraktion

$$1101_2 = 13_{10}$$
$$-1010_2 = -10_{10}$$
(Borgen) 1
$$0011_2 = 3_{10}$$

3. Multiplikation

$$110_2 \times 101_2 = 6_{10} \times 5_{10}$$
$$110_2$$
$$0_2$$
$$110_2$$
$$11110_2 = 30_{10}$$

4.Division

Beispiel: $100001_2 : 110_2 = 0101{,}1_2$ $\qquad 33_{10} : 6_{10} = 5{,}5_{10}$

```
100001 : 110 = 0101,1

 100                          = 4_10
-110               x0         = -0x6_10
 1000                         = 4_10  -> *2+0 = 8_10
-  110              x1        = -1x6_10
 00100                        = 2_10  -> *2+0 = 4_10
-    110             x0       = -0x6_10
     1001                     = 4_10  -> *2+1 = 9_10
-     110             x1      = -1x6_10
     0011,0 ------------> ,   = 3_10  -> *2+0 = 6_10
-       11 0             x1   = -1x6_10
           0                  = 0_10
```

Übung – Binäre Addition

Basis 2	Basis 16	Basis 10
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

```
 00011110
+00001011
    1111
---------
 00101001
```

Basis 2	Basis 16	Basis 10
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

00011111
-00001001

10110
0 1 1 0 1 0
− 0 0 1 1 0 1

→ EK 1 1 0 0 1 0
ZK 1 1 0 0 1 1

0 1 1 0 1 0
+ 1 1 0 0 1 1
1 | 0 0 1 1 0 1

$(-1) + (-1)$

	0001
Ek	1110
Zk	1111

$$
\begin{array}{r|rrrr}
 & 1 & 1 & 1 & 1 \\
+ & 1 & 1 & 1 & 1 \\
{\scriptstyle 1} & {\scriptstyle 1} & {\scriptstyle 1} & & \\
\hline
1 & 1 & 1 & 1 & 0
\end{array}
$$

Übung – Binäre Multiplikation

Basis 2	**Basis ~~8~~** 16	**Basis 10**
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

00011111 * 00001001 =

11111
00000
00000
11111
1 1 1 1

100010111

Übung – Binäre Division

Basis 2	**Basis 8**	**Basis 10**
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

0110001 : 111 =

1.1 Darstellung reeller Zahlen

Festpunktdarstellung: $$z = \sum_{i=-c}^{d-1} z_i \cdot 2^i$$

- komplette Menge der reellen Zahlen kann nicht durch endliche Zeichenreihen über B dargestellt werden
- Anwendung bei Prozeßsteuerungen zur Geschwindigkeitssteigerung

Gleitpunktdarstellung: b, e ganzzahlig, b ≥ 2, 0 ≤ m < b. $$z = m \cdot b^e$$

- m heißt Mantisse, e heißt Exponent, b heißt Basis
- 3.14159, 0.314159E1, 0.0314159E2 (typisch für Programmiersprachen)
- im Rechner: m, e sind binär codiert, b = 2

Normalisierte Gleitpunktdarstellung: 1 ≤ m < b oder m=0, falls z=0

☛ in Binärdarstellung ist erstes Bit von m gleich L

Gleitpunktdarstellung 1937 von Konrad Zuse erfunden

Für Details siehe: http://www.infoeins.de/literatur.php

1.1 Gleitpunktzahlen nach IEEE 754-1985

1	8	23
S	E	M
Vorzei-chen	Exponent (verschoben)	Mantisse (gekappt)

1	11	52
S	E	M
Vorzei-chen	Exponent (verschoben)	Mantisse (gekappt)

- 1.0 ≙ 0 01111111 00000000000000000000000
  0.5 ≙ 0 01111110 00000000000000000000000

- Rechnen: $z_1 = m_1 \cdot 2^{e_1}$, $z_2 = m_2 \cdot 2^{e_2}$

  $z_1 \pm z_2$ = normalisiere $(m_1 \pm m_2 \cdot 2^{e_2 - e_1}) \cdot 2^{e_1}$

  $z_1 \cdot z_2$ = normalisiere $(m_1 \cdot m_2) \cdot 2^{e_1 + e_2}$

  $z_1 \div z_2$ = normalisiere $(m_1 \div m_2) \cdot 2^{e_1 - e_2}$

Überlauf: zu großer Exponent

Unterlauf: zu kleiner Exponent

Rundungsfehler: Fehler durch Runden der Mantisse

1.1 Beispiel: Gleitpunktmultiplikation

Durchzuführende Schritte:

1. Multiplikation der Mantisse (mehrere Einzelmultiplikationen, Addieren der Zwischenergebnisse)
2. Normierung der Mantisse
3. Angleichen der Exponenten
4. Addition der Exponenten

Beispiel (im Dezimalsystem): $0.792 \cdot 10^5 \ * \ 0.116 \cdot 10^{-3}$

```
0.792 * 0.116                (Schritt 1)
-------------
        792
         792
         4752
-------------
     0.091872 = 0,918 · 10^-1      (Schritt 2. und 3.)
```

$5 + (-3) + (-1) = 1$ (Schritt 4)

$\Rightarrow$ das Ergebnis ist $0.918 \cdot 10^1$

1.1 Entscheidungsgehalt

Entscheidungsgehalt eines Zeichenvorrats mit n Zeichen:

Kleinste Anzahl H von Entscheidungen, mit denen festgestellt werden kann, welches Zeichen vorliegt.

- H = 1 für n = 2, allgemein H ≤ ld(n) (ld: logarithmus dualis)
- Maßeinheit: bits (kleines b!)

Beispiel: Entscheidungsgehalt eines Zeichenvorrats mit 2 Zeichen

☛ H = 1 bit

Shannonsche Informationstheorie: (*Shannon, 1916 - 2001*)

- Ziel: minimaler Übertragungsaufwand für codiertes Wort c(w)
  - Mitübertragung der Codierungsvorschrift w, oder
  - minimiere Erwartungswert der Länge von c(w) zu vorgegebener Menge von Wörtern M
- Information (Entropie, Unsicherheit): $\boxed{H(p) = -\sum_{i=1}^{m} p_i \cdot \mathrm{ld}(p_i)}$ [bit]

  mit Wahrscheinlichkeitsverteilung $p = (p_1, p_2, \ldots, p_m)$

1.1 Huffman Codes

D.A. Huffman, 1925-1999

- Gegeben Zeichenvorrat $\Sigma=\{z_1,\ldots,z_n\}$ und Wahrscheinlichkeiten des Auftretens $p(z_i)=p_i$, wobei $\boxed{\sum_{i=1}^{n} p_i = 1}$
- Gesucht: Codierung der Zeichen mit minimaler Länge c(w) von Wörtern w
- algorithmische Lösung:
  - Ordne Zeichen nach fallender Wahrscheinlichkeit an: $p_1 \geq p_2 \geq \ldots \geq p_n$
  - Fasse die Zeichen z_{n-1}, z_n zu neuem Zeichen z' mit $p' = p_{n-1} + p_n$ zusammen und ordne z' entsprechend p' in die Ordnung ein.
    Ergebnis: Zeichenvorrat mit einem Zeichen weniger.
  - Wiederhole den vorangehenden Schritt, bis nur ein Zeichen mit $p_1=1$ übrig ist.
  - Konstruiere „Codebaum“:

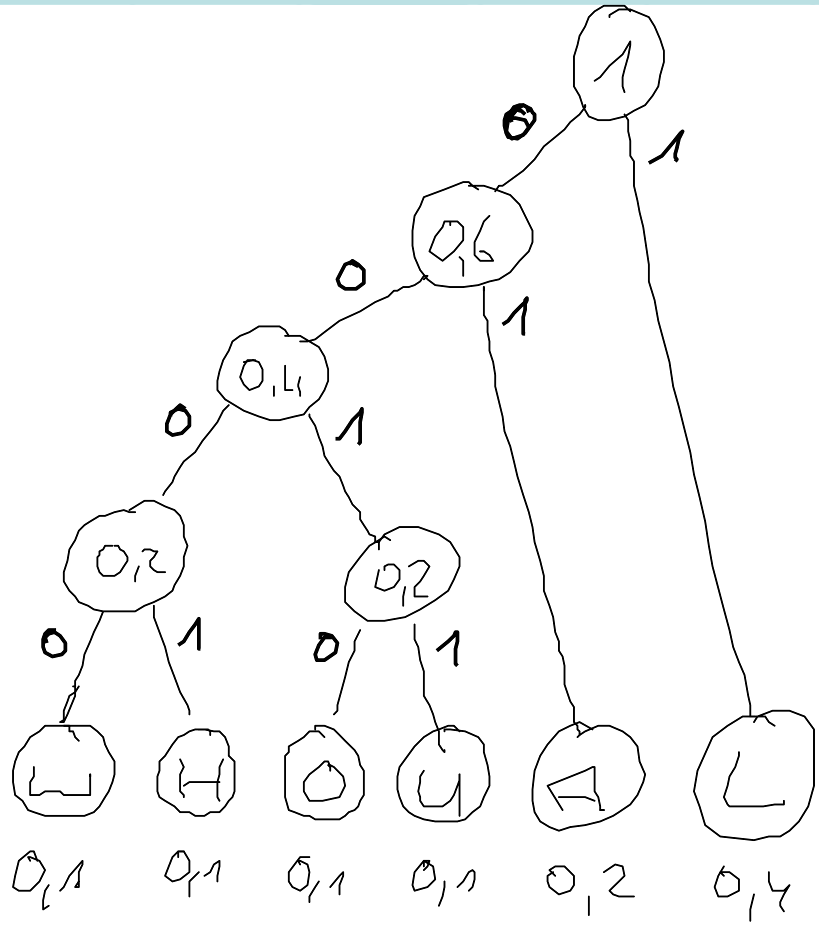
$$\begin{tikzcd} & p' \arrow[dl, "O"'] \arrow[dr, "L"] & \\ p_{n-1} & & p_n \end{tikzcd}$$

  - Repräsentiere jedes Zeichen durch die Folge der O bzw. L von der Baumwurzel zum Zeichen

Übung – Huffmancodes

„HALLO ULLA“

H	1
A	2
L	4
O	1
U	1
␣	1



L	1
A	01
U	0011
O	0010
H	0001
␣	0000

L A L A L A ␣ O L A

101101101000000101101!

Wissenschaftliche Untersuchung dazu: http://angel.elte.hu/wave/

1.1 Huffman Codes

Beispiel: englische Sprache

E	LOO	**S**	OLLO	**U**	OOOLO	**K**	LLOLOOOLL
T	OOL	**H**	OLOL	**G**	OOOOL	**X**	LLOLOOOOL
A	LLLL	**D**	LLOLL	**Y**	OOOOO	**J**	LLOLOOOOO
O	LLLO	**L**	OLLLL	**P**	LLOLOL	**Q**	LLOLOOOLOL
N	LLOO	**F**	OLOOL	**W**	OLLLOL	**Z**	LLOLOOOLOO
R	LOLL	**C**	OLOOO	**B**	OLLLOO		
I	LOLO	**M**	OOOLL	**V**	LLOLOOL		
	3, 4		4, 5		5, 6, 7		9, 10

Präfixcode: Kein Wort des Codes ist Anfang eines anderen Codes

Beispiele: Huffmancode, Morsecode