

Grundlagen der Programmierung

Prof. Dr. Franziska Matthäus
 Prof. Dr. Matthias Kaschube
 Dr. Karsten Tolle
 Mimoun El Masiani (B.Sc.)



Übungsblatt 3

Ausgabe: 06.11.2023
 Abgabe: 13.11.2023
 12:00 Uhr

Zahlensysteme

Hinweis:

- Es sind grundsätzlich Rechenwege anzugeben, es sei denn es findet sich ein expliziter Hinweis, dass dies nicht nötig ist.
- Es dürfen keine Lösungen aus dem Skript, dem Internet oder anderen Quellen abgeschrieben werden. Diese Quellen dürfen nur mit Quellenangaben verwendet werden und es muss ein hinreichend großer Eigenanteil in den Lösungen deutlich zu erkennen sein.
- Digitale Abgaben, die nicht im Format **.pdf** oder **.txt** für Texte oder **.py** für Code erfolgen, werden nicht bewertet. Bei Abgaben mehrerer Dateien müssen diese als **.zip** zusammengefasst werden.
- Achten Sie darauf die Variable **__author__** in allen Quellcode Dateien korrekt zu setzen (am Anfang des Quellcodes):
__author__ = "<Matr-Nr>, <Nachname>"
- Außerdem muss Ihr Name in jeder abgegebenen **.pdf** und **.txt** Datei zu finden sein.
- Abgaben, die vollständig per Hand geschrieben und eingescannt werden, sind **NUR nach Absprache mit dem Tutor erlaubt**.

Σ 10 Punkte

Aufgabe 1 – Zahlenumrechnung Algorithmus

3 Punkte

Beschreiben Sie einen Algorithmus (z.B. als Pseudocode), der eine beliebige natürliche Zahl in eine Binärzahl umwandelt. **Implementieren** Sie den Algorithmus in Python 3.X.

Schreiben Sie den Quellcode auf und die Ergebnisse für 3 Testfälle.

Wichtig: Eine Dokumentation wird nicht verlangt. Versuchen Sie (keine Verpflichtung) den Algorithmus so robust wie möglich zu machen.

Aufgabe 2 – Zahlenumrechnung

3 Punkte

Füllen Sie die untenstehende Tabelle aus. Geben Sie dabei die Berechnung mit an.

Dezimalzahl	Binärzahl	Hexadezimal	Oktal
262			
		2BD	
	1101101111010		
			71

Aufgabe 3 – Rechnen mit Binärzahlen

2 Punkte

Berechnen Sie folgende Ausdrücke. Verwenden Sie für **a) das 2er-Komplement in 8 Bit**. Geben Sie jeweils eine kurze Begründung bzw. den Rechenweg an.

a) $10111_{(2)} - 1001_{(2)}$

b) $49A_{(16)} + B4A3_{(16)}$

Aufgabe 4 – Aussagenlogik

2 Punkte

Überprüfen Sie folgende Aussage mithilfe einer Wahrheitstabelle und geben Sie an, ob die Aussage erfüllbar ist.

a) $(A \Rightarrow B) \Leftrightarrow (\neg A \vee B)$

b) $A \vee (B \wedge C) \Leftrightarrow (A \wedge B) \vee (A \wedge C)$

Hinweis zu den logischen Operatoren: \Rightarrow : Implikation, \Leftrightarrow : Äquivalenz, \neg : nicht, \vee : oder, \wedge : und

Aufgabe 1

zahl = natürliche Zahl zum Umwandeln

binärzahl = leerer string

if zahl = 0:

 sib "0" zurück

for loop: solange zahl > 0

 rest = zahl modulo 2

 binärzahl = rest + binärzahl

 zahl = zahl / 2

gib binärzahl zurück

Aufgabe 2 – Zahlenumrechnung

3 Punkte

Füllen Sie die untenstehende Tabelle aus. Geben Sie dabei die Berechnung mit an.

Dezimalzahl	Binärzahl	Hexadezimal	Oktal
262	100000110	106	406
701	1010111101	2BD	1275
7039	1101101111010	1B7A	15572
87	111001	33	71

Zweierkomplemente: 1011_2 und 0111_2

(leichter) (1er-komplement + 1)

$$\Rightarrow \begin{array}{r} 10111_2 \\ + 0111 \\ \hline 11110_2 \end{array}$$

→ zurückwandeln:

$$11110_2 - 1 = 11101 \quad (1\text{er-Komplement})$$

$$11101 \rightarrow -00010 = -10_2 = \underline{\underline{-2_{10}}}$$

$$b) 49A_{16} + B4A3_{16}$$

$$\downarrow \quad \searrow$$

$$10010011010_2 + \underline{1011010010100011_2}$$

$$\begin{array}{r} 10010011010_2 \\ + 1011010010100011_2 \\ \hline 1011100100111101_2 \\ \hline \end{array}$$

B 3 3 D₁₆

$$\Rightarrow 49A_{16} + B4A3_{16} = \underline{\underline{B93D_{16}}}$$

Es ist der Aufgabenstellung nicht klar zu entnehmen, in welchem Zahlensystem die Rechnung geschehen soll.
Die Aufgabe heißt "Rechnen mit Binärzahlen", daher hier in Binär.

Aufgabe 4 – Aussagenlogik

2 Punkte

Überprüfen Sie folgende Aussage mithilfe einer Wahrheitstabelle und geben Sie an, ob die Aussage erfüllbar ist.

a) $(A \Rightarrow B) \Leftrightarrow (\neg A \vee B)$

b) $A \vee (B \wedge C) \Leftrightarrow (A \wedge B) \vee (A \wedge C)$

Hinweis zu den logischen Operatoren: \Rightarrow : Implikation, \Leftrightarrow : Äquivalenz, \neg : nicht, \vee : oder, \wedge : und

a)

A	B	$A \Rightarrow B$	$\neg A \vee B$	$(A \Rightarrow B) \Leftrightarrow (\neg A \vee B)$
0	0	1	1	1
0	1	1	1	1
1	0	0	0	1
1	1	1	1	1

✓ ist erfüllbar

b)

A	B	C	$B \wedge C$	$A \vee (B \wedge C)$	$(A \wedge B) \vee (A \wedge C)$	$A \vee (B \wedge C) \Leftrightarrow (A \wedge B) \vee (A \wedge C)$
0	0	0	0	0	0	1

0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	1
0	1	1	1	1	0	0	0	0
1	0	0	0	1	0	0	0	0
1	0	1	0	1	0	1	1	1
1	1	0	0	1	1	0	1	1
1	1	1	1	1	1	1	1	1

ist nicht
erfüllbar X

