



## 1. Übungsblatt - Codierung

Digitaltechnik und Rechnersysteme • Wintersemester 2025/2026

Rustam Khavaiashkhov

### 1 Gruppenübung

#### 1.1 ASCII Code

Dekodieren Sie die folgende Nachricht welche im 7-Bit ASCII-Code codiert ist (siehe Code-Tabelle aus Vorlesung):

10010001100101110110011011001101110100000101011111011111110010110110011001000100001

Hello world!

#### 1.2 Codes mit variabler Codelänge

Entwickeln Sie einen Huffman-Code mit variabler Länge für die folgenden Symbole die mit ihrer Wahrscheinlichkeit gegeben sind:

Symbol	Wahrscheinlichkeit	
A	0,45	a 1
B	0,13	b 010
C	0,12	c 011
D	0,16	d 000
E	0,09	e 0011
F	0,05	f 0010

Wie lautet die mittlere Codelänge für die Codierung?

2,24

## 2 Hausübung

### 2.1 Codes mit variabler Codelänge (10 Punkte)

Gegeben ist folgender Code mit variabler Codelänge:

Symbol	Codewort
A	101
B	100
C	01
D	00
E	110
F	111

a) Wie lautet der Inhalt der folgenden Nachricht?

01101111110100101100110 CAFE BABE

b) Ermitteln Sie die Wahrscheinlichkeit für jedes Symbol in dieser konkreten Nachricht.

b)  
 $A = 2/8 = 0.25$   
 $B = 2/8 = 0.25$   
 $C = 1/8 = 0.125$   
 $D = 0$   
 $E = 2/8 = 0.25$   
 $F = 1/8 = 0.125$

c) Geben Sie mit Hilfe der Huffman-Codierung ein Code mit minimaler mittlerer Codelänge für diese Wahrscheinlichkeit an.

d) Wie viel Bit werden durch die neue Codierung eingespart wenn die Nachricht aus Aufgabenteil a) codiert werden soll?  $23 - 18 = 5 \text{ Bits}$

e) Wie lautet die mittlere Codelänge für die alte und neue Codierung? 2.875 und 2.25

c.  
 $A = 01$   
 $B = 00$   
 $C = 111$   
 $D = 10$   
 $E = 110$   
 $F = 110$

### Anhang ASCII-Tabelle

Codewort	Zeichen	Codewort	Zeichen	Codewort	Zeichen	Codewort	Zeichen
0000000	NUL	0100000	SP	1000000	@	1100000	`
0000001	SOH	0100001	!	1000001	A	1100001	a
0000010	STX	0100010	"	1000010	B	1100010	b
0000011	ETX	0100011	#	1000011	C	1100011	c
0000100	EOT	0100100	\$	1000100	D	1100100	d
0000101	ENQ	0100101	%	1000101	E	1100101	e
0000110	ACK	0100110	&	1000110	F	1100110	f
0000111	BEL	0100111	'	1000111	G	1100111	g
0001000	BS	0101000	(	1001000	H	1101000	h
0001001	TAB	0101001	)	1001001	I	1101001	i
0001010	LF	0101010	*	1001010	J	1101010	j
0001011	VT	0101011	+	1001011	K	1101011	k
0001100	FF	0101100	,	1001100	L	1101100	l
0001101	CR	0101101	-	1001101	M	1101101	m
0001110	SO	0101110	.	1001110	N	1101110	n
0001111	SI	0101111	/	1001111	O	1101111	o
0010000	DLE	0110000	0	1010000	P	1110000	p
0010001	DC1	0110001	1	1010001	Q	1110001	q
0010010	DC2	0110010	2	1010010	R	1110010	r
0010011	DC3	0110011	3	1010011	S	1110011	s
0010100	DC4	0110100	4	1010100	T	1110100	t
0010101	NAK	0110101	5	1010101	U	1110101	u
0010110	SYN	0110110	6	1010110	V	1110110	v
0010111	ETB	0110111	7	1010111	W	1110111	w
0011000	CAN	0111000	8	1011000	X	1111000	x
0011001	EM	0111001	9	1011001	Y	1111001	y
0011010	SUB	0111010	:	1011010	Z	1111010	z
0011011	ESC	0111011	;	1011011	[	1111011	{
0011100	FS	0111100	<	1011100	\	1111100	
0011101	GS	0111101	=	1011101	]	1111101	}
0011110	RS	0111110	>	1011110	^	1111110	~
0011111	US	0111111	?	1011111	_	1111111	DEL