Karlsruher Institut für Technologie

Institut für Technik der Informationsverarbeitung

Felix Pistorius

pistorius@kit.edu

Karlsruher Institut für Technologie (**KIT**)

Institut für Technik der Informationsverarbeitung (ITIV)

Digitaltechnik

2. Übung

Shannons Informationstheorie

Definition Der Informationsgehalt eines Zeichens hängt logarithmisch von dessen Auftrittswahrscheinlichkeit ab:

$$H_e = \log_2\left(\frac{1}{p}\right)$$

Definition Mittlere Codewortlänge:

$$\overline{m} = \sum_{i=1}^{n} p(x_i) * m(xi)$$

Definition Entropie der Sendequelle:

$$H = \sum_{i=1}^{n} p(x_i) * He(xi) = \sum_{i=1}^{n} p(x_i) * ld\ \frac{1}{p(xi)}$$

Shannons Informationstheorie

- **Postulat** Es gibt kein Codierungsverfahren mit $\overline{m} < H$
  → *H* ist das **theoretische Minimum** der mittleren Codewortlänge

- Codierungsverfahren, die $\overline{m}$ möglichst dicht an ihren möglichen (Minimal-)Wert **H** heranführen, heißen **optimale Codes**

- Vereinfacht gesagt gilt für optimale Codes:
  - Häufig auftretende Symbole
    → möglichst kurze Codewörter
  - seltene Symbole
    → längere Codewörter

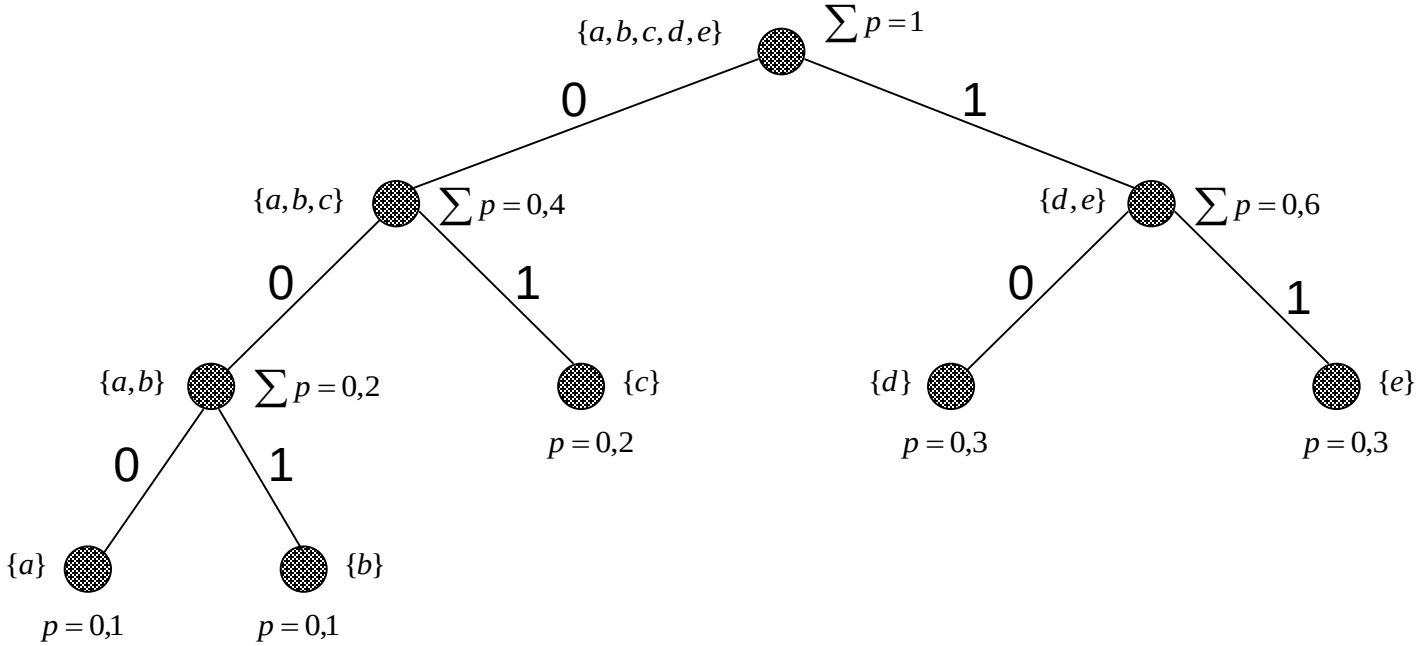
Shannon-Fanø-Code

- Konstruktionsvorschrift:
  - **Zeichenvorrat** nach **aufsteigender Wahrscheinlichkeit** linear sortieren.
  - **Zwei Teilmengen** so konstruieren, dass die **Summen-wahrscheinlichkeiten** der beiden Teilmengen **möglichst** gleich groß sind. D.h. die addierten Auftrittswahrscheinlichkeiten der in beiden Teilmengen enthaltenen Zeichen ergeben möglichst die gleiche Summe.
  - Die **erste Teilmenge** erhält das Codierungszeichen “0“ und die **zweite Teilmenge** eine “1“
  - Mit den jeweils **resultierenden Teilmengen rekursiv** in gleicher Weise fortfahren
    → bis nur noch 1 Zeichen in resultierenden Teilmengen enthalten ist

Shannon-Fanø-Code

- Beispiel:

Zeichen	a	b	c	d	e
Wahrscheinlichkeit	0,1	0,1	0,2	0,3	0,3



Codierung:

Zeichen	a	b	c	d	e
Kodierung	000	001	01	10	11

Karlsruher Institut für Technologie

Institut für Technik der Informationsverarbeitung

Felix Pistorius

pistorius@kit.edu

Karlsruher Institut für Technologie (**KIT**)

Institut für Technik der Informationsverarbeitung (ITIV)

1. Aufgabe: Fanø–Code

1. Aufgabe

- Einige Studenten wollen ihre Verabredungen über das Internet koordinieren.
- Zu diesem Zweck legen sie den Zeitpunkt fest, nur über den Treffpunkt kommt keine Einigkeit zustande.
- Um nun die Übertragungszeiten gering zu halten, beschließen sie, die beliebtesten Treffpunkte im Fanø–Code zu kodieren.

1. Aufgabe – 1.1

- Die Gaststätten sollen in “billig” und “teuer” eingeteilt werden. Wie groß ist der Informationsgehalt der Aussage “billig”, wenn “billig” Preise bis max. € 3,00 beinhaltet?

Gaststätte	Bevorzugt von Anzahl an Studenten	Kosten pro Getränk / €
Brasil	1	4,00
Dom	2	4,00
Krokodil	1	3,50
Miro	2	3,00
Titanic	6	2,50
Ubu	6	3,00
Wien	5	3,50
Zero	3	3,50

1. Aufgabe – 1.1

- Die Gaststätten sollen in “billig” und “teuer” eingeteilt werden. Wie groß ist der Informationsgehalt der Aussage “billig”, wenn “billig” Preise bis max. € 3,00 beinhaltet?

$$p = \frac{\sum billige\ Kneipen}{\sum alle\ Kneipen} = \frac{3}{8} = 0{,}375$$

$$H_e = \mathrm{ld}\frac{1}{p} = \mathrm{ld}\frac{1}{0{,}375} = \underline{\underline{1{,}42\ bit}}$$

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

1. Aufgabe – 1.2

- Berechnen Sie die Wahrscheinlichkeit für den Besuch jeder einzelnen Gaststätte. Gehen Sie davon aus, dass die Wahrscheinlichkeit von der Beliebtheit abhängig ist.

Gaststätte	Bevorzugt von	Kosten pro Getränk / €	Auftrittswahrscheinlichkeit: p = Einzel El. / Gesamt
Brasil	1	4,00	p(Brasil) = 1 / 26 ≈ 0.0385
Dom	2	4,00	p(Dom) = 2 / 26 ≈ 0,077
Krokodil	1	3,50	p(Krokodil) = 1 / 26 ≈ 0,0385
Miro	2	3,00	p(Miro) = 2 / 26 ≈ 0,077
Titanic	6	2,50	p(Titanic) = 6 / 26 ≈ 0,231
Ubu	6	3,00	p(Ubu) = 6 / 26 ≈ 0,231
Wien	5	3,50	p(Wien) = 5 / 26 ≈ 0,192
Zero	3	3,50	p(Zero) = 3 / 26 ≈ 0,115
	Σ = 26		Σ = 1

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

1. Aufgabe – 1.3

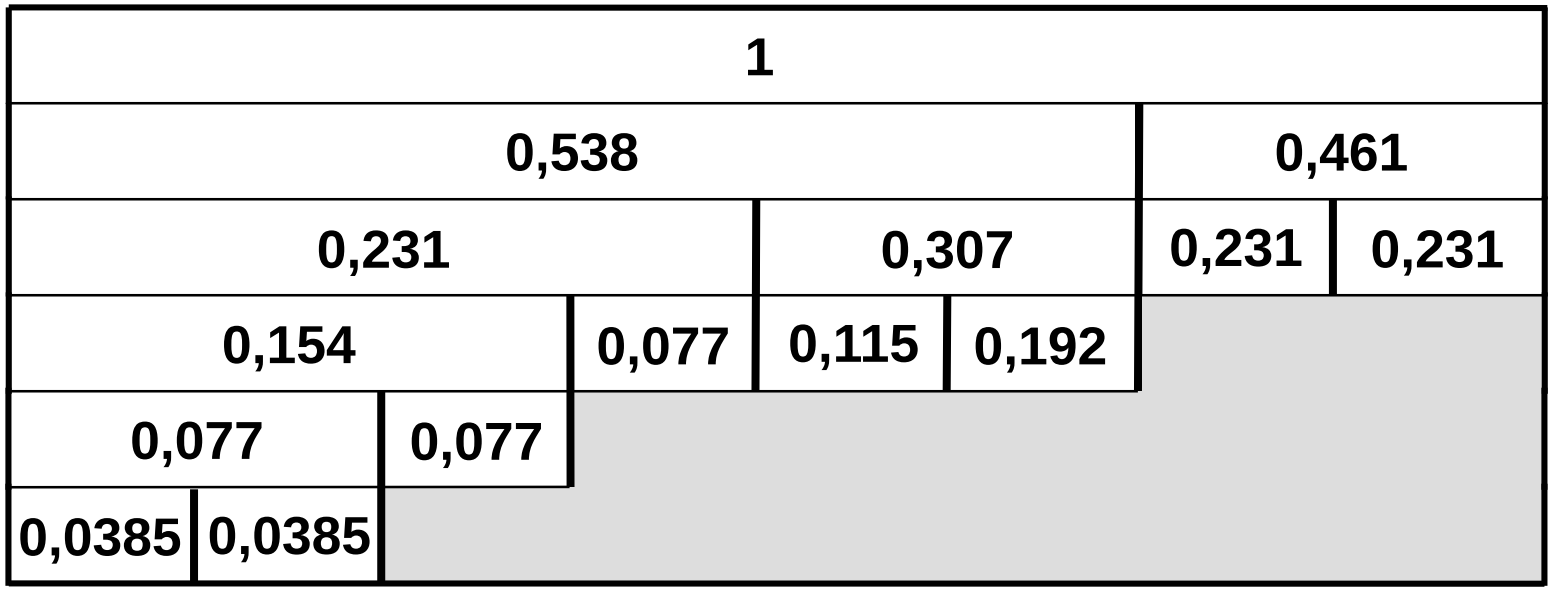
Gast-stätte	Krokodil	Brasil	Miro	Dom	Zero	Wien	Ubu	Titanic
p	0,0385	0,0385	0,077	0,077	0,115	0,192	0,231	0,231

Brasil	1	4,00	~~p(Brasil) = 1 / 26 ≈ 0,0385~~
Dom	2	4,00	~~p(Dom) = 2 / 26 ≈ 0,077~~
Krokodil	1	3,50	~~p(Krokodil) = 1 / 26 ≈ 0,0385~~
Miro	2	3,00	~~p(Miro) = 2 / 26 ≈ 0,077~~
Titanic	6	2,50	~~p(Titanic) = 6 / 26 ≈ 0,231~~
Ubu	6	3,00	~~p(Ubu) = 6 / 26 ≈ 0,231~~
Wien	5	3,50	~~p(Wien) = 5 / 26 ≈ 0,192~~
Zero	3	3,50	~~p(Zero) = 3 / 26 ≈ 0,115~~
	Σ = 26		Σ = 1

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

1. Aufgabe – 1.3

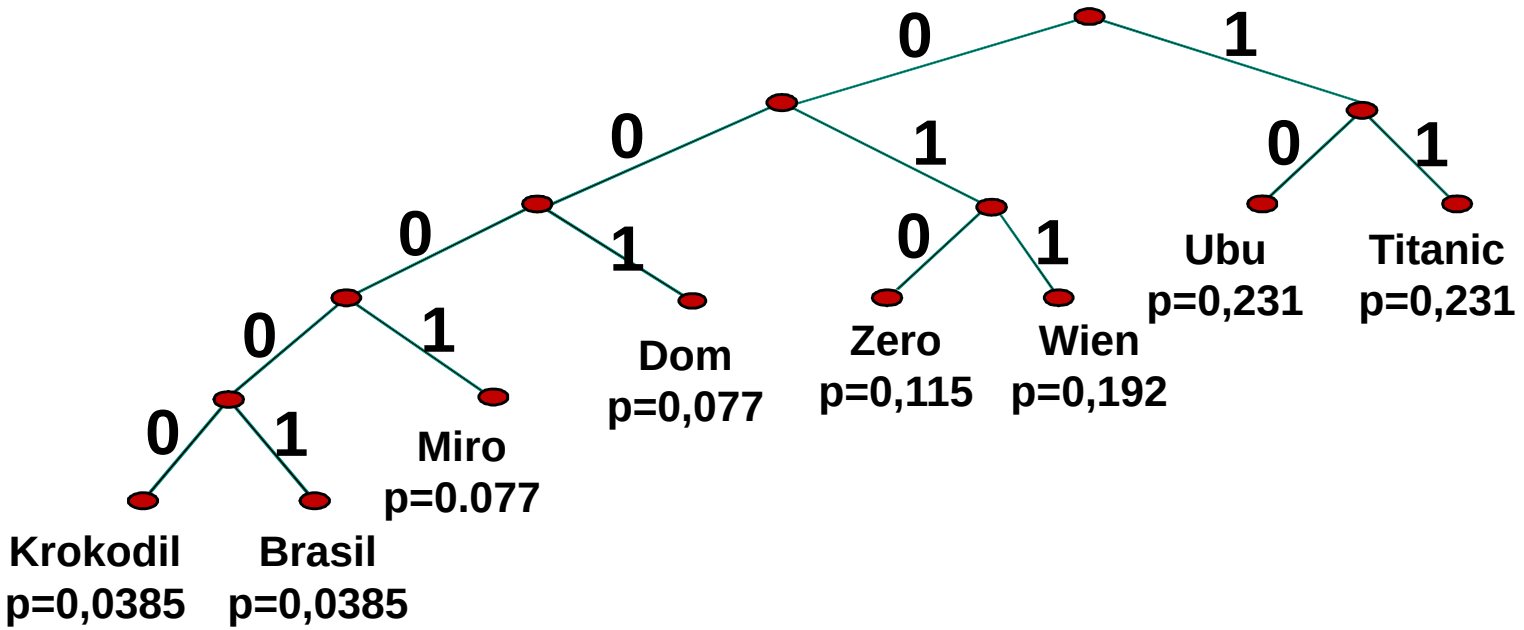
Gast-stätte	Krokodil	Brasil	Miro	Dom	Zero	Wien	Ubu	Titanic
p	0,0385	0,0385	0,077	0,077	0,115	0,192	0,231	0,231



KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

1. Aufgabe – 1.3

Gast-stätte	Krokodil	Brasil	Miro	Dom	Zero	Wien	Ubu	Titanic
Codierung	00000	00001	0001	001	010	011	10	11



KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

1. Aufgabe – 1.4

- Wie hoch ist die **mittlere Codewortlänge** und welchen Wert weist das **theoretische Minimum** auf?

Gast-stätte	Titanic	Ubu	Wien	Zero	Dom	Miro	Brasil	Krokodil
p	0,231	0,231	0,192	0,115	0,077	0,077	0,0385	0,0385
Codierung	11	10	011	010	001	0001	00001	00000
m	2	2	3	3	3	4	5	5

- **Mittlere Codewortlänge**:

$$\bar{m} = \sum_{i=1}^{n} p(x_i) * m(x_i) = 0{,}231 * 2 + \cdots + 0{,}0385 * 5 \approx 2{,}77\ bit$$

- Das **theoretische Minimum** (durchschnittlicher Informationsgehalt):

$$H = \sum_{i=1}^{n} p(x_i) * ld \frac{1}{p(x_i)} = 0{,}231 * ld \frac{1}{0.231} + \cdots + 0{,}0385 * ld \frac{1}{0.0385} \approx 2{,}72\ bit$$

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

Karlsruher Institut für Technologie

Institut für Technik der Informationsverarbeitung

Felix Pistorius

pistorius@kit.edu

Karlsruher Institut für Technologie (**KIT**)

Institut für Technik der Informationsverarbeitung (ITIV)

2. Aufgabe: Zahlensysteme

Zahlensysteme

- Zahlensysteme dienen zur transparenten Darstellung von Zahlen durch geeignete Ziffern und deren systematischen Anordnung
  -> elementar für die Automatisierung in der Digital- sowie Rechentechnik

- Polyadische Zahlensysteme geben den Ziffern ihren Wert in Abhängigkeit von ihrer Stelle innerhalb der systematischen stellenorientierten Anordnung

- Das Dezimalsystem ist der uns geläufigste Vertreter der polyadischen Zahlensysteme

- Die Stellenwerte entsprechen den Potenzen der Basis des jeweiligen polyadischen Zahlensystems

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

Polyadische Zahlensysteme: Aufbau

Beispiele:

1689_{10}	$= 1 * 10^3 + 6 * 10^2 + 8 * 10^1 + 9 * 10^0$	$= 1689$
1022_3	$= 1 * 3^3 + 0 * 3^2 + 2 * 3^1 + 2 * 3^0$	$= 35$
1011_2	$= 1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 1 * 2^0$	$= 11$
41_{12}	$= 4 * 12^1 + 1 * 12^0$	$= 49$

Allgemein: Aufbau einer Zahl N

$$N = d_n * R^n + \ldots + d_1 * R^1 + d_0 * R^0$$

N: Zahl im Zahlensystem

R: Basis, (Grundzahl, Radix) $R \geq 2$

R^i: Wertigkeit der i-ten Stelle

d_i: Ziffer der Stelle i

Z: Menge der Ziffern: $d_i \in Z = \{0,1,2, \ldots , R-1\}$

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

2. Aufgabe

- Ergänzen Sie folgende Tabelle durch Konvertierung der angegebenen Zahlen in die jeweiligen Zahlensysteme.

dual	oktal	dezimal	hexadezimal
1011 1001			
1101 0110			
	363		
	1021		
		317	
		1150	
			ED3
			C8E

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

2. Aufgabe

- Beispiel:
  - Binär → dezimal

$$(11010110)_B$$

$$=(1*2^7 + 1*2^6 + 0*2^5 + 1*2^4 + 0*2^3 + 1*2^2 + 1*2^1 + 0*2^0)_D$$

$$=(128 + 64 + 0 + 16 + 0 + 4 + 2 + 0)_D$$

$$=214_D$$

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

2. Aufgabe

Dez	Bin	Hex
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7

Dez	Bin	Hex
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

2. Aufgabe

- Beispiel:
  - Binär → oktal

    $(11010110)_B$

    $= (11\ 010\ 110)_B$

    $= (\ 3 \quad 2 \quad 6\ )_O$

  - Binär → hexadezimal

    $(11010110)_B$

    $= (1101\ 0110)_B$

    $= (\ D \quad 6\ )_H$

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

2. Aufgabe

dual	oktal	dezimal	hexadezimal
1011 1001			
1101 0110	326	214	D6
	363		
	1021		
		317	
		1150	
			ED3
			C8E

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

2. Aufgabe

- Beispiel:
  - oktal → binär → hexadezimal → dezimal

$$1021_O$$

$$=(001\ 000\ 010\ 001)_B$$

$$=(\underbrace{0010}\ \underbrace{0001}\ \underbrace{0001})_B$$

$$=(\quad 2 \quad\quad 1 \quad\quad 1 \quad)_H$$

$$=(2*16^2+1*16^1+1*16^0)_D$$

$$=(512 + 16 + 1)_D$$

$$= 529_D$$

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

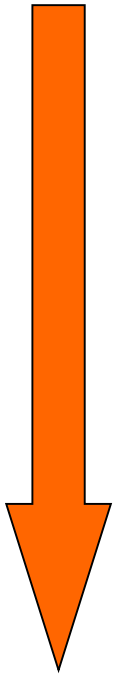
2. Aufgabe

dual	oktal	dezimal	hexadezimal
1011 1001			
1101 0110	326	214	D6
	363		
10 0001 0001	1021	529	211
		317	
		1150	
			ED3
			C8E

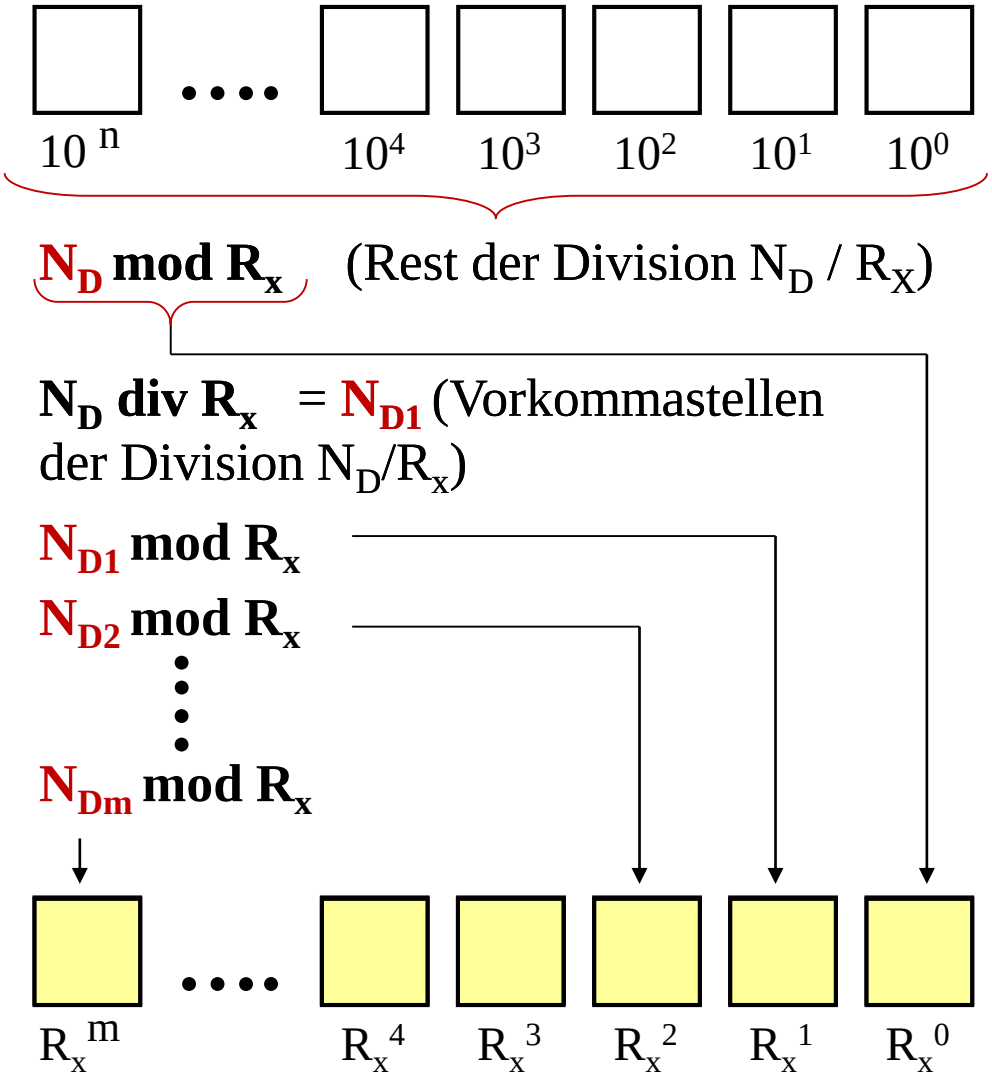
KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

Wandlung in andere Zahlensystemen

Dezimalzahl: N_D



Zahl in beliebigem Zahlensystem: $N_{irgendwas}$



Basis R_x des Zahlensystems: $N_{irgendwas}$

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

Wandlung in andere Zahlensystemen

Dezimalzahl: $N_D = 2016$

2	0	1	6
10^3	10^2	10^1	10^0

Beispiel:
Wandlung der Dezimalzahl 2016 in eine Hexadezimalzahl

2016 mod 16 = 0

2016 div 16 = 126

126 mod 16 = 14

126 div 16 = 7

7 mod 16 = 7

Hexadezimalzahl: N_H

7	E	0
16^2	16^1	16^0

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

2. Aufgabe

- Beispiel: dezimal → oktal

1150_D

1150 / 8 = 143 Rest 6

143 / 8 = 17 Rest 7

17 / 8 = 2 Rest 1

2 / 8 = 0 Rest 2

1150_D

$= 2176_O$

$= 010\ 001\ 111\ 110_B$

$= 47E_H$

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

2. Aufgabe

dual	oktal	dezimal	hexadezimal
1011 1001	271	185	B9
1101 0110	326	214	D6
1111 0011	363	243	F3
10 0001 0001	1021	529	211
0001 0011 1101	475	317	13D
0100 0111 1110	2176	1150	47E
1110 1101 0011	7323	3795	ED3
1100 1000 1110	6216	3214	C8E

Karlsruher Institut für Technologie

Institut für Technik der Informationsverarbeitung

Felix Pistorius

pistorius@kit.edu

Karlsruher Institut für Technologie (**KIT**)

Institut für Technik der Informationsverarbeitung (ITIV)

3. Aufgabe: Huffman–Code

3. Aufgabe

- Bei einer Alarmanlage sollen alle Einbrüche und Einbruchsversuche in einem nichtflüchtigen Speicher protokolliert werden. Da bei diesem Baustein jede Speicherzelle nur 1x geschrieben werden kann, soll dieser möglichst gut ausgenutzt werden.
- Aus den Statistiken der Polizei ist bekannt, dass Einbrüche durch eine Tür 4x häufiger sind als durch ein Bad- oder Küchenfenster. Weiterhin ist bekannt, dass Einbrüche durch eine Tür 8x häufiger sind als durch ein sonstiges Fenster im Erdgeschoss. Schließlich zeigt die Statistik, dass ein Einbruch durch ein Fenster im 1. Stockwerk 16x seltener auftritt, als ein Einbruch durch eine Tür.

3. Aufgabe – 3.2

- Erstellen Sie eine optimale Codierung für die möglichen Orte und tragen Sie Ihre Ergebnisse in die oben vorgegebene Tabelle ein. Nutzen Sie hierzu das Huffmanverfahren.

Ort	Häufigkeit	Wahrscheinlichkeit	Ermittelte Codierung	
Wohnungstür	16	16/50		**WT: 16/50**
Balkontür	16	16/50		**BT: 16/50**
Badfenster	4	4/50		**BF: 4/50**
Küchenfenster	4	4/50		**KF: 4/50**
Schlafzimmerfenster	2	2/50		**SF: 2/50**
Wohnzimmerfenster 1	2	2/50		**WF1: 2/50**
Wohnzimmerfenster 2	2	2/50		**WF2: 2/50**
Kinderzimmerfenster 1	1	1/50		**KF1: 1/50**
Kinderzimmerfenster 2	1	1/50		**KF2: 1/50**
Atelierfenster 1	1	1/50		**AF1: 1/50**
Atelierfenster 2	1	1/50		**AF2: 1/50**
	$\sum = 50$			

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

Optimale Codes: Huffman-Code

Huffman's Codierungs-Algorithmus (HKA) (1)

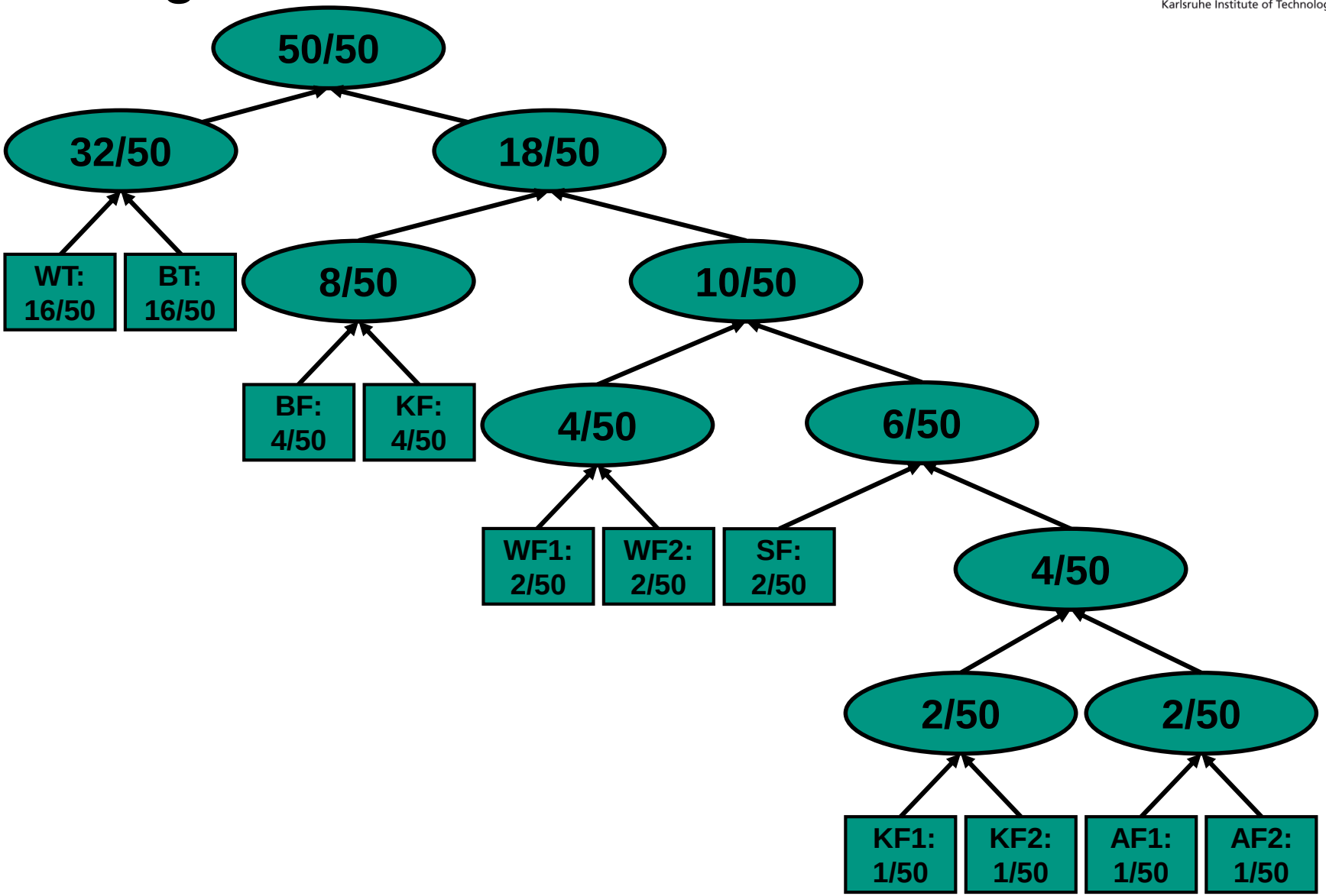
- HKA ist ein *Greedy-Algorithmus*
- HKA konstruiert einen *Baum* (= azyklischer zusammenhängender Graph) T, der die *Codierung* der einzelnen Zeichen repräsentiert
- Sortierte Liste Q → jeweils die beiden Zeichen in Q mit niedrigsten Auftrittswahrscheinlichkeiten (AWS) finden
- Zwei gefundene Zeichen werden zu neuem *Element z* "verschmolzen", dessen AWS die Summe der beiden Einzelwahrscheinlichkeiten zugeordnet wird

  → Einsortierung von z in Q entsprechend seiner AWS
- Abbruch des HKA:

  nur noch 1 Objekt in Q übrig → *Codierungsbaum* T

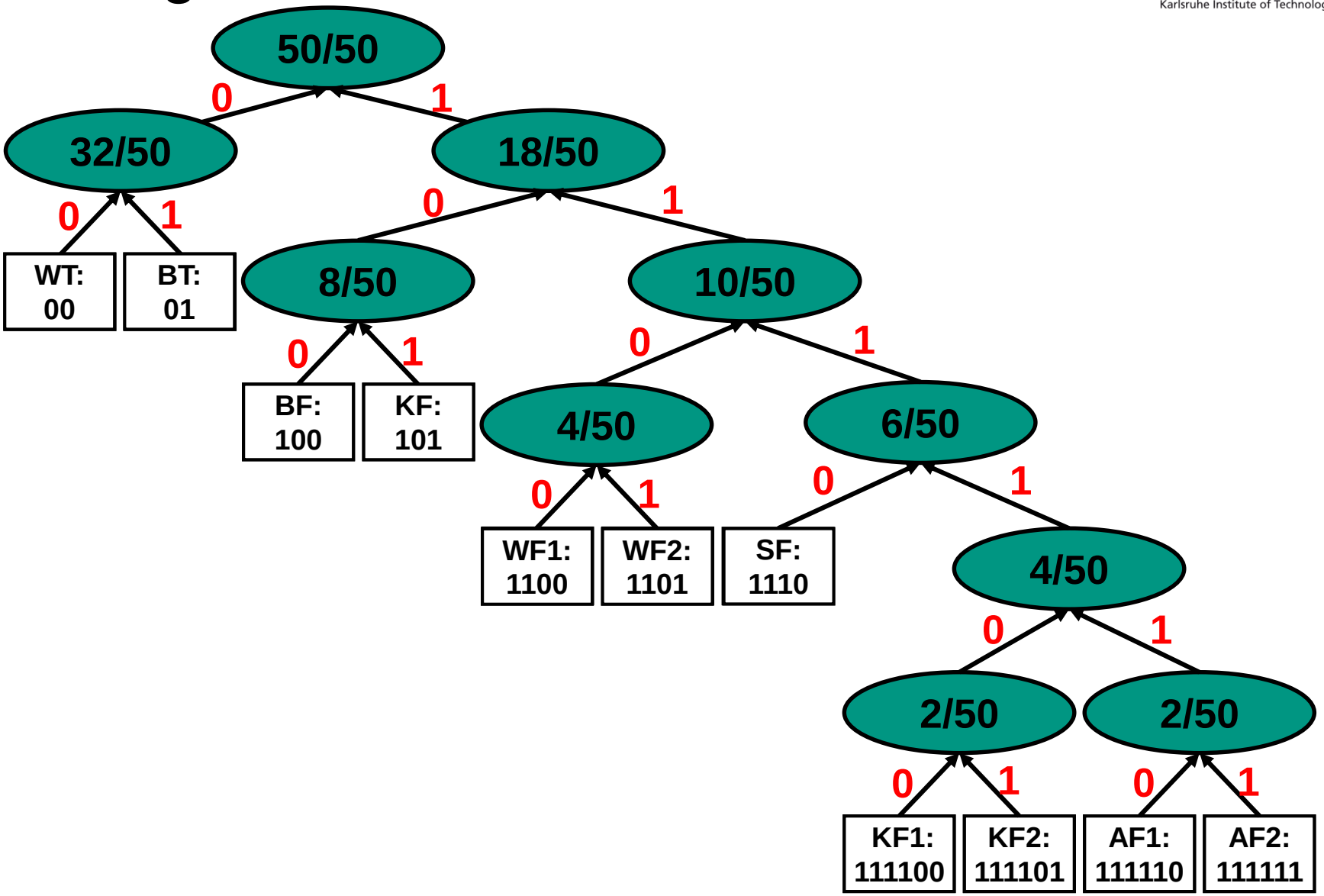
KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

3. Aufgabe – 3.2



KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft
Institut für Technik der Informationsverarbeitung (ITIV) | © 2016

3. Aufgabe – 3.2


50/50
0
1
32/50
18/50
0
1
0
1
WT:
00
BT:
01
8/50
10/50
0
1
0
1
BF:
100
KF:
101
4/50
6/50
0
1
0
1
WF1:
1100
WF2:
1101
SF:
1110
4/50
0
1
2/50
2/50
0
1
0
1
KF1:
111100
KF2:
111101
AF1:
111110
AF2:
111111

3. Aufgabe – 3.3

Ort	Häufigkeit	Wahrscheinlichkeit	Ermittelte Codierung	
Wohnungstür	16	16/50	00	WT: 16/50
Balkontür	16	16/50	01	BT: 16/50
Badfenster	4	4/50	100	BF: 4/50
Küchenfenster	4	4/50	101	KF: 4/50
Schlafzimmerfenster	2	2/50	1110	SF: 2/50
Wohnzimmerfenster 1	2	2/50	1100	WF1: 2/50
Wohnzimmerfenster 2	2	2/50	1101	WF2: 2/50
Kinderzimmerfenster 1	1	1/50	111100	KF1: 1/50
Kinderzimmerfenster 2	1	1/50	111101	KF2: 1/50
Atelierfenster 1	1	1/50	111110	AF1: 1/50
Atelierfenster 2	1	1/50	111111	AF2: 1/50

- Wie viele Codeworte können Sie in einem Speicher mit einer Kapazität von 1024 Bit abspeichern? Berechnen Sie dazu zuerst die mittlere Codewortlänge.
- Mittlere Codewortlänge: $\bar{m} = \sum_{i=1}^{n} p(x_i) * m(xi)$

$$\bar{m} = 2\left(2 * {}^{16}/_{50}\right) + 2\left(3 * {}^{4}/_{50}\right) + 3\left(4 * {}^{2}/_{50}\right) + 4\left(6 * {}^{1}/_{50}\right) = {}^{136}/_{50}\ (\approx 2,72)\ \text{bit}$$

- Speicherkapazität: $n = \frac{Speicherkapazität}{\bar{m}} = \frac{1024}{{}^{136}/_{50}} = \frac{51200}{136}\ (\approx 376,5)\ CWs$