



# Informatik I WS 07/08

## Tutorium 24

8.11.07

Bastian Molkenthin

E-Mail: [infotut@sunshine2k.de](mailto:infotut@sunshine2k.de)

Web: <http://www.sunshine2k.de>



Universität Karlsruhe (TH)  
Forschungsuniversität · gegründet 1825



- Information und Bezugssysteme
- Informationstheorie
- Steuerung und Regelung
- Algorithmen
- O-Kalkül
- UML



Stelle den Zusammenhang zwischen Nachricht, Information und Bezugssystem dar!

## **Nachricht:**

Mitteilung, bei der vom Medium und den Einzelheiten der Signale und Signalparameter abstrahiert wird.

## **Information:**

Einer Nachricht zugeordnete Bedeutung.

**Nachricht**  **Bedeutung**

- Für die Zuordnung von Bedeutung wird ein Bezugssystem benötigt.
- Information wird gewonnen durch die Interpretation von Nachrichten auf Grundlage eines Bezugssystem!



„3 Zentimeter sind kürzer als 2 Zoll.“

Erklären sie, welche Informationen Sie benötigen, um die Korrektheit dieser Aussage zu erkennen!

**Lösung:**

Vorschrift 1: Ziffer 3 mit Längeneinheiten Zentimeter

Vorschrift 2: Ziffer 2 mit Längeneinheiten Zoll

Vorschrift 3: 3 Zentimeter entspricht  $3 \cdot 1$  Zentimeter

Vorschrift 4: 2 Zoll entspricht  $2 \cdot 1$  Zoll

Vorschrift 5: kürzer als ist eine Ordnungsrelation

Vorschrift 6: 2,54 Zentimeter ist gleich 1 Zoll

 Somit ist die Aussage korrekt!

## Aufgabe 2 : Bezugssystem



Interpretiert folgende Nachricht anhand ihres Bezugssystems!  
Verwende Zeichen-, Buchstaben-, Wörter-, Satz- sowie  
Aussagenebene!

Nachricht:

1100-111-01-0-10-01-11-010-101-010-1100-1-11-01

Interpretationsvorschrift:

H = 11

L = 101

M = 0

T = 01

E = 010

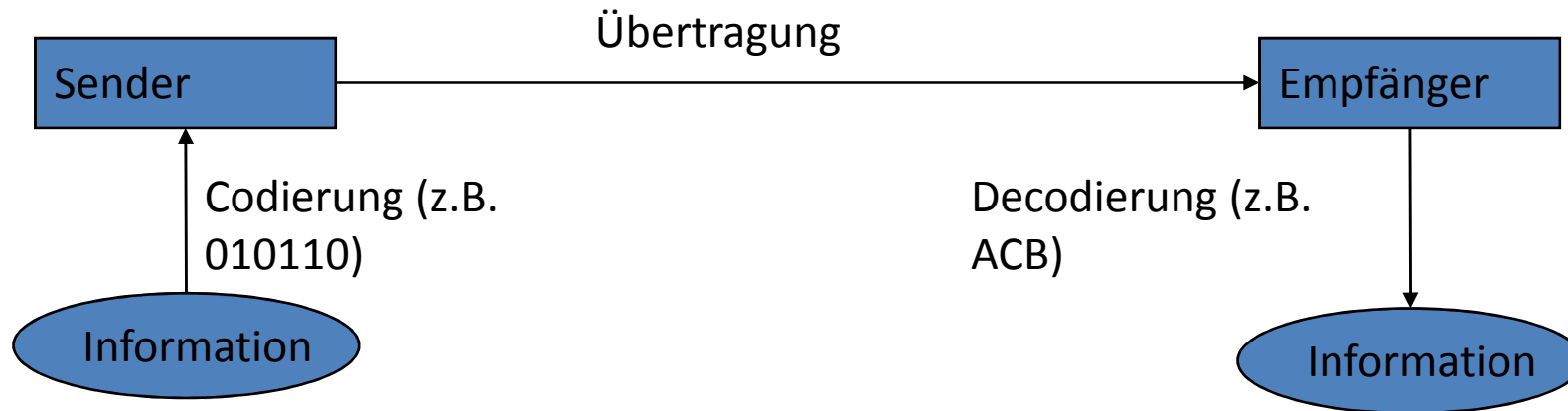
I = 1100

A = 10

S = 111

C = 1

- = Trennzeichen



Definition:

- A enthält den Zeichenvorrat einer Nachrichtenquelle X
- $P(X = a_i)$  ist die Wahrscheinlichkeit für das Auftreten von Zeichen  $a_i$
- dann ist die **Entropie  $H(X)$**  der Nachrichtenquelle X:

$$H(X) = \sum_{a_i \in A} P(X = a_i) * \log_2 \frac{1}{P(X = a_i)}$$



### Aufgabe:

Bestimmt die Entropie für die Nachrichtenquelle 'BCCA'.

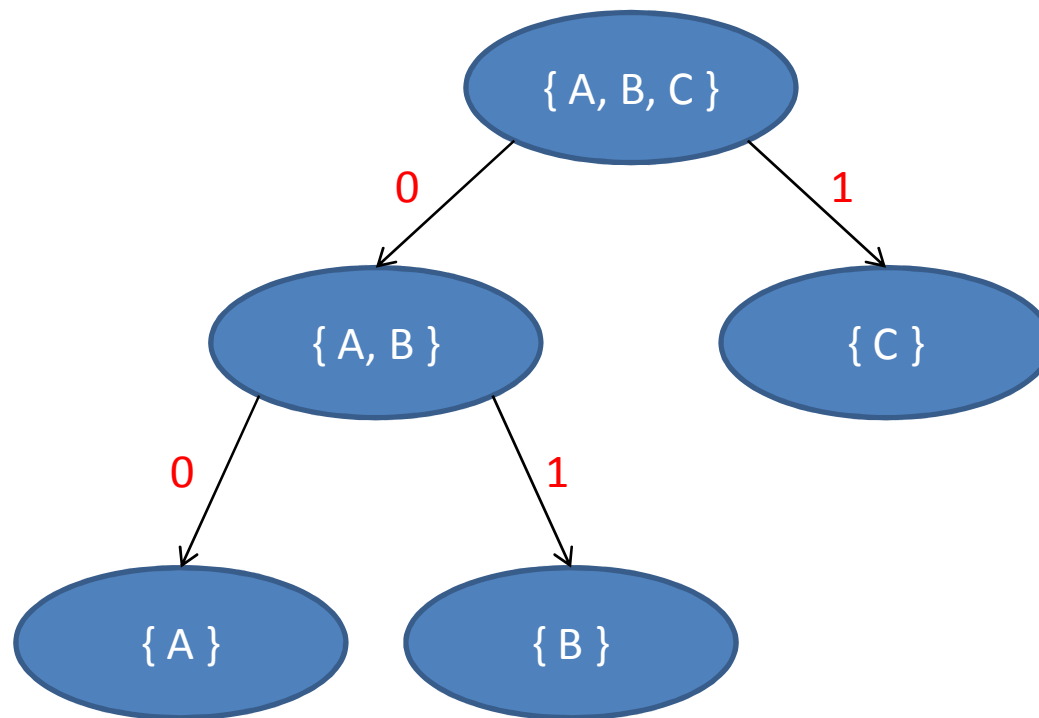
### More facts:

- Entropie heißt auch mittlerer Entscheidungsgehalt.
- Einheit ist „bit pro Zeichen“.
- Liefert die Quelle **sicher** immer eine 0 bzw. 1 dann ist der mittlere Informationsgehalt  $H = 0$ .
- Entscheidungsgehalt ist also „Unsicherheit über das nächste Zeichen“.
- $H$  ist maximal wenn alle Zeichen gleichverteilt.



## Aufgabe:

Geben sie einen Entscheidungsbaum mit Gleichverteilung an, der die Nachricht am kürzesten codiert.



Codierungstabelle

C	1
A	00
B	01





Die mittlere Codewortlänge besagt, wie viele Bit im Schnitt benötigt werden, um ein Zeichen zu kodieren.

$$\text{Mittlere Codewortlänge} = \sum_{a_i \in A} p(a_i) * \text{Codelänge}(a_i)$$

Beispiel:

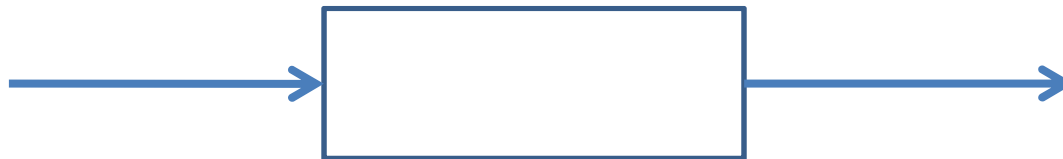
$p(B) = p(A) = \frac{1}{4}$  und werden jeweils mit 2 Zeichen kodiert

$p(C) = \frac{1}{2}$  und wird mit 1 Zeichen kodiert

$$\text{Mittlere Codewortlänge} = 2 * \frac{1}{4} + 2 * \frac{1}{4} + \frac{1}{2} * 1 = 1,5 \text{ bit}$$



## Steuerung:



Ein Algorithmus oder eine Schaltung nimmt Einfluß auf eine beliebige Größe.

## Regelung:



Ein Algorithmus oder eine Schaltung nimmt Einfluß auf eine beliebige Größe. Diesmal findet eine Rückkopplung statt und die beeinflusste Größe wird gemessen und fließt wieder in den Algorithmus ein.



## **(eine mögliche) Definition:**

Ein Algorithmus ist ein Verfahren mit einer präzisen (d.h. in einer eindeutigen Sprache abgefassten) endlichen Beschreibung unter Verwendung effektiver (im Sinne von tatsächlich ausführbarer) Verarbeitungsschritte.

Was für Algorithmen kennt ihr?

- im Alltag
- in der Informatik



Auch für PseudoCode gibt es Richtlinien. So wird jeder Schritt nummeriert, Schleifen, Ein- und Ausgaben, sowie das Setzen von Werten wie folgt beschrieben:

**Ein- und Ausgaben:** Schritt 1: Lese x.  
Schritt 2: Gebe x aus.

**Schleifen:** Schritt 1: Solange (BEDINGUNG)  
Schritt 1.1: Anweisung 1  
Schritt 1.2: Anweisung 2

**Setzen von Werten:** Schritt 1: Setze  $x = y$

**Setzen von Werten:** Schritt 1: Falls(BEDINGUNG)  
Schritt 1.1: Anweisung 1  
sonst  
Schritt 1.2: Anweisung 2



## **Aufgabe 1:**

Sei  $n$  eine natürliche Zahl. Gesucht ist die Anzahl ihrer Ziffern.  
(Beispiel : Die Ziffer 999 hat 3 Ziffern.)

## **Aufgabe 2:**

Gebe einen Algorithmus  $M$  an, der zwei (natürliche) Zahlen dividiert und den Rest ausgibt.  
(Beispiel :  $M(13,5)$  ergibt „2 Rest 3“)



## Was ist die Effizienz eines Algos?

Bedarf an Speicherplatz und Zeit, die Algorithmus zur Ausführung benötigt. Dies ist abhängig von der Anzahl der Eingabewerte.

## Wie bestimme die Effizienz:

- Ausprobieren

Implementierung der Algorithmen, Ausführen und Zeit messen

➡ Abhängig von Sprache, Übersetzer, Hardware etc. !

- Theoretische Verfahren

- Abstraktion von Sprache etc.

- Abhängig von der Eingabelänge

- mathematisch fundiert und beweisbar

➡ z.B. O-Kalkül



## Definition?

$$O(g(n)) = \{ f(n) \mid \exists n_0 \in \mathbb{N}, \exists c \in \mathbb{R} \forall n \geq n_0 : 0 \leq f(n) \leq c * g(n) \}$$

## Also...

$O(g(n))$  enthält alle Funktionen, die asymptotisch höchstens so schnell wachsen wie  $g(n)$ .

## Wir kümmern uns nicht...

... was am Anfang passiert :  $\exists n_0 \in \mathbb{N}, \forall n \geq n_0$

... um konstante Faktoren:  $\exists c \in \mathbb{R} \dots c * g(n)$




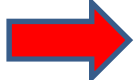
- Einzelnen Anweisung wie „Gebe x aus“, „Berechne x+y“, ... wird die Effizienz  $O(1)$  zugeordnet
- Algorithmen, welche in „größeren“ Klassen wie  $O(n)$ ,  $O(n^3)$  liegen, entstehen meist durch (verschachtelte) Schleifen oder rekursiven Aufruf von Funktionen
- „Vereinfachen“ im O-Kalkül:
  - $O(1) + O(3) \rightarrow O(1)$
  - $O(n) + O(n) \rightarrow O(n)$
  - $O(n) * O(n) \rightarrow O(n^2)$
  - $O(n) + O(n^2) \rightarrow O(n^2)$



# O-Kalkül (Beispiele)



Schritt 1:	Setze $k = 1$	$O(1)$	
Schritt 2:	Wiederhole solange $k \leq n$	$n * (O(1)+O(1))$	
Schritt 2.1:	Setze $s = s+k$	$= O(n)$	
Schritt 2.2:	Setze $k = k+1$		
Schritt 3:	Gebe $s$ aus	$O(1)$	 $O(n)$

Schritt 1:	Setze $k = i = 0$	$O(1)$	
Schritt 2:	Solange $k < n$ :	$n *$	
Schritt 2.1:	Solange $i < n$ :	$n *$	
Schritt 2.1.1:	Setze $i = i+1$	$O(1)$	
Schritt 2.1.2:	Setze $x = x+1$	$O(1)$	
Schritt 2.2:	Setz $k = k+1$	$O(1)$	
Schritt 2.3:	Setze $i = 0$	$O(1)$	
Schritt 3:	Gebe $x$ aus	$O(1)$	 $O(n^2)$



*Klassen* sind Schablonen für Gegenstände.

In ihnen sind Gemeinsamkeiten einer Menge von Gegenständen vereint, vor allem:

- Verhalten ( ➡ Methoden)
- Eigenschaften ( ➡ Attribute)

ClassName
Attribut 1
Method 1

*Objekte* sind konkrete Instanzen einer Klasse.

Erst Objekte haben Werte für die Attribute der Klasse und kein eigenes „Verhalten“.

MyClass : ClassName
Attribut 1 = Value 1
Attribut 2 = Value 2

Für dieses Übungsblatt : Datentypen müssen nicht angegeben werden.

Für Informatik I gilt: Attribut- und Methodennamen auf Englisch!



Ein PC lässt sich durch seine CPU-Taktrate, seinen Arbeitsspeicher, seine Festplattenkapazität und sein Betriebssystem bezeichnen. Erstelle ein UML-Klassendiagramm inklusive der Methoden zum Lesen und Setzen der Attribute.

PC
CPU
RAM
HD
OS
getCPU()
setCPU()
getRAM()
setRAM()
...

Nicht vollständig  
wegen Platzgründen!



Erstelle ein Objektdiagramm welches folgenden PC beschreibt!

Intel Pentium IV mit 3.2 GHZ, 512 MB DDR RAM,  
Western Digital IDE 10000U/min 400GB, Microsoft Windows XP

MeinPC : PC
CPU = 3.2
RAM = 512
HD = 400
OS = „MS Windows XP“



# Fragen ???



Viel Spaß mit dem Übungsblatt!