



Technische  
Universität  
Braunschweig

Institute of Operating Systems  
and Computer Networks



# Algorithmen und Datenstrukturen

## Große Übung #1

Phillip Keldenich, Arne Schmidt

27.10.2016

# Organisatorisches

# Homepage

Aktuelle Informationen, Hausaufgaben, Slides auf:

<https://www.ibr.cs.tu-bs.de/courses/ws1617/aud/webpages/index.html>

Anmeldung zu kleinen Übungen über:

<https://www.ibr.cs.tu-bs.de/courses/ws1617/aud/>

# Anmeldung

Group	Time (biweekly)	Place	Tutor	Participants
1	Dienstag, 08:00 - 09:30	IZ 305	Sören van der Wall	
2	Dienstag, 08:00 - 09:30	IZ 358	Jakob Keller	
3	Dienstag, 13:15 - 14:45	IZ 358	Eva Bolle	
4	Dienstag, 16:45 - 18:15	IZ 161	Micha Horiboge	
5	Mittwoch, 08:00 - 09:30	IZ 358	Micha Horiboge	
6	Mittwoch, 08:00 - 09:30	IZ 305	Sören van der Wall	
7	Mittwoch, 13:15 - 14:45	IZ 305	Alexander Hill	
8	Mittwoch, 15:00 - 16:30	IZ 358	Jakob Keller	
9	Mittwoch, 15:00 - 16:30	IZ 160	Alexander Hill	
10	Mittwoch, 16:45 - 18:15	IZ 358	Moritz Pfister	
11	Mittwoch, 16:45 - 18:15	Container 3	Yannic Lieder	
12	Donnerstag, 13:15 - 14:45	IZ 305	Antje Mönch	
13	Donnerstag, 13:15 - 14:45	IZ 160	Eva Bolle	
14	Donnerstag, 15:00 - 16:30	Container 3	Mai Hellmann	
15	Donnerstag, 16:45 - 18:15	IZ 160	Mai Hellmann	
16	Freitag, 11:30 - 13:00	IZ 358	Moritz Pfister	
17	Freitag, 13:15 - 14:45	IZ 305	Dennis Luck	
18	Freitag, 13:15 - 14:45	IZ 358	Yannic Lieder	
19	Freitag, 15:00 - 16:30	IZ 305	Dennis Luck	
20	Freitag, 15:00 - 16:30	IZ 358	Antje Mönch	

Anmeldung bis Sonntag 24 Uhr.

# Semesterplan

## Semesterplan AuD WS16/17

Woche	Vorlesung (Di. + Mi.)	Gr. Übung (Do.)	Kl. Übung (Mo.-Fr.)	HA Ausgabe (Mo.)	HA Abgabe (Mo. bis 13:15 Uhr)	HA Rückgabe (in kl UE)
43	0,1	1		HA0a*		
44	2,3		1	HA0b*, HA1		HA0a
45	4,5	2				
46	6,7		2	HA2	HA1	HA0b
47	8,9	3				
48	10,11		3	HA3	HA2	HA1
49	12,13	4				
50	14,15		4	HA4	HA3	HA2
51	Weihnachtsferien					
52						
1						
2	16,17	5	5	HA5	HA4	HA3
3	18,19					
4	20,21	6	6	HA6	HA5	HA4
5	22,23					
6	24,25	7	7		HA6**	HA5
*: Mündliche Aufgaben, Bearbeitung in ersten kleinen Übungen; keine Bewertung						
**: Die Besprechung des 6. Hausaufgabenblattes findet in der letzten Großen Übung statt. Die korrigierten Hausaufgaben können im Laufe der 7. Kalenderwoche abgeholt werden.						

# Hausaufgaben

- 8 Blätter
  - 2 unbewertet
  - 6 bewertet
- Je 60 Punkte
- Studienleistung: 50% aller Punkte
  - Studienleistung ist **keine** Voraussetzung, um an der Prüfung teilzunehmen
- Zusammen arbeiten, einzeln aufschreiben und abgeben

# Klausur

Findet statt am 01.03.2017, 14:00 - 17:00 Uhr.

# Mailingliste

- Anmeldung über Homepage.
- Sämtliche Ankündigungen werden dort verkündet.
- Bietet Möglichkeit Fragen zu stellen.
- Hiwis können über die Mailingliste erreicht werden.



# Fragen!

Stellt Eure Fragen:

- über die Mailingliste
- in: Vorlesung, Gr. Übung, Kl. Übungen
- Euren Tutoren
- Sprechstunde von Phillip Keldenich und Arne Schmidt (Jeweils nach Vereinbarung)
- Sprechstunde von Prof. Sándor Fekete (Mi. 13:15 - 14:00 Uhr)

# Fragen?

# Problem

Allgemeine Fragestellung. Meistens formuliert mit:

**Eingabe:** Was ist gegeben?

**Ausgabe:** Was ist gesucht?

Lösung eines Problems:

*Angabe eines Algorithmus.*

# Instanz

Eine Problemstellung mit konkreter Eingabe.

Lösung einer Instanz:

*Angabe einer konkreten Ausgabe.*

# Ein Beispiel - Größter Gemeinsamer Teiler

Problem ggT:

**Gegeben:** Zwei Zahlen  $a$ ,  $b$ .

**Gesucht:** Der größte gemeinsame Teiler  $q$  von zwei Zahlen  $a$  und  $b$ .

Lösung: Euklidischer Algorithmus (dazu gleich mehr)

Instanzen:

- ggT(5, 102); Lösung: 1
- ggT(8, 64); Lösung: 8

## Beispiel 2 - Fibonacci-Zahlen

Definition der Fibonacci-Zahlen:

$$F(0) := 0, F(1) := 1, F(n) := F(n - 1) + F(n - 2)$$

Problem Fibonacci:

**Gegeben:** Eine Zahl  $n$ .

**Gesucht:** Die  $n$ -te Fibonacci-Zahl

Lösung: Rekursion aufschlüsseln. (Siehe Blatt0a)

Instanzen:

- $F(6)$ ; Lösung: 8
- $F(17)$ ; Lösung: 1597
- $F(200)$ ; Lösung: 280571172992510140037611932413038677189525

- Beschreiben durch:
  - Prosatext
  - *Pseudocode*
- Ausführbarkeit
- Endlichkeit
- Endliche Ausführzeit
- Endlicher Speicherbedarf

Also nicht:

- „Gib 5 oder 1 aus.“
- „Sei  $F(0) = 0$ ,  $F(1) = 1$ ,  
 $F(2) = 1$ ,  $F(3) = 2$ ,  $F(4) = 3$ ,  
 $F(5) = 5$ ,  $F(6) = 8, \dots$ “

# Warum Pseudocode?

Euklidischer Algorithmus als Text:

Solange die Zahl  $b$  nicht 0 ist, wählen wir  $h = a \bmod b$ , setzen  $a$  auf  $b$  und  $b$  auf  $h$ . Nachdem  $b = 0$ , geben wir  $a$  zurück

Euklidischer Algorithmus als Pseudocode:

```
function EUKLID( $a, b$ )  
  while  $b \neq 0$  do  
     $h \leftarrow a \bmod b$   
     $a \leftarrow b$   
     $b \leftarrow h$   
  end while  
  return  $a$   
end function
```

(Für modulo gilt  $r = a \bmod b$ , sodass  $a = q * b + r$  für ein  $q \in \mathbb{Z}$  gilt mit  $r < b$ .)



## Zuweisung:

- $a \leftarrow b$  (Weist  $a$  den Wert  $b$  zu.)
- $a \leftarrow b + c$  (Weist  $a$  die Summe von  $b$  und  $c$  zu.)
- $A \leftarrow B$  (Weist der Menge  $A$  die Menge  $B$  zu.)

# Pseudocode - Bausteine

## Schleifen:

- **for**  $a$  **in** start, ..., ende **do**  
    ...  
    **end for**
- **for**  $a \leftarrow b$  **(down)to**  $c$  **do**  
    ...  
    **end for**
- **while** Bedingung **do**  
    ...  
    **end while**
- **repeat**  
    ...  
    **until** Bedingung

## Bedingte Anweisung:

**if** Bedingung **then**

*Aktionen, falls Bedingung wahr ist*

**else**

*Aktionen, falls Bedingung falsch ist*

**end if**

## Ausgaben/Rückgaben:

- **print** *a* (*Gibt a aus und beendet die Funktion nicht*)
- **return** *a* (*Gibt a zurück und beendet die Funktion*)

## Methoden:

- Funktion-Körper:

```
function NAME(Param1, Param2,...)
```

```
    ...
```

```
end function
```

- Aufruf:

```
NAME(a, b,...) )
```

# Pseudocode - Beispiel

Problem SUM:

**Gegeben** Zwei Zahlen  $s$  und  $e$

**Gesucht** Die Summe von  $s$  bis  $e$ ,  
oder 0 falls  $s > e$ .

```
function SUM( $s$ ,  $e$ )  
  if  $s > e$  then  
    return 0  
  end if  
  for  $i \leftarrow s + 1$  to  $e$  do  
     $s \leftarrow s + i$   
  end for  
  return  $s$   
end function
```

Instanz:

Berechne die Summe von 4 bis 8,  
also SUM(4,8)

Lösungsschritte:

- $4 > 8$ ? Nein.
- Betrachte For-Schleife nach jeder Iteration:

$i$	$s$
5	9
6	15
7	22
8	30

- Gib 30 zurück.

# Pseudocode - Variationen

- Auf Deutsch
  - Wenn ... Dann ... Sonst ...
  - Solange ...
  - Für  $i \leftarrow 1, \dots, k \dots$
- Anstatt  $\leftarrow$  schreibt man gelegentlich =
- Mischung aus Prosa und Pseudocode:

```
function PROSACODE(s)
    if (Es existiert  $k \in \mathbb{Z}$ , sodass  $0 < k < s$ ) then
        Wähle  $h = 1$ 
    else
         $h = 2$ 
    end if
    return  $s - h$ 
end function
```

# Zusammenfassung

- Organisation
  - Anmeldung kl. Übungen!
  - Hausaufgaben
  - Fragen stellen
- Unterschied Problem/Instanz
- Pseudocode aufschreiben und ausführen