

Tut 01

Monday, 15. April 2019 11:54

0. ORGANISATORISCHES

Fu Zheng (Frank)

f.zheng@campus.tu-berlin.de

Sprechstunde: Mi 10-12h, MAR 6.029

- Sprechstunde in Form einer betreuten Rechnerzeit
 - Fragen zur VL, Tut, HA
 - Was tun wenn der Prof. mich mobbt?
- Formalitäten im Folgenden sind auch auf dem Merkblatt
- Bin ich hier richtig? (insb. WiIngs)
 - Wir sind nicht das fortsetzende Modul im SoSe, wenn ihr letztes Semester mit dem 9LP-Modul begonnen habt!
 - vgl. ISDA!

0.1.1. Hausaufgaben

- pro Woche muss 1 HA-Blatt bearbeitet werden
 - ab jeden Mo. auf ISIS
 - 3-4er Gruppen, Einteilung nächste Woche
 - Abgabe auf ISIS, bis spätestens vor Tutoriumsbeginn
 - alles oder nichts: 50% oder mehr richtig → bestanden!
- pro Hausaufgabenblock darf 1 HA nicht bestanden werden
- Online Test: 1/3 bestehen
- Do's & Dont's (Tut. Blatt)

0.1.2 Klausur

- Das ganze für die Zulassung zur Klausur
- Am Ende des Semesters, schriftlich
- Klausur ist EASY bei fleißiger Teilnahme
- Alle Klarheiten beseitigt?

1. WIEDERHOLUNG JAVA

- Wichtig: Das Formale (Syntax) wird vorausgesetzt aus InfTech I
- Zweck: Abgefahrenes Zeug erstellen, ohne dass uns die Programmiersprache daran hindert!

```
public interface Comparable {  
    int compareTo(Comparable c);  
}
```

Lsg.:

```
public class Rechteck implements Comparable {  
    private double l, w;  
    public Rechteck(double l, double w) {  
        this.l = l;  
        this.w = w;  
    }  
    public double getFlaecheninhalt() {  
        return l * w;  
    }  
    public int compareTo(Comparable c) {  
        if (c instanceof Rechteck) {  
            if (getFlaecheninhalt() > ((Rechteck) c).getFlaecheninhalt())  
                return 1;  
            else if (getFlaecheninhalt() == ((Rechteck) c).getFlaecheninhalt())  
                return 0;  
            else  
                return -1;  
        } else {  
            System.out.println("Kein Rechteck!");  
            return -2;  
        }  
    }  
}
```

- TestComparable kompilieren, ausführen

2. WIEDERHOLUNG REKURSION

- WTF?!
- Methoden, die sich selbst aufrufen
- Langweilige Beispiele: Fakultät, Fibonacci
- Coole Beispiele: Folien Rekursion

4. WIEDERHOLUNG BOOLESCHE OPERATOREN

Logische Grundoperatoren

Elementar:	NOT	AND	OR																																							
Symbol:																																										
WT:	<table border="1"><tr><td>x</td><td>y</td><td>z</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr></table>	x	y	z	0	1	1	1	0	0	<table border="1"><tr><td>x</td><td>y</td><td>z</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	x	y	z	0	0	0	0	1	0	1	0	0	1	1	1	<table border="1"><tr><td>x</td><td>y</td><td>z</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	x	y	z	0	0	0	0	1	1	1	0	1	1	1	1
x	y	z																																								
0	1	1																																								
1	0	0																																								
x	y	z																																								
0	0	0																																								
0	1	0																																								
1	0	0																																								
1	1	1																																								
x	y	z																																								
0	0	0																																								
0	1	1																																								
1	0	1																																								
1	1	1																																								
Formel:	$z = \bar{x}$	$z = x \cdot y$	$z = x + y$																																							
C/C++:	log. $z = !x$; (bitw.) $z = \sim x$;	$z = x \& \& y$; $z = x \& y$;	$z = x \parallel y$; $z = x \mid y$;																																							

Verschaltet	NAND	NOR	XOR																																													
Symbol:																																																
WT:	<table border="1"><tr><td>x</td><td>y</td><td>z</td></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	x	y	z	0	0	1	0	1	1	1	0	1	1	1	0	<table border="1"><tr><td>x</td><td>y</td><td>z</td></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	x	y	z	0	0	1	0	1	0	1	0	0	1	1	0	<table border="1"><tr><td>x</td><td>y</td><td>z</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	x	y	z	0	0	0	0	1	1	1	0	1	1	1	0
x	y	z																																														
0	0	1																																														
0	1	1																																														
1	0	1																																														
1	1	0																																														
x	y	z																																														
0	0	1																																														
0	1	0																																														
1	0	0																																														
1	1	0																																														
x	y	z																																														
0	0	0																																														
0	1	1																																														
1	0	1																																														
1	1	0																																														
Formel:	$z = \overline{x \cdot y}$	$z = \overline{x + y}$	$z = x \oplus y$																																													
C/C++:	$z = !(x \& \& y)$; $z = \sim(x \& y)$;	$z = !(x \parallel y)$; $z = \sim(x \mid y)$;	$z = x != y$; $z = x \wedge y$;																																													