CARL VON OSSIETZKY universität OLDENBURG

Teil

Imperative Programmierung

Unterrichtseinheit 4

Anweisungen und Programme (Hamster-Modell)

Dr. Dietrich Boles

Gliederung

- Lexikalik
- Token
- Schlüsselwörter
- Bezeichner
- Hamster-Befehle
- Anweisungen
- Hamster-Programme
- Kommentare
- Beispiele
- Codekonventionen
- Programmentwicklung
- Zusammenfassung

Lexikalik

- Java zugrundeliegender Zeichensatz: **Unicode**
- 16-Bit-Zeichensatz (2^{16} Zeichen)
- erste 128 Zeichen: **ASCII** (7-Bit-Zeichensatz)

Möglichst nur ASCII-Zeichen bzw.

Zeichen auf der Tastatur verwenden !!!!!!!!!!!!

Token

- Token: lexikalische Einheiten
  - Symbole: `<, =, <=, ...`
  - Schlüsselwörter: `while, if, ...`
  - Bezeichner: *`Prozedurnamen, Klassennamen, ...`*
  - Literale: `true, 23, 24.5f, "hello world", ...`

- Trennung von Token:
  - Leerzeichen (Blank)
  - Tabulator
  - Zeilenende
  - Zeilenvorschub
  - Seitenvorschub
- Unterscheidung von Groß- und Kleinbuchstaben!

Schlüsselwörter

abstract	assert	boolean	break
byte	case	catch	char
class	const	continue	default
do	double	else	enum
extends	false	final	finally
float	for	goto	if
implements	import	instanceof	int
interface	long	native	new
null	package	private	protected
public	return	short	static
strictfp	super	switch	synchronized
this	throw	throws	transient
true	try	void	volatile
while			

Bezeichner

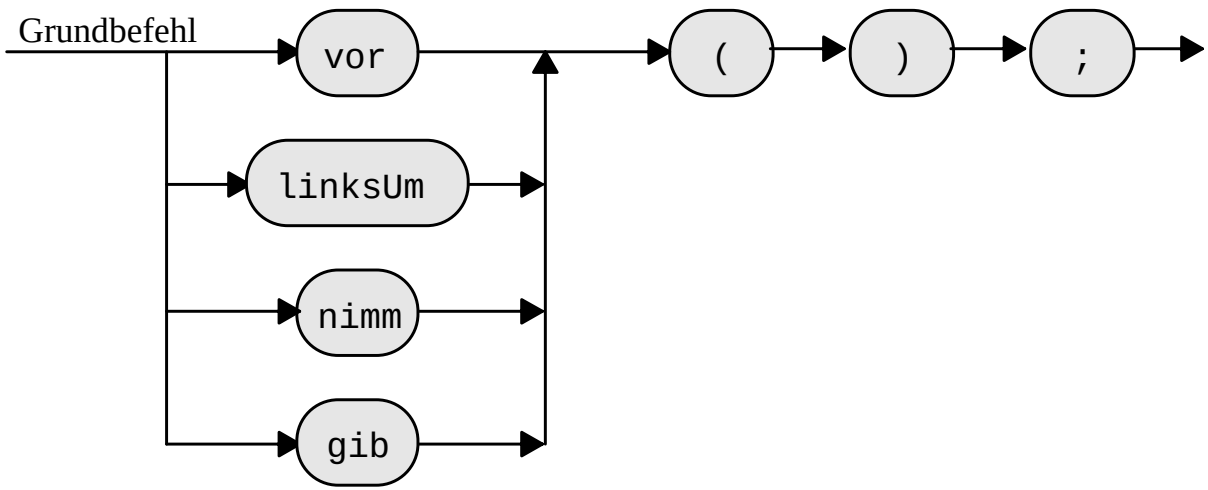
- Benennung von deklarierten Einheiten und Labeln:
  - Klassennamen
  - Variablennamen
  - Prozedurnamen
  - ...

- Beginn mit **Buchstabe**, **Unterstrich (_)** oder **$-Zeichen**
- anschließend: **Buchstaben**, **Ziffern, Unterstriche, $-Zeichen**
- Möglichst keine Umlaute und kein ß verwenden!
- Beispiele:
  - **`Katze`**
  - **`Kaetzchen`**
  - **`_zahl`**
  - **`$more_money$`**

Hamster-Befehle

Vier Grundbefehle:

Syntax:

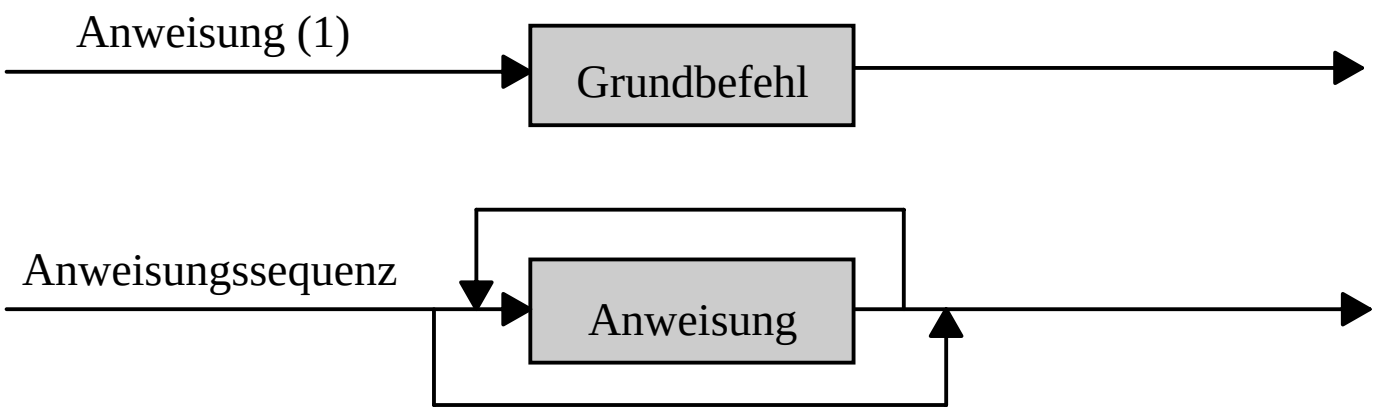


Semantik:

`vor();`	ein Feld nach vorne springen
`linksUm();`	90 Grad nach links drehen
`nimm();`	ein Korn von der aktuellen Kachel aufnehmen
`gib();`	ein Korn aus dem Maul auf der akt. Kachel ablegen

Anweisungen

Syntax:



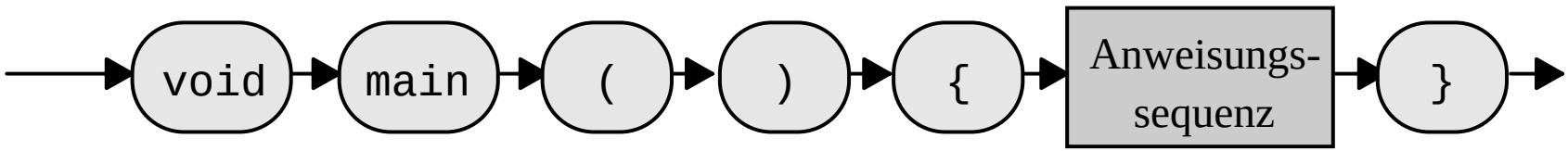
Semantik:

Die Anweisungen der Anweisungssequenz werden nacheinander ausgeführt.

Hamster-Programme

Syntax:

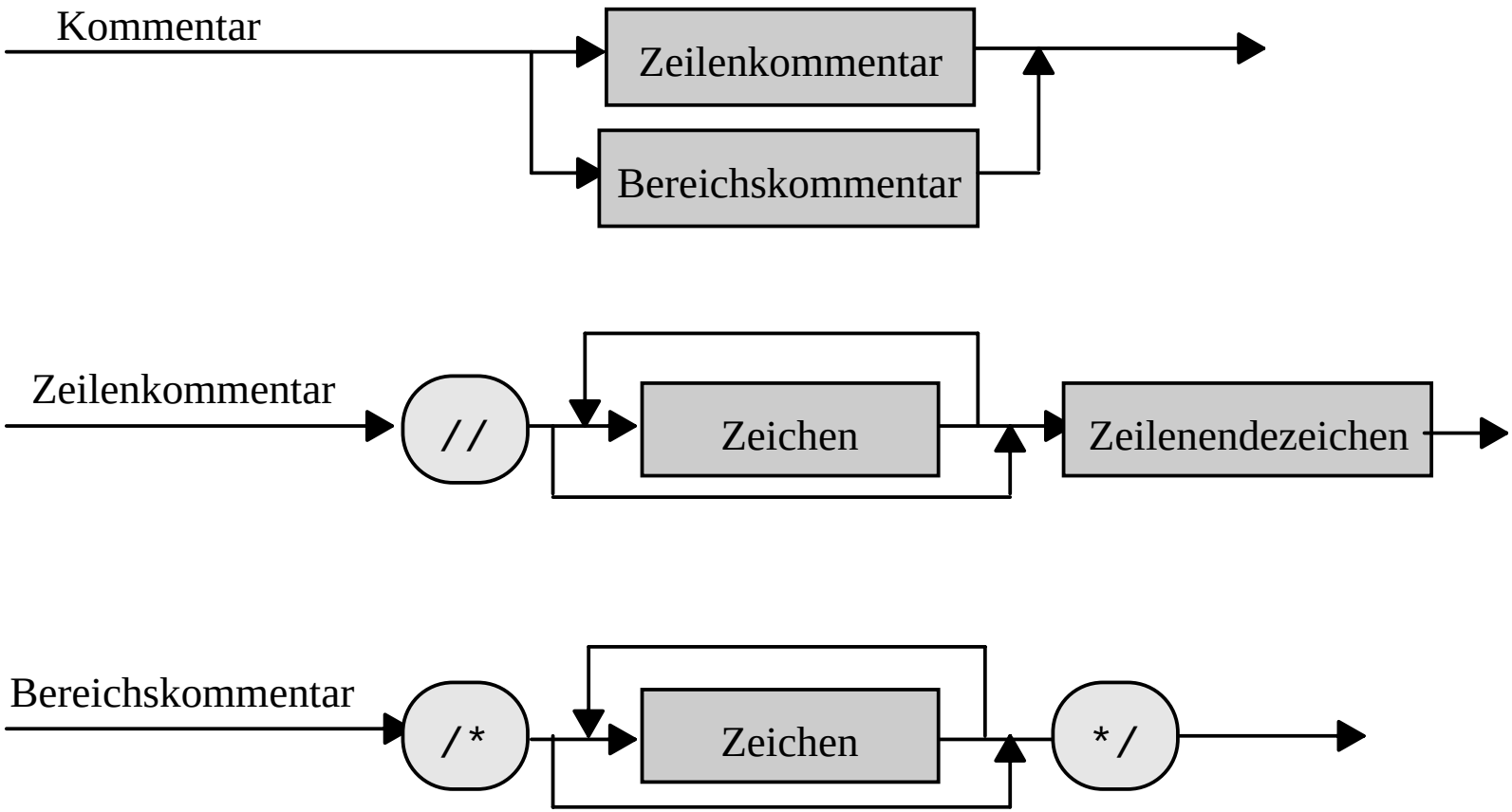
Programm (1)



Semantik:

Beim Aufruf des Programms wird die Anweisungssequenz ausgeführt.

Kommentare


Kommentar
Zeilenkommentar
Bereichskommentar
Zeilenkommentar
//
Zeichen
Zeilenendezeichen
Bereichskommentar
/*
Zeichen
*/

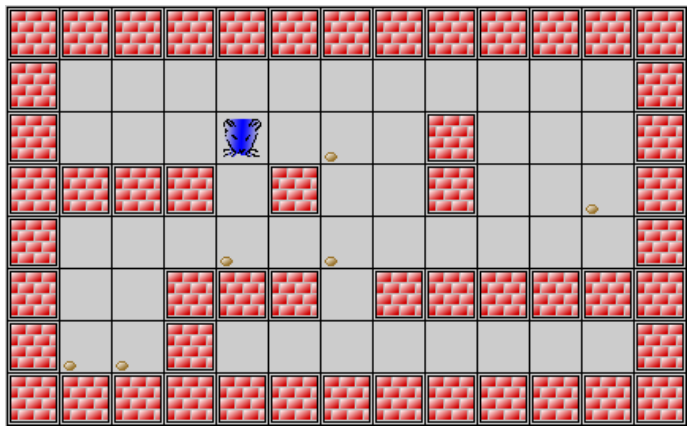
Beispiel 1

Aufgabe:

Gegeben sei das folgende Hamster-Territorium.

Der Hamster soll zwei Körner einsammeln.

Landschaft:



Programm:

```
void main() {
    // friss erstes Korn
    vor(); vor(); nimm();

    /*
     * friss zweites Korn
     */
    linksUm(); vor(); vor(); nimm();
}
```

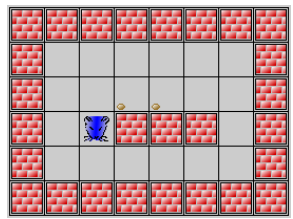
Demo

Beispiel 2

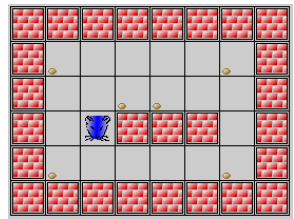
Aufgabe:

Gegeben sei das folgende Territorium.
Der Hamster habe vier Körner im Maul.
Er soll in jeder Ecke eins ablegen und
in seine Ausgangssituation zurückkehren.

Landschaft:



vorher



nachher

Programm:

```
void main() {
    // begib dich an den Rand
    vor(); linksUm();
    // laufe in die rechte untere Ecke
    vor(); vor(); vor(); vor(); gib(); linksUm();
    // laufe in die rechte obere Ecke
    vor(); vor(); vor(); gib(); linksUm();
    // laufe in die linke obere Ecke
    vor(); vor(); vor(); vor(); vor(); gib(); linksUm();
    // laufe in die linke untere Ecke
    vor(); vor(); vor(); gib(); linksUm();
    // begib dich in die Ausgangsposition zurück
    vor(); linksUm(); vor(); linksUm(); linksUm();
}
```

Demo

Codekonventionen

- auf Lesbarkeit des Codes achten
- `void main() {` in eine Zeile
- `}` unterhalb des `v` von `void`
- innere Anweisungen um 4 Spalten einrücken
- pro Zeile **eine** Anweisung
- Leerzeile vor Kommentaren
- Bereichskommentare folgendermaßen strukturieren:

```
/*
 * Kommentar
 */
```

Programmentwicklung

Demo: Hinweise zur Programmentwicklung

Zusammenfassung

- Programme bestehen aus einer Menge an Anweisungen
- Die Anweisungen werden der Reihe nach ausgeführt

- Programme sollen nicht nur korrekt, sondern auch gut verständlich sein:
  - Kommentare: kurze Beschreibung "komplexer" Programmteile
  - Codekonventionen: Einheitlichkeit und Übersichtlichkeit des Sourcecodes