

Einführung in die Programmierung für NF

Algorithmen, Datentypen und Syntax

Ziele

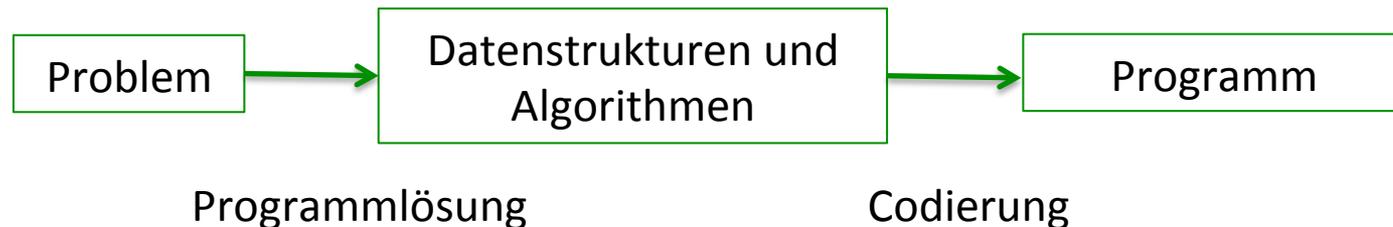
- Begriffsbildungen: Informatik, Algorithmus, Programm, Compiler und andere
- Einordnung von Java
- Ein einfaches Java-Programm erstellen, übersetzen und ausführen
- Java-Programme dokumentieren

Ablauf einer Programmierung

- Programm

Beschreibung von Datenstrukturen und Algorithmen in einer „dem Computer verständlichen“ Sprache (Programmiersprache)

- Programmierung



- Software-Entwicklung

Systematische Konstruktion von Programmen und komplexen Softwaresystemen (-> Systemarchitektur)

Algorithmen

- Ein Algorithmus ist eine aus endlich vielen Schritten bestehende, eindeutige und ausführbare Handlungsvorschrift zur Lösung eines Problems oder einer Klasse von Problemen.
- Alltagsbeispiele

Prozess	Algorithmus	Typische Schritte im Algorithmus
Pullover stricken	Strickmuster	Stricke Rechtsmasche, stricke Linksmasche
Modellflugzeug bauen	Montageanleitung	Leime Teil A an den Flügel B
Kuchen backen	Rezept	Nimm 3 Eier; schaumig schlagen

Eigenschaften von Algorithmen

- Für jede Eingabe gibt es nach endlich vielen Schritten ein Resultat (Terminierung)
- Zu jedem Zeitpunkt der Ausführung gibt es nur eine Möglichkeit der Fortsetzung (Determinismus)
- Bei gleicher Eingabe und unter gleichen Bedingungen gibt es stets dasselbe Ergebnis (Determiniertheit)

Bestandteile von Algorithmen

- Sequenzen (Folge von Anweisungen)
 - Zu einem Zeitpunkt wird nur ein Schritt ausgeführt.
 - Jeder Schritt wird genau einmal ausgeführt: keiner wird wiederholt, keiner wird ausgelassen.
 - Die Schritte werden in der Reihenfolge ausgeführt, in der sie niedergeschrieben sind (d.h. nacheinander).
 - Mit der Beendigung des letzten Schrittes endet der gesamte Algorithmus.

Bestandteile von Algorithmen

- Selektion (Bedingte Anweisung)
 - Einfach: Falls Bedingung dann Sequenz
 - Mit Alternative:
Falls Bedingung dann Sequenz 1 sonst Sequenz 2
 - Pseudocode:

```
if    bedingung
then sequenz1
else sequenz2
```

Bestandteile von Algorithmen

- Iteration (Wiederholung, Schleife)
 - Wiederhole Anweisung(en) bis Bedingung
 - Wiederhole Anweisung(en) eine bestimmte Anzahl oft

- Pseudocode:

```
while bedingung  
do sequenz
```

Pseudocode

- Pseudocode ist eine Möglichkeit, Programme ohne konkrete Programmiersprache anzugeben

- Pseudocodebeispiel:

```
function kochen() {  
    do something;  
}
```

Rezept

Nudeln mit Fertigsoße

250g Nudeln

1 Glas Fertigsoße

1 Prise Salz

0.5l Wasser

Topf mit Wasser füllen und zum Kochen bringen.

Wenn das Wasser kocht Salz hinzufügen und anschließend die Nudeln 12min kochen lassen.

Die Soße ebenfalls in einen Kochtopf geben und diesen mit der Soße langsam 5min erhitzen.

Die Nudeln abgießen. Danach werden die Nudeln mit der Soße auf einem Teller angerichtet.

Rezept als Pseudocode

```
public NudelnMitFertigsosse{  
  
    int nudelnGewicht = 250;  
    int fertigsosse = 1;  
    Int priseSalz = 1;  
    double wasser = 0,5;  
  
    public void kochen (int zutat, int kochtopf, int zeit){  
        fülle zutat in kochtopf;  
        stelle kochtopf auf herd;  
        schalte herd ein;  
        koche bis zeit abgelaufen;  
    }  
  
    public void anrichten (int zutat1, int zutat2, int teller){  
        platziere zutat1 auf teller;  
        platziere zutat2 auf teller;  
    }  
}
```

Klassen

- Java-Programme bestehen aus einer Menge von Klassen
- Eine Klasse besteht aus
 - **Attributen:** Beschreiben den Zustand einer Variablen dieser Klasse
 - **Methoden:** Beschreiben eine Folge von Anweisungen für diese Klasse

DATENTYPEN

Datentypen

- **Ganze Zahlen**

- **byte** 1 Byte -128 bis 127
- **short** 2 Byte -32768 bis 32767
- **int** 4 Byte -2,147,483,648 bis 2,147,483,647
- **long** 8 Byte

- **Gleitkommazahlen**

- **float** 4 Byte Genauigkeit: 7-8 Dezimalstellen
- **double** 8 Byte Genauigkeit: 15-16 Dezimalstellen

Datentypen

Besondere Datentypen: String

- Strings sind Zeichenketten, die aus Buchstaben, Zahlen und Sonderzeichen bestehen können
- Darstellung von Zeichen durch Umrahmung mit Anführungszeichen

Beispiel: "Hallo ihr 4!"

Spezialzeichen: '\n' (Zeilenumbruch)
'\'' (Apostroph)

Datentypen

Besondere Datentypen: Boolean

- Für die Steuerung des Programmablaufs benutzt man Wahrheitswerte.
- Typ **boolean** hat genau zwei Werte: **true** und **false**.
- Vergleichstest auf Zahlen liefern Boolesche Werte als Ergebnis

$a < b$ kleiner

$a > b$ größer

$a \leq b$ kleiner-gleich

$a \geq b$ größer-gleich

$a == b$ gleich

$a != b$ ungleich

OPERATOREN

Operatoren

Operation	Java-Ausdruck	Bedeutung
Negation	!b	„b gilt nicht“
Konjunktion (sequentiell)	b1 && b2	„b1 und b2“
Konjunktion (strikt)	b1 & b2	„b1 und b2“
Disjunktion (sequentiell)	b1 b2	„b1 oder b2“
Disjunktion (strikt)	b1 b2	„b1 oder b2“
Exklusion	b1^b2	„entweder b1 oder b2“

- Bei strikten Operatoren werden zuerst erst beide Argumente ausgewertet und dann der Ausdruck. Wenn ein Argument undefiniert ist, dann ist der ganze Ausdruck undefiniert. Beispiel: `(false & undef) == undef`.
- Bei sequentiellen Operatoren wird von links nach rechts ausgewertet und das zweite Argument wird ignoriert, wenn das Ergebnis nach der Auswertung von b1 schon „klar“ ist. Beispiel: `(false && undef) == false`.

Wahrheitstabellen

- Konjunktion („und“)

false & false == false
false & true == false
true & false == false
true & true == true

&,&&	true	false
true	true	false
false	false	false

- Disjunktion („oder“)

true | false == true
true | true == true

	true	false
true	true	true
false	true	false

- Exklusive Disjunktion („entweder-oder“)

true ^ false == true
true ^ true == false

^	true	false
true	false	false
false	true	false

METHODEN

Methoden

- Eine Methode besteht aus einer **Sequenz von Anweisungen** (Algorithmen)

Rückbezug

– Beispiel: Rezept

Attribute: Zutaten und Mengen

Methoden: kochen, anrichten

Konventionen

- Klassennamen beginnen mit großen Buchstaben
- Methodennamen und Variablennamen beginnen mit kleinen Buchstaben
- Zusammengesetzte Namen werden zusammengeschrieben, jeder (innere) Teilname beginnt mit einem großen Buchstaben

Beispiele: Klasse HalloWelt, Methoden getName, getObject

Syntax

- Alle Anweisungen enden mit einem Strichpunkt
- Für den Moment hat ein Java-Programm für uns immer folgende Form:

```
public class <Klassenname> {  
    public static void main(String[] args) {  
        <Anweisungen>  
    }  
}
```

Deklaration

- Einführung (Deklaration) von Variablen

```
int x = 3;
```

```
int y = 4;
```

- Zuweisung:

```
x = y + 3;
```

(Variablen müssen vorher deklariert sein.)

- Mehrere Anweisungen können hintereinander geschrieben werden.

Methodenaufrufe

- Methodenaufrufe. z.B.:
 - `System.out.println("Hallo!");`
 - `anrichten (nudeln, fertigSosse, teller);`
- Allgemeiner Methodenaufruf
`methodenname (<Parameter>);`

Vielen Dank für Ihre Aufmerksamkeit