

Vorlesung Modellierung und Programmierung I

Inhaltsverzeichnis

Teil I Imperative Programmierung

0	Maschinenzahlen.....	2
0.1	Additionssysteme.....	2
0.2	Positionssysteme.....	2
0.3	Dezimal- und Dualsystem.....	3
0.3.1	Dezimalsystem.....	3
0.3.2	Dualsystem.....	4
0.4	Weitere Beispiele für Positionssysteme.....	6
0.5	Zusammenfassung Zahlendarstellung.....	7
0.6	Maschinenzahlen.....	8
0.6.1	Interne Darstellung von ganzen Zahlen $z \in \mathbb{Z}$	8
0.6.2	Interne Darstellung von rationalen Zahlen $z \in \mathbb{Q}$	9
0.7	IEEE - Standard.....	11
1	Algorithmen und Programme.....	1-2
1.1	Algorithmus.....	1-3
1.2	Programm.....	1-4
1.2.1	Hauptkomponenten des Aufbaus eines Rechners.....	1-4
1.2.2	Ein Programm für einen Algorithmus.....	1-6
1.2.3	Rekursion.....	1-7
1.2.4	Codierung.....	1-8
1.3	Programmierparadigmen.....	1-9
1.4	Anhang zum Modellrechner.....	1-11
1.4.1	Backus-Naur-Form (BNF).....	1-11
1.4.2	Syntaxdiagramm.....	1-12
2	Einstieg in das Programmieren mit Java.....	2-2
2.1	Geschichte.....	2-2
2.2	Konzept.....	2-3
2.2.1	Compiler und Interpreter.....	2-3
2.2.2	Bytecode und virtuelle Maschinen.....	2-4
2.3	Grundlagen der Java-Programmierung.....	2-6
2.3.1	Installation von Java.....	2-6
2.3.2	Java Application.....	2-6
2.3.3	Java Applet.....	2-7
2.3.4	Java Script.....	2-9
2.4	Ein- und Ausgaben.....	2-12
2.4.1	Ausgabe.....	2-12
2.4.2	Eingabe.....	2-12
2.4.3	Installation des Pakets <code>Tools</code>	2-14
2.4.4	Methoden der Klasse <code>IOTools</code>	2-14

3	Grundelemente der Java-Programmierung	3-2
3.1	<i>Alphabet</i>	3-2
3.2	<i>Bezeichner</i>	3-3
3.3	<i>Kommentare</i>	3-3
3.4	<i>Elementardatentypen</i>	3-5
3.5	<i>Konstanten (Literele)</i>	3-7
3.5.1	Unbenannte Konstanten	3-7
3.5.2	Benannte Konstanten	3-8
3.6	<i>Variablen</i>	3-9
3.7	<i>Ausdrücke</i>	3-11
3.8	<i>Zusammenfassung</i>	3-13
4	Anweisungen	4-2
4.1	<i>Strukturierte Programmierung</i>	4-2
4.1.1	Geschichte	4-2
4.1.2	Strukturierung im Kleinen	4-2
4.2	<i>Einige Beispielanwendungen</i>	4-4
4.2.1	Addierer (<i>do</i> -Schleife)	4-4
4.2.2	Ein- Mal- Eins (<i>for</i> -Schleife, <i>if</i> -Anweisung)	4-5
4.2.3	Linaere Gleichung (<i>if-else</i> -Anweisung)	4-6
4.2.4	Einfacher Rechner (<i>switch</i> -Anweisung)	4-8
4.2.5	Summieren (<i>while</i> -Schleife, <i>break</i> , <i>continue</i>)	4-10
5	Referenzdatentypen - Felder	5-2
5.1	<i>Eindimensionale Felder - Vektoren</i>	5-3
5.1.1	Vereinbarung	5-3
5.1.2	Referenzen sind keine Felder	5-4
5.1.3	Kopieren eindimensionaler Felder	5-6
5.1.4	Beispiel „Sortieren eines Vektors“	5-6
5.2	<i>Zweidimensionale Felder - Matrizen</i>	5-8
5.2.1	Vereinbarung	5-8
5.2.2	Kopieren zweidimensionaler Felder	5-9
5.2.3	Beispiel „Tic Tac Toe“	5-9
6	Referenzdatentypen - Klassen	6-2
6.1	<i>Deklaration und Instanziierung</i>	6-3
6.1.1	Festlegen eines Strukturtyps als Klasse	6-3
6.1.2	Festlegen einer Struktur als Objekt einer Klasse	6-4
6.1.3	Kopieren von Objekten	6-7
6.2	<i>Felder von Klassen</i>	6-8
6.3	<i>Klassen von Klassen</i>	6-11

7	Methoden	7-2
7.1	Prozedurorientierte Programmierung	7-2
7.2	Klassenmethoden.....	7-5
7.2.1	Methodenvereinbarung	7-8
7.2.2	Methodenaufruf.....	7-9
7.2.3	Referenzdatentypen in der Schnittstelle einer Methode	7-10
7.2.4	Beispiel „Polynomwertberechnung mittels Horner Schema“	7-10
7.2.5	Überladen von Methoden.....	7-14
7.3	Die Kommandozeile	7-15
7.4	Rekursionen.....	7-16
7.4.1	Beispiel „Eine unendliche Geschichte“	7-16
7.4.2	Beispiel „Türme von Hanoi“	7-17
7.4.3	Iteration oder Rekursion	7-19
7.5	Konstanten und Methoden der Klasse <code>java.lang.Math</code>	7-20

Anhänge zu Teil I

Einführung in HTML

Java-Ausdrücke

Edsger Wybe Dijkstra

Java-Anweisungen

Methoden der Klasse `java.lang.Math`

Teil II Objektorientierte Programmierung

8	Das objektorientierte Programmierparadigma	8-2
8.1	<i>Strukturierung im Großen - Modularisierung</i>	8-2
8.2	<i>Modularisierung</i>	8-5
8.2.1	Objekte zur Kapselung von Attributen und Methoden	8-5
8.2.2	Klassen zur Datenabstraktion	8-6
8.3	<i>Vererbung, Generalisierung und Spezialisierung</i>	8-9
8.3.1	Abgeleitete Klassen - Spezialisierung	8-9
8.3.2	Überschreiben von Methoden	8-10
8.3.3	Abstrakte Klassen - Generalisierung	8-11
8.3.4	Einfach- und Mehrfachvererbung	8-13
8.4	<i>Polymorphismus und dynamisches Binden</i>	8-15
8.5	<i>Zusammenfassung - Konzepte der OOP</i>	8-17
8.6	<i>Methoden der Klasse <code>java.lang.String</code></i>	8-18
9	Klassen	9-2
9.1	<i>Instanzvariablen und Instanzmethoden</i>	9-2
9.1.1	Vereinbarung	9-2
9.1.2	<code>this</code> - Selbstreferenz	9-5
9.1.3	<code>toString()</code>	9-5
9.2	<i>Klassenvariablen und Klassenmethoden</i>	9-7
9.2.1	Vereinbarung	9-7
9.2.2	<code>super</code> - Superklassenreferenz	9-10
9.3	<i>Klassenkonstanten</i>	9-11
9.4	<i>Instanziierung und Initialisierung</i>	9-13
9.4.1	Konstruktoren	9-13
9.4.2	Beispiel „Boot“	9-15
9.4.3	Der Mechanismus der Objekterzeugung	9-22
9.5	<i>Pakete</i>	9-23
9.5.1	Erstellen von Paketen	9-23
9.5.2	Installation komprimierter Pakete	9-24
9.6	<i>Zugriffsrechte</i>	9-25
10	Modellierung	10-2
10.1	<i>Entwurfs- und Zerlegungstechniken</i>	10-3
10.1.1	Zielgerichteter Entwurf	10-3
10.1.2	Kompositioneller Entwurf	10-3
10.1.3	Objektorientierter Entwurf	10-3
10.2	<i>Klassenbeziehungen</i>	10-5
10.2.1	UML (Unified Modeling Language)	10-5
10.2.2	Assoziation, Aggregation und Komposition	10-5
10.2.3	Entwurfsmuster	10-7
10.3	<i>Modulbildung</i>	10-8
10.4	<i>Beispiel „Milchladen“</i>	10-11
10.5	<i>Vor- und Nachteile objektorientierter Programmierung</i>	10-21

11	Dynamische Verwaltung großer Datenmengen	11-2
11.1	<i>Einige spezielle Klassen des Paketes java.lang.*</i>	11-2
11.1.1	Klasse Throwable, Exception und Error	11-2
11.1.2	Klasse Object	11-7
11.1.3	Wrapper-Klassen	11-8
11.2	<i>Abstrakte Klassen und Interfaces</i>	11-9
11.2.1	Abstrakte Klassen	11-9
11.2.2	Interfaces	11-10
11.3	<i>Collection-Klassen java.util.*</i>	11-12
11.3.1	Generische Datentypen - Template	11-13
11.3.2	Interface Collection<E>	11-13
11.3.3	Interface Iterator<E>	11-14
11.3.4	Interface Set<E>	11-15
11.3.5	Interface List<E>	11-16
11.4	<i>Klassen Arrays und Collections</i>	11-23
11.5	<i>Zusammenfassung</i>	11-23
12	Grafische Benutzerschnittstellen	12-2
12.1	<i>Komponenten</i>	12-2
12.1.1	AWT und Swing	12-2
12.1.2	Beispiel „Hallo Welt!“	12-3
12.2	<i>Ereignisverarbeitung</i>	12-6
12.3	<i>Grafik</i>	12-10
12.3.1	Die Methoden paint und repaint	12-10
12.3.2	Die abstrakte Klasse java.awt.Graphics	12-10
12.3.3	Beispiel „Simple Grafik“	12-11
13	MVC-Architektur	13-2
13.1	<i>Das MVC-Konzept</i>	13-2
13.1.1	Entwurfsmuster	13-2
13.1.2	Mechanismus der MVC-Architektur	13-3
13.1.3	Klassen java.util.Observable und java.util.Observer	13-3
13.2	<i>Beispiel „Zähler modulo 10“</i>	13-10
13.2.1	Design und Implementierung des Modells	13-10
13.2.2	Design und Implementierung des Views	13-11
13.2.3	Design und Implementierung des Controllers	13-13
13.2.4	Implementierung des Programmaufbaus	13-15
13.2.5	Beispiel „Zähler modulo 10“ mit grafischer Oberfläche	13-17
13.3	<i>Steuerung mehrerer View und Controller zu einem Model</i>	13-23
13.3.1	Abstrakte Klassen	13-24
13.3.2	Abgeleitete Klassen	13-28
13.3.3	Beispiel „Ampelsteuerung“	13-31
13.4	<i>Zusammenfassung</i>	13-34

14	Dateiverwaltung, das Stream-Konzept	14-2
14.1	Datenströme	14-2
14.1.1	Datenströme in Java, Paket <code>java.io.*</code>	14-3
14.1.2	Standarddatenströme	14-4
14.2	Klasse <code>java.io.File</code>	14-7
14.3	Textdateien	14-9
14.3.1	Ungepufferte Reader- und Writer- Klassen	14-10
14.3.2	Gepufferte Reader- und Writer- Klassen	14-12
14.3.3	Beispiel „TextDatei“	14-14
14.4	Daten- und Objektdateien	14-20
14.4.1	Klassen <code>InputStream</code> und <code>OutputStream</code>	14-20
14.4.2	Dateien für Elementardatentypen	14-23
14.4.3	Dateien für Elementar- und Referenzdatentypen	14-27
14.4.4	Beispiel „Obst“	14-30
14.5	Übersicht über häufig verwendete Datenströme	14-36
14.6	Weitere wichtige Klassen	14-38
15	Parallele Programmierung	15-2
15.1	Die Klasse <code>java.lang.Thread</code>	15-2
15.2	Beispiel „0-1-Printer“ als <code>Thread</code>	15-3
15.3	Das Interface <code>java.lang.Runnable</code>	15-4
15.4	Beispiel „0-1-Printer“ mit <code>Runnable</code>	15-5
15.5	Zusammenfassung	15-6
16	Rechnerkommunikation – verteilte Systeme	16-2
16.1	Netzwerktechnologie	16-2
16.1.1	Adressen	16-2
16.1.2	Ports und Sockets	16-3
16.2	Server/Client-Programmierung	16-5
16.2.1	Aufbau einer Server/Client-Verbindung	16-5
16.2.2	Die Klassen <code>java.net.ServerSocket</code> , <code>java.net.Socket</code>	16-7
16.3	Beispiel „Chatroom“	16-9
16.3.1	Modell	16-9
16.3.2	Server	16-10
16.3.3	Client	16-15

Anhänge zu Teil II

Methoden der Klasse `java.lang.String`

Übersicht über häufig verwendete Datenströme