

# Inhaltsverzeichnis

<b>1 Einige Grundbegriffe der Informatik</b>	<b>1</b>
<b>2 Konstruktion von Algorithmen</b>	<b>7</b>
2.1 Fallbeispiel Währungsumrechnung . . . . .	7
2.2 Das erste Smalltalk-Programm . . . . .	8
2.2.1 Eingabe von Programmtexten . . . . .	9
2.2.2 Ausführung von Programmen . . . . .	11
2.2.3 Flexibilisierung der Währungsumrechnung . . . . .	14
2.3 Beispiel: Lösung einer quadratischen Gleichung . . . . .	17
2.3.1 Der Algorithmus . . . . .	17
2.3.2 Das Programm . . . . .	19
2.3.3 Verallgemeinerung der Lösung quadratischer Gleichungen . . . . .	20
2.4 Zusammenfassung . . . . .	27
<b>3 Grundzüge der objektorientierten Programmierung mit Smalltalk</b>	<b>29</b>
3.1 Objekte, Nachrichten, Methoden . . . . .	30
3.1.1 Nachrichten . . . . .	34
3.1.2 Fallunterscheidung . . . . .	37
3.1.3 Blöcke . . . . .	37
3.1.4 Erzeugung von Objekten – Klassen . . . . .	38
3.2 Literale . . . . .	43
3.3 Variablen und Zuweisungen . . . . .	47
3.4 Verweissemantik . . . . .	51
3.4.1 Nutzung des Object Explorers . . . . .	54
<b>4 Wiederholungen</b>	<b>57</b>
4.1 Maximumssuche . . . . .	58
4.2 Weitere Smalltalk-Nachrichten für Wiederholungen . . . . .	70
4.2.1 Zählschleifen . . . . .	70
4.2.2 Intervalldurchlauf . . . . .	71
4.2.3 Behälterdurchlauf . . . . .	73
<b>5 Die Entwicklungsumgebung VisualWorks</b>	<b>75</b>
5.1 Überblick . . . . .	75
5.2 Starten der Entwicklungsumgebung . . . . .	77
5.3 Launcher mit Transcript . . . . .	77
5.3.1 Anlegen eines eigenen Images . . . . .	79
5.3.2 Einstellen von Systemparametern . . . . .	79
5.3.3 Benutzung des Transcripts . . . . .	82

5.4	Workspace	83
5.5	Inspector	85
5.6	Debugger	86
5.7	System-Browser	87
<b>6</b>	<b>Betrachten einer Beispielklasse: Circle</b>	<b>95</b>
6.1	Klassenhierarchien und Vererbung	95
6.2	Methodenimplementierungen	99
6.3	Alternativimplementierung der Klasse Circle	106
<b>7</b>	<b>Definition neuer Klassen</b>	<b>111</b>
7.1	Fallbeispiel Währungsumrechnung	111
7.1.1	Anlegen einer neuen Klasse	112
7.1.2	Eigene Klassenmethoden für die Exemplarerzeugung	118
7.1.3	Definition von Exemplarmethoden	122
7.1.4	Erweiterung des Umrechners	125
7.2	Fallbeispiel Kinobetrieb	127
7.2.1	Analyse der Problembeschreibung	127
7.2.2	Implementierung	130
7.3	Definition von Klassenvariablen	135
<b>8</b>	<b>Klassenhierarchien – Vererbung – Polymorphie</b>	<b>139</b>
8.1	Die Smalltalk-Klassenhierarchie	139
8.1.1	Aufbau	139
8.1.2	Das Zahlen-Konzept von Smalltalk	141
8.1.3	Die Integer-Klassen	146
8.1.4	Die Klassen Float, Double und Fraction	152
8.1.5	Gemeinsame Methoden der Zahlenklassen	153
8.1.6	Gemischte Ausdrücke	158
8.1.7	Die Wahrheitswerte	159
8.1.8	Zeichen und Zeichenketten	162
8.1.9	Datum und Zeit	166
8.2	Abstrakte, konkrete Klassen	170
8.3	Generische Methoden	173
8.4	Polymorphie	175
8.5	Fallbeispiel Quadratische Gleichungen	176
8.5.1	Die Klasse QuadratGleichng	178
8.5.2	Klassen für Lösungsobjekte	180
8.5.3	Die Lösungsmethoden	184
8.5.4	Anwendungsbeispiele	186
8.5.5	Nutzung von Vererbung und Polymorphie	187
8.5.6	Testprogramme als Klassenmethoden	187
8.5.7	Fehlerbehandlung	189
<b>9</b>	<b>Fehlersuche in Smalltalk-Programmen</b>	<b>193</b>
9.1	Syntaxfehler	193
9.2	Unbekannte Variablen	194

9.3	Unbekannte Nachrichtenselektoren . . . . .	195
9.4	Laufzeitfehler . . . . .	195
9.5	Methoden-Debugging . . . . .	197
<b>10</b>	<b>Objektbehälter</b>	<b>205</b>
10.1	Ungeordnete Behälter . . . . .	206
10.1.1	Die Klasse <code>Set</code> . . . . .	206
10.1.2	Die Klasse <code>Bag</code> . . . . .	209
10.1.3	Die Klasse <code>Dictionary</code> . . . . .	211
10.2	Geordnete Behälter . . . . .	216
10.2.1	Die Klasse <code>Array</code> . . . . .	218
10.2.2	Die Klasse <code>OrderedCollection</code> . . . . .	219
10.2.3	Die Klasse <code>SortedCollection</code> . . . . .	222
10.2.4	Die Klasse <code>Interval</code> . . . . .	224
10.2.5	Die Klasse <code>String</code> . . . . .	226
10.2.6	Die Klasse <code>Symbol</code> . . . . .	226
10.3	Umwandlung von Behältern . . . . .	226
10.4	Fallbeispiel Kinobetrieb . . . . .	227
10.4.1	Aufgabe: Anzeige aller Kinos im Transcript . . . . .	228
10.4.2	Aufgabe: Anzeige des Profits eines besimmtten Kinos . . . . .	234
10.4.3	Aufgabe: Anzeige aller Kinos der Größe nach geordnet . . . . .	236
<b>11</b>	<b>Weitere Smalltalk-Grundlagen</b>	<b>239</b>
11.1	Blöcke . . . . .	239
11.1.1	Blöcke als Objekte . . . . .	239
11.1.2	Blöcke mit Parametern . . . . .	241
11.1.3	Anwendungen . . . . .	243
11.1.4	Fallbeispiel Endlicher Automat . . . . .	245
11.2	Vererbung – Methodensuche . . . . .	252
11.2.1	Regeln für Methodensuche . . . . .	254
11.2.2	Bedeutung der Pseudovariablen <code>self</code> und <code>super</code> . . . . .	255
11.3	Metaklassen . . . . .	257
11.4	Objektidentität . . . . .	261
11.4.1	Gleichheit versus Identität . . . . .	262
11.4.2	Gleichheit von Objekten selbst definierter Klassen . . . . .	265
11.4.3	Objektkopien . . . . .	268
<b>12</b>	<b>Algorithmischer Exkurs: Rekursion</b>	<b>273</b>
12.1	Rekursive Algorithmen . . . . .	274
12.2	Korrekttheit von rekursiven Algorithmen . . . . .	280
12.3	Rekursive Denkweise . . . . .	282
12.4	Unendliche Strukturen . . . . .	283
<b>13</b>	<b>Datenströme und Dateien</b>	<b>287</b>
13.1	Sequentieller Zugriff auf geordnete Behälter . . . . .	287
13.2	Sequentieller Zugriff auf Dateien . . . . .	291

<b>14 Gestaltung von Smalltalk-Programmen</b>	<b>295</b>
14.1 Standard-Methodenprotokolle	296
14.2 Das <code>printOn:</code> -Framework	298
14.3 Auslagerung von Teilalgorithmen in eigenständige Methoden	303
14.4 Benutzungsoberflächen – Das Model-View-Controller-Paradigma	308
14.5 Beziehungen zwischen Klassen	312
14.5.1 Vererbung	312
14.5.2 Assoziation	315
14.5.3 Aggregation	317
<b>15 Systematisches Testen</b>	<b>319</b>
15.1 Komponententests	319
15.2 Testautomatisierung mit SUnit	321
15.2.1 Fallbeispiel Kinobetrieb	321
15.2.2 Weitere <code>TestCase</code> -Nachrichten	325
15.2.3 Ein weiterer Test für die Klasse <code>Kosten</code>	325
15.3 Testgetriebene Entwicklung	326
<b>16 Entwicklung von Web-Anwendungen</b>	<b>329</b>
16.1 Fallbeispiel Währungsumrechner	330
16.1.1 Zielvorstellung	331
16.1.2 Das Model des Währungsumrechners	331
16.1.3 Ein erster Blick auf Seaside	334
16.1.4 Realisierung der Web-Oberfläche	335
16.1.5 Implementierung der Funktionalität	347
16.1.6 Nachteile der ersten Version	350
16.1.7 Refactoring der Methode <code>renderFormularOn:</code>	350
16.2 Verbesserung der Bedienbarkeit des Währungsumrechners	352
16.2.1 Auswahllisten in Seaside	353
16.2.2 Prüfung der Betragseingabe	354
16.3 Einführung eines Verwaltungsdialogs	357
16.3.1 Anlegen der Komponente <code>WuVerwaltung</code>	358
16.3.2 Aufruf der Komponente <code>WuVerwaltung</code>	359
16.3.3 Der Call/Answer-Mechanismus von Seaside	360
16.3.4 Implementierung des Pflegedialogs	363
16.4 Einbindung von CSS	368
16.4.1 Verknüpfung von HTML und CSS	370
16.4.2 Definition einer <code>style</code> -Methode	372
16.4.3 Bereitstellung einer CSS-Datei in einer Seaside-File-Library	373
16.5 Verallgemeinerung des Umrechners	375
16.6 Weitere Konzepte von Seaside	379
<b>17 Wie geht es weiter?</b>	<b>381</b>
<b>A Erweiterung des VisualWorks-Images</b>	<b>387</b>
A.1 Einspielen von SUnitToo	388
A.2 Einspielen des Object Explorers	389

<b>A.3 Einspielen von Seaside . . . . .</b>	<b>389</b>
<b>A.4 Das Cincom Public Repository . . . . .</b>	<b>390</b>
<b>Literaturverzeichnis</b>	<b>393</b>
<b>Tabellenverzeichnis</b>	<b>397</b>
<b>Index</b>	<b>399</b>