

Inhaltsverzeichnis

Vorwort	3
0 Ausgangslage	5
1 Rekursive Datenstrukturen	6
1.1 Listen (ca. 29 Std.)	6
1.1.1 Warteschlange	6
1.1.1.1 Grundprinzip	7
1.1.1.2 Implementierung mit festen Referenzen	9
1.1.1.3 Implementierung mit einem Feld	10
1.1.1.4 Grenzen der Implementierung mit einem Feld	13
1.1.2 Grundprinzip der Liste und rekursive Abläufe	13
1.1.2.1 Grundprinzip der verketteten Liste	13
1.1.2.2 Rekursive Abläufe	14
1.1.2.3 Entfernen und Einfügen bei einer Warteschlange	16
1.1.3 Die einfach verkettete Liste	20
1.1.3.1 Von der Warteschlange zur Liste	20
1.1.3.2 Die Klassen KNOTEN und LISTE (erste Version)	21
1.1.3.3 Methoden der Klasse KNOTEN	24
1.1.3.4 Methoden der Klasse LISTE	28
1.1.3.5 Implementierungen von KNOTEN und LISTE	30
1.1.3.6 Klasse LISTENELEMENT – Entwurfsmuster Kompositum	33
1.1.3.7 Erweiterungen der Klasse ‚einfach verkettete Liste‘	42
1.1.4 Stapel und Schlange als spezielle Formen der Liste	58
1.1.4.1 Stapel	58
1.1.4.2 Implementierung der Klasse STAPEL	60
1.1.4.3 Implementierung der Klasse SCHLANGE	61
1.2 Bäume (ca. 16 Std.)	62
1.2.1 Effizienzsteigerung durch bessere Struktur	62
1.2.1.1 Der Weg zum Baum	62
1.2.1.2 Struktur von Bäumen	64
1.2.1.3 Suchen im geordneten Binärbaum	65
1.2.1.4 Erste Implementierung des geordneten Binärbaums	66
1.2.2 Schrittweise Implementierung eines typischen Baums	67
1.2.2.1 Die Methode Einfuegen	67
1.2.2.2 Baum und Kompositum	68
1.2.2.3 Baumdurchlauf	70
1.2.3 Anwendungen und Vertiefungen	72
1.2.3.1 Beispiel Wörterbuch	72
1.2.3.2 Entfernen von Knoten	73
1.3 Graphen (ca. 13 Std.)	76
1.3.1 Die Datenstruktur Graph	76
1.3.1.1 Anwendungen und Definition	76
1.3.1.2 Die Struktur des Graphen als Klassendiagramm	79
1.3.1.3 Die Datenstruktur Graph als Verallgemeinerung des Baumes	80
1.3.2 Umsetzung in eine Programmiersprache	81
1.3.2.1 Adjazenzmatrix	81

1.3.2.2 Die Klassen DATENELEMENT, KNOTEN und GRAPH_MATRIX	83
1.3.3 Verfahren für den Graphendurchlauf	86
1.3.4 Klassendiagramm der Klasse GRAPH_MATRIX	92
1.3.5 Ausblick	93
2 Softwaretechnik (ca. 26 Std.)	94
2.1 Planung und Durchführung kooperativer Arbeitsabläufe	94
2.1.1 Planungsgrundlagen	94
2.1.2 Diagramme und ihre Verwendung	95
2.1.3 Entwurfsmuster	101
2.2 Praktische Softwareentwicklung	109
2.2.1 Durchführung des Projekts	109
2.2.2 Anforderungen an Projektthemen	110
2.2.3 Datenbankankbindung	111
3 Anwendungsorientierte Aufgaben	116
3.1. IT-Kompetent	116
3.1.1 Aufgabenstellung	116
3.1.1.1 IT-Kompetent – Grundaufgabe	117
3.1.1.2 IT-Kompetent – Verwaltung der Kundendaten (Erweiterung von 3.1.1.1) ...	117
3.1.1.3 IT-Kompetent – Überblick durch Sortierung (Erweiterung von 3.1.1.2).....	118
3.1.1.4 IT-Kompetent – jetzt effektiver speichern (Erweiterung von 3.1.1.2 oder 3.1.1.3).....	118
3.1.2 Lösungshinweise und Lösungsvorschläge	119
3.1.2.1 Lösungshinweise: IT-Kompetent – Grundaufgabe	119
3.1.2.2 Lösungshinweise: IT-Kompetent – Verwaltung der Kundendaten	120
3.1.2.3 Lösungshinweise: IT-Kompetent – Überblick durch Sortierung	120
3.1.2.4 Lösungshinweise: IT-Kompetent – jetzt effektiver speichern	121
3.2. Rangierbahnhof	121
3.2.1 Aufgabenstellung	121
3.2.1.1 Rangierbahnhof – Situation aus dem Kapitel Listen	121
3.2.1.2 Rangierbahnhof – Abgeänderte Situation	122
3.2.2 Lösungshinweise und Lösungsvorschläge	123
3.2.2.1 Lösungshinweise: Rangierbahnhof – Situation aus dem Kapitel Listen	123
3.2.2.2 Lösungshinweise: Rangierbahnhof – Abgeänderte Situation	125
3.3 Postfixnotation	126
3.3.1 Aufgabenstellung	126
3.3.1.1 Überführung der Infix- in die Postfixnotation	127
3.3.1.2 Vertiefung 1 – Berücksichtigung klammerloser Ausdrücke, also z. B. Beachtung von der „Punkt vor Strich“-Regel	129
3.3.1.3 Vertiefung 2 – Berechnung von Termen	130
3.3.1.4 Vertiefung 3 – Rechenbaum	131
3.3.3 Lösungshinweise und Lösungsvorschläge	132
3.3.3.1 Lösungshinweise: Überführung der Infix- in die Postfix-Notation	132
3.3.3.2 Lösungshinweise zu Vertiefung 1	133
3.3.3.3 Lösungshinweise zu Vertiefung 2	134
3.3.3.4 Lösungshinweise zu Vertiefung 3	134