

Inhaltsverzeichnis

I: Information
A: Aufgaben
L: Lösungen

I A L

Hinweise zum Selbststudium	1
1 Einführung	3
2 Methode der Strukturierten Programmierung	6
2.1 Ziele der Strukturierten Programmierung	8
2.2 Schrittweise Verfeinerung	11
2.3 Beschränkung der Strukturblockkarten	19
2.4 Blockkonzept	23
2.5 Lesbarkeit	27
2.6 Datenunterscheidung	32
3 Darstellungsmittel für die Strukturierte Programmierung	36
3.1 Baumdiagramme	37
3.2 Struktogramme	40
3.3 Verbale Entwurfssprache	46
3.4 DIN-Normen	48
4 Beispiele und Übungen zu Struktogrammen	50
4.1 Praktische Hinweise	50
4.2 Einfache Strukturblöcke	54
4.3 Zusammengesetzte Strukturblöcke	56
5 Umsetzung des Entwurfs in Primärkode	58
5.1 Implementierung mit COBOL74	59
5.2 Implementierung mit COBOL85	74
5.3 Implementierung mit Assembler	89
5.4 Implementierung mit FORTRAN77	98

I A L

6 Beispiele und Übungen zur Strukturierten Programmierung	109
6.1 Lineare Verarbeitung	110 248
6.2 Mischen	112 260
Erweiterung	115 266
6.3 Gruppenwechsel	118 275
6.4 Tabellenverarbeitung	122
Einlesen und Aufbauen einer Tabelle	122 282
Verarbeiten von Tabellendaten	124 283
Sortieren von Tabellenelementen	129 286
7 Strukturiert programmieren im System COLUMBUS-COBOL	132
7.1 COLUMBUS-COBOL-Quellprogramm	135
7.2 COLUMBUS-COBOL-Strukturblöcke	138
7.3 COLUMBUS-COBOL-Prozeduren	146
7.4 COLUMBUS-Dienstprogramme (COBOL)	153
7.5 COLUMBUS-COBOL-ET-Prozedur	162
7.6 COLUMBUS-COBOL-Optimierung	168
8 Strukturiert programmieren im System COLUMBUS-Assembler	171
8.1 COLUMBUS-Assembler-Quellprogramm	173
8.2 COLUMBUS-Assembler-Strukturblöcke	177
8.3 COLUMBUS-Assembler-Bedingungen	188
8.4 COLUMBUS-Assembler-Prozeduren	192
8.5 COLUMBUS-Assembler-Datenkonzept	199
8.6 COLUMBUS-Dienstprogramme (Assembler)	209
9 Strukturierte Programmierung und Software-Entwicklung	216
9.1 DV-Grobkonzept	218
9.2 DV-Feinkonzept	227
9.3 Implementierung und Test	231
9.4 Reverse Engineering	234
Lösungen zu den Aufgaben	236
Sachverzeichnis	288