

# Inhaltsverzeichnis

<b>Grundlagen-Test</b>	1
<b>1. Einführung</b>	3
1.1. Notwendigkeit der Programmiersprache	3
1.2. Programm, Befehl und Arbeitsspeicher	3
1.3. Entwicklung des Assemblerbefehls MVC	6
1.4. Allgemeines zur Übersetzung Assemblersprache – Maschinen- sprache	8
<b>2. Programmentstehung</b>	9
2.1. Konstanten- und Speicherbereichsdefinitionen	9
2.1.1. Definition von Konstanten, DC	9
2.1.2. Definition von Speicherbereichen, DS	10
2.2. Längenangaben	12
2.2.1. Implizite Länge	12
2.2.2. Explizite Länge	16
2.3. Variationen bei der Definition von Konstanten	19
2.4. Assembleranweisungen	20
2.4.1. Die START-Anweisung	20
2.4.2. Die END-Anweisung	21
2.5. Übersetzung des Primärprogramms	24
2.5.1. Befehlsformat	24
2.5.2. Adreßpegel und Adreßbuch	26
<b>Übungen zu den Kapiteln 1 und 2</b>	29
<b>3. Stufen zum Programmlauf</b>	33
3.1. Erfassung der Programmdaten	33
3.1.1. Das Assemblerformular	33
3.1.2. Assemblerprogramm auf Datenträgern	38
3.2. Übersetzungsvorgang, das Modul	39
3.3. Modul – Phase	40
3.4. Befehlszähler und Startadresse	41
<b>4. Makroaufrufe</b>	44
4.1. Ein/Ausgabe	44
4.2. Das Logische Ein/Ausgabesystem, EAS (IOCS)	44
4.2.1. Die Makroaufrufe GET und PUT	45
4.2.2. Der Makroaufruf EOJ	51
4.2.3. Auflösung der Makroaufrufe bei der Übersetzung	52

<b>5. Vergleichs- und Sprungbefehle</b>	<b>54</b>	
5.1. Allgemeine Betrachtung	54	
5.2. Programmablaufplan	54	
5.3. Programmschleife	54	
5.4. Der logische Vergleichsbefehl CLC	60	
5.5. Der Sprungbefehl BC	64	
5.6. Pseudobefehle	70	
<b>6. Assemblerprotokoll und Dump</b>	<b>73</b>	
6.1. Aufgabenstellung	73	
6.1.1. Programmablaufplan »Vertreterprovision«	74	
6.1.2. Codierung	76	
6.2. Das Assemblerprotokoll	79	
6.3. Vergleich Objektprogramm – Primärprogramm	79	
6.4. Testlauf	82	
6.5. Erstellung eines Speicherabzugs, Dump	84	
<b>Übungen zu den Kapiteln 3 bis 6</b>	<b>86</b>	
<b>Ausblick</b>	<b>91</b>	
<b>7. Lösungen</b>	<b>93</b>	<b>A1</b>
<b>8. Anhang</b>	<b>107</b>	<b>A15</b>
8.1. Dv-Anlage	108	A16
8.2. Symbole für Programmablaufpläne	109	A17
8.3. Das Assemblerformular	110	A18
8.4. Die Assembleranweisung START	112	A20
8.5. Die Assembleranweisung END	112	A20
8.6. Die Assembleranweisung »Define Storage«, DS	113	A21
8.7. Die Assembleranweisung »Define Constant«, DC	114	A22
8.8. Der Befehl »Move Characters«, MVC	115	A23
8.9. Der Befehl »Compare Logical Characters«, CLC	116	A24
8.10. Der Befehl »Branch on Condition«, BC	117	A25
8.11. Pseudobefehle	118	A26
8.12. Der Makroaufruf »Lesen Satz«, GET	119	A27
8.13. Der Makroaufruf »Ausgeben Satz«, PUT	119	A27
8.14. Der Makroaufruf »Programmende«, EOJ	119	A27
8.15. EBCDI-Code-Tabelle	120	A28
8.16. Umwandlungstabelle Sedenzimal – Dezimal	122	A30
<b>Literaturverzeichnis</b>	<b>123</b>	
<b>Sachverzeichnis</b>	<b>124</b>	

## **Inhaltsübersicht Teil II**

1. Relative Adressierung von Assemblerprogrammen (USING, BALR)
2. Programmierung der Ein/Ausgabe mit den Makroaufrufen DTFSR, DTFEN, OPEN, CLOSE, GET, PUT, CNTRL
3. Funktionsweise und Erläuterung der wichtigsten logischen Befehle, Sprungbefehle, dezimalarithmetischen Befehle sowie der dazu erforderlichen Konstanten- und Speicherbereichsdefinitionen
4. Programmierung eines Lohnabrechnungsprogramms
5. Lösungen und Erläuterungen zu den zahlreichen Fragen, Aufgaben und Programmübungen
6. Anhang in Form einer knappen Zusammenfassung aller behandelten Befehle und Anweisungen, jeweils mit Beispielen

## **Inhaltsübersicht Teil III**

1. Dualarithmetik mit Registerbefehlen
2. Dualarithmetik mit RX-Befehlen, sowie die Programmierung einer Tabellenverarbeitung als Anwendung der Indexadressierung
3. Adressenrechnung
4. Druckaufbereitung mit dem EDIT-Befehl an einer Vielzahl von möglichen Aufgabenstellungen
5. Code-Umsetzung mit dem Translate-Befehl
6. Programmunterteilung und -Verknüpfung, Modultechnik und Segmentierung
7. Spezielle Übungen und Programme zum gesamten Stoffumfang aller drei Teile
8. Lösungen und Erläuterungen der Fragen, Aufgaben und Programme
9. Anhang mit Zusammenstellung aller behandelten Befehle und Anweisungen