Chapter 1: Assembler Language: Introduction

You will learn:

- Differences between machine and assembler instructions.
- Macro instructions.
- Assembly and link editing.
- Assembler processing.
- Macro and copy libraries.
- Control Sections.
- Assembler concepts - addressability, relocability, and registers.
- Coding conventions.
- Machine instruction formats - RR, RX, RS, SS, and SI.
- How to work with long control sections.

Chapter 2: Defining Data Constants and Symbols

You will learn:

- Data types.
- Defining constants.
- Truncation and padding.
- Alignment - constants and boundary.
- CNOP instruction.
- Defining literals.
- Defining storage: DS instruction.
- Defining symbols: EQU instruction.

Chapter 3: Assembler Listings

You will learn:

- External Symbol Dictionary.
- Source and object program.
- Relocation dictionary.
- Cross-Reference Table.
- How to use diagnostics and statistics.

Chapter 4: Fixed Point Instructions

You will learn:

- Load and Store instructions.
- Add instructions - A, AR, and AH.
- Subtract instructions - SR, S, and SH.
- Conditional processing.
- Codes used by instructions.
- Condition codes.
- Branching - BC and BCR instructions.
- BE instruction.
- B instruction.
- Extended mnemonic codes.
- LA: Load Address instruction.
- Branch On Count instruction - BCT and BCTR.
- MR: Multiply instruction.
- Shift instructions.
- Divide instructions.

Chapter 5: Logical Instructions

You will learn:

- MVC instruction - moving data.
- MVI instruction - moving one byte.
- CLC: Compare Logical Character instruction.
- CLI: Compare Logical Immediate instruction.

Chapter 6: File Handling

You will learn:

- The role and utilization of an access method.
- z/OS File and DCB: Data Control Block.
- OPEN, GET, PUT, and CLOSE macros.

Chapter 7: Packed Arithmetic

You will learn:

- Pack instruction.
- Symbols to facilitate coding.
- Conversion instruction.
- ZAP instruction.
- Decimal arithmetic instructions.
- CP instruction for comparing two packed decimal fields.
- Multiplying and dividing decimal instructions.
- How to use a pattern with special bytes.
- Message bytes.
- Simple selection.
- The Significance - Start Byte Hex 21.
- ED instruction.
- Using the Fill byte.
- Zero suppression - significance off and on.
- EDIT and MARK instructions.

Chapter 8: Calls and Linkage

You will learn:

- Subroutines.
- Save areas.
- Register usage.
- CALL macro.
- Parameter list.
- Passing a variable number of parameters.
- BAL: Branch and BALR: link instructions.
- External references and entry points.
- EXTRN statement.
- V-type constant.
- Entry point - called routine.
- Long control sections and multiple base registers.

Chapter 9: Table Handling

You will learn:

- RX instruction formats.
- Arrays.
- Tables.
- LOAD macro.
- DELETE macro.
- Initializing tables and arrays.

Chapter 10: VSAM Macros

You will learn:

- Accessing VSAM files using macros.
- Role of the ACB: Access Method Control Block.
- Using the OPEN, CLOSE, and TCLOSE macros.
- Using the MODCB and SHOWCB macros in conjunction with the ACB, EXLIST, and RPL macros.
- Using the GET, PUT, POINT, ENREQ, and ERASE macros.
- Return codes.
- ShowCB macro and feedback field.

Chapter 11: Assembler Dumps

You will learn:

- Program check, dump generation, and debugging.
- PSW and registers.
- Program relocation in the z/OS operating system.

Chapter 12: More Logical Instruction

You will learn:

- AND instructions.
- OR instructions.
- Non-Exclusive OR instructions.
- Setting bit switches.
- Exclusive OR instructions.
- TM: Test Under Mask instructions.
- Inserting and storing one byte.
- IC: Insert Character instruction.
- STC: Store Character instruction.
- EXECUTE instruction.
- BXH: Branch On Index High instruction.
- BXLE: Branch On Index Low or Equal instruction.
- TR: Translate and TRT instructions.
- CLM: Compare Logical Characters Under Mask instruction.
- ICM: Insert Character Under Mask instruction.
- STCM: Store Character Under Mask.

Chapter 13: 31-bit Addressing

You will learn:

- MVS and z/OS: system architecture.
- Addressing modes.
- Assembler programming with 31-bit mode.
- BSM - Branch and Set Mode.
- BASSM - Branch and Save and Set Mode.