

Enterprise PL/I Debugging and Maintenance

Course Summary

Description

The student will learn to approach debugging systematically, using the appropriate features of the PL/I language to facilitate debugging. In addition, he or she will learn to interpret the various forms of debugging output from PL/I, including CEEDUMPs.

Objectives

By the end of this course, students will be able to:

- Use the compile and run-time facilities of PL/I to determine the cause of a "program bug"
- Describe the outputs of the IBM Enterprise PL/I compiler, and use these outputs effectively in problem determination
- Understand the structure of the PL/I run time environment, and be able to find data elements in a PLIDUMP output listing
- Approach debugging in an orderly, efficient fashion
- Understand how PL/I calls, and passes parameters to subroutines
- Use the Binder map to determine in which CSECT an address is located.

Topics

- Introduction To PL/I Debugging
- Language Environment and PL/I
- How does PL/I store variables
- Where Does PL/I store variables
- The Enterprise PL/I Compiler
- Computer Exercise: PONION, continued
- Subroutines, Arguments, and Parameters
- Common PL/I programming errors

Audience

This course is designed for PL/I programmers who need to debug PL/I programs, understand the PL/I run-time environment, or tune PL/I programs for performance.

Prerequisite

The student should have experience coding PL/I programs, be comfortable with the text editor you use, and be able to code JCL to run simple batch jobs.

Duration

Two Days

Enterprise PL/I Debugging and Maintenance

Course Outline

- I. Introduction To PL/I Debugging**
 - A. Setting the Context
 - B. Overall Strategy
 - C. Computer Exercise: Course setup
- II. Language Environment and PL/I**
 - A. Debugging Resources
 1. Compiler Parameters
 2. Compiler Messages
 3. Execution-Time Parameters
 - B. Debugging Techniques
 1. PL/I Statements
 2. Conditions
 3. Computer Exercise: PONION
- III. How does PL/I store variables**
 - A. Arithmetic Variables
 - B. Bit and Character Variables
 - C. Pointer and Offset Variables
 - D. Area Variables
 - E. Arrays
 - F. Structures
 - G. Parameters
- IV. Where Does PL/I store variables**
 - A. Storage Classes
 - B. Static, Automatic, Controlled, and Based Storage
 - C. Dynamic Storage Areas (DSAs)
 - D. The User Heap
 - E. Computer Exercise: PONION, continued
- V. The Enterprise PL/I Compiler**
 - A. Inputs and Outputs
 - B. Compile Time Parameters
 - C. Anatomy of a PL/I Compiler Listing
 - D. Anatomy of a Binder Listing
 - E. Execution Time Parameters
- VI. Computer Exercise: PONION, continued**
 - A. Generating Dumps
 1. PLIDUMP
 2. CEE3DMP
 3. CEE3ABD
 - B. Working with Dumps
 1. Common Abends
 2. CEEDUMPs
- VII. Subroutines, Arguments, and Parameters**
 - A. Locators and Descriptors
 1. Dummy Arguments
 2. Dynamic Calls
 - B. Subroutines in Other Languages
 - C. PL/I Debugging Techniques
 1. ON ERROR
 2. PUT DATA, PUT LIST
 3. Pointer debugging
 4. Conditions
 5. Computer Exercise: Locators and Descriptors
- VIII. Common PL/I programming errors**
 - A. Good coding techniques
 - B. Poor coding techniques
 - C. Performance Considerations