



C Programming in the Guardian Environment

U4143S

HPE course number	U4143S
Course length	4 days
Delivery mode	ILT
View schedule, local pricing, and register	View now
View related courses	View now

Through a series of lectures and labs, you are taught how to use both the C compiler and the native mode C compiler used with HPE NonStop™ servers, how to take advantage of various extensions, and how to interface with Guardian system procedures. In learning to use those compilers, you will become familiar with various aspects of the unique C run-time environment such as memory model issues, diagnostics, special debugging issues, and the retrieval of specific run-time and environment information. Prior C knowledge is essential.

Why HPE Education Services?

- IDC MarketScape leader 4 years running for IT education and training*
- Recognized by IDC for leading with global coverage, unmatched technical expertise, and targeted education consulting services*
- Key partnerships with industry leaders OpenStack®, VMware®, Linux®, Microsoft®, ITIL, PMI, CSA, and (ISC)²
- Complete continuum of training delivery options—self-paced eLearning, custom education consulting, traditional classroom, video on-demand instruction, live virtual instructor-led with hands-on lab, dedicated onsite training
- Simplified purchase option with HPE Training Credits

Audience

- System analysts
- System programmers
- Programmers
- Programmer analysts

Prerequisites

- Concepts and Facilities for HPE NonStop Systems (U4147S)
- ANSI/ISO C Programming experience
- Recommended: Experience with the Inspect or Native Inspect symbolic debugger
- Recommended: TAL programming experience

Course objectives

- Perform C Programming within the Guardian system procedures

- Understand the unique aspects of the C run-time environment such as memory model issues, diagnostics, special debugging issues, and the retrieval of specific run-time and environment information

Benefits to you

- Building TNC C and native C programs
- Native mode considerations
- The C run-time environment
- Compiler operation and diagnostics
- Debugging C programs with Inspect and Native Inspect
- Interfacing to Guardian system procedures
- Mixed language programming

*Realize Technology Value with Training, IDC Infographic 2037, Sponsored by HPE, January 2016

Detailed course outline

Module 1: Compiling and Linking TNS C programs

- Basic operation of the TNS C compiler
- Compiler syntax and the analysis of a compiler listing
- Binder commands needed to combine multiple object files to build a single executable program
- Lab Exercise: Compile a complete C program
- Lab Exercise: Separate compilation and binding

Module 2: Native Mode Considerations

- Native mode
- Benefits and constraints of native mode and the native mode development tools
- Process attributes and organization

Module 3: Native Mode Compilation

- Various native mode migration issues particular to C programming
- Using the TNS/E native compiler (CCOMP)
- Usage of the eld utility for linking separately compiled modules
- Lab Exercise: Use the CCOMP compiler to build runnable programs from separately compiled units

Module 4: Run-time Environment

- NonStop server run-time environment for C programs
- Obtaining and processing environment and startup protocol messages
- Essentials of memory models available for NonStop C and show the four basic areas of heap, RTL, global and stack
- 32 bit pointers and access to system procedures that require 16 bits
- Large and wide memory models
- Lab Exercise: Gain hands-on experience in fetching process startup information

Module 5: Tandem C Compiler Operation and Diagnostics

- Subset of the available compiler pragmas for Tandem C
- Compiler pragmas for the native C compiler
- Compile-time and run-time diagnostic facilities
- Demonstrating, with examples, various aspects of compiler operations

Module 6: Debugging C Programs

- Basic framework of Inspect and Native Inspect commands which can be used for debugging NonStop server C programs
- Information regarding Inspect and Native Inspect commands specifically helpful for the NonStop server C environment
- Demonstrating a debugging session
- Lab Exercise: Use Inspect and Native Inspect on a C program

Module 7: Interfacing to Guardian Procedures

- Specifics of interfacing to Guardian procedures from a C program
- Interface declarations for non-C functions
- Usage of the Guardian procedure calls reference manual and special/problem Guardian procedures
- Lab Exercise: Make calls to Guardian procedures needed for various terminal display options

Module 8: Mixed-Language Programming

- Mixed language programming as it relates to NonStop C
- Interface declarations, common run-time environment (CRE), specific alignment rules, data storage requirements, and global data sharing
- Lab Exercise: Gain hands-on experience with interface between NonStop C and other languages such as TAL

Onsite-delivery equipment requirements:

- One NonStop server with NonStop operating system, version H06 or later
 - 6530 terminals or equivalent PC/workstation for instructor demonstrations
 - 6530 terminal or equivalent PC/workstation per student
 - C, NMC, nld, noft, TAL and pTAL software
 - Printer for compilation listings
-

Next steps

- Guardian API Programming (U4152S)

Learn more at
hpe.com/ww/learnnonstop

Follow us:



© Copyright 2015–2016 Hewlett Packard Enterprise Development LP. The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

Microsoft is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries. The OpenStack Word Mark is either a registered trademark/service mark or trademark/service mark of the OpenStack Foundation, in the United States and other countries and is used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation or the OpenStack community. Pivotal and Cloud Foundry are trademarks and/or registered trademarks of Pivotal Software, Inc. in the United States and/or other countries. Linux is the registered trademark of Linus Torvalds in the U.S. and other countries. VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other third-party trademark(s) is/are property of their respective owner(s).