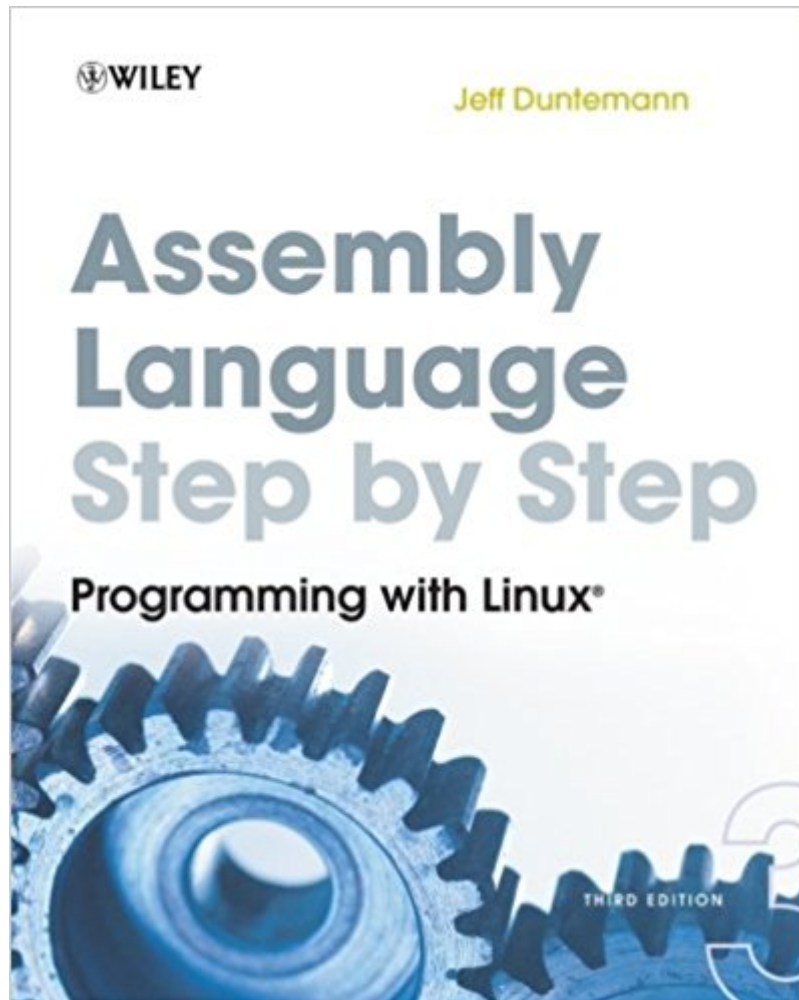




Ebook Directory
the best source of ebook

The book was found

Assembly Language Step-by-Step: Programming With Linux



Synopsis

The eagerly anticipated new edition of the bestselling introduction to x86 assembly language. The long-awaited third edition of this bestselling introduction to assembly language has been completely rewritten to focus on 32-bit protected-mode Linux and the free NASM assembler. Assembly is the fundamental language bridging human ideas and the pure silicon hearts of computers, and popular author Jeff Dunteman retains his distinctive lighthearted style as he presents a step-by-step approach to this difficult technical discipline. He starts at the very beginning, explaining the basic ideas of programmable computing, the binary and hexadecimal number systems, the Intel x86 computer architecture, and the process of software development under Linux. From that foundation he systematically treats the x86 instruction set, memory addressing, procedures, macros, and interface to the C-language code libraries upon which Linux itself is built. Serves as an ideal introduction to x86 computing concepts, as demonstrated by the only language directly understood by the CPU itself. Uses an approachable, conversational style that assumes no prior experience in programming of any kind. Presents x86 architecture and assembly concepts through a cumulative tutorial approach that is ideal for self-paced instruction. Focuses entirely on free, open-source software, including Ubuntu Linux, the NASM assembler, the Kate editor, and the Gdb/Insight debugger. Includes an x86 instruction set reference for the most common machine instructions, specifically tailored for use by programming beginners. Woven into the presentation are plenty of assembly code examples, plus practical tips on software design, coding, testing, and debugging, all using free, open-source software that may be downloaded without charge from the Internet.

Book Information

Paperback: 648 pages

Publisher: Wiley; 3 edition (October 5, 2009)

Language: English

ISBN-10: 0470497025

ISBN-13: 978-0470497029

Product Dimensions: 7.4 x 1.5 x 9.1 inches

Shipping Weight: 2 pounds (View shipping rates and policies)

Average Customer Review: 4.1 out of 5 stars 97 customer reviews

Best Sellers Rank: #84,945 in Books (See Top 100 in Books) #8 in [Books > Computers & Technology > Programming > Languages & Tools > Assembly Language Programming](#) #24 in [Books > Computers & Technology > Operating Systems > Linux > Programming](#) #97

Customer Reviews

Learn assembly language, and you learn the machine In this third edition of his bestselling guide to Intel x86 assembly language under Linux, Jeff Duntemann positions assembly not as unapproachable geek arcana but as a first programming language, suitable for readers who have no previous programming experience. As the fundamental language of the CPU, assembly lays the groundwork for all other programming languages, especially native-code C, C++, and Pascal. By mastering assembly, programmers will learn how x86 computers operate all the way down to "the bare silicon," at a level of detail that no other approach can equal. Assembly Language Step by Step, Third Edition, helps you:

- Review the fundamental concepts behind computing and programming, including the hexadecimal and binary number bases
- Understand the evolution of the Intel CPUs and how modern x86 processors operate
- Grasp the process of programming itself, from editing source code through assembly, linking, and debugging
- Comprehend x86 32-bit protected-mode memory addressing
- Learn the x86 instruction set by dissecting numerous complete example programs
- Work with the wealth of free programming utilities under Ubuntu Linux, including the Kate editor, the NASM assembler, and the GNU toolset
- Master practical details of Linux programming, including procedures, macros, the INT 80h call gate, and calls to the standard C libraries

Jeff Duntemann has been writing about computing for over thirty years, and is the author of numerous books on programming, wireless networking, and system administration. He has been a columnist in Dr. Dobb's Journal, and has edited well-known programming publications like PC Techniques and Visual Developer. After hours, he enjoys blogging, astronomy, amateur radio, and writing science fiction.

Still relevant after all this time and pertinent for anyone learning Intel 32-bit architecture specific assembler. Code examples (primarily the labs) can appear campy at times, but deliver appropriately being concerned with progressive, modular design philosophies. Additionally, Duntemann sticks to the assembler ideologies of module development rather than using abstraction/poly-morphism (with proprietary libraries) allowing the reader an opportunity to learn rather than decipher subject matter. Globally, the content is on point and covers much ground addressing issues that every assembler developer will immediately face whether developing low-level or optimizing high-level

libraries. Based on numerous factors and an established curve, regarding the sparse nature of content on the subject, feel the five star rating deserved.

I ordered the latest edition even though I don't particularly need to learn at the level Duntemann is teaching. I enjoy his writing and wanted to see how he updated the book to deal with LINUX and modern x86 assembly language. As per usual it's well written. If I ever write a tutorial book I will reread this to remind myself how it should be done. The only (minor) negative observations I have are: 1) The book only addresses 32 bit programming in detail, but 32 bit programs will run in 64 bit without change. 2) Duntemann dives into open source software and teaches/recommends a particular debugger that is no longer being developed. Most of the commands will map to other debuggers (it's all GDB underneath) but this weakens an otherwise excellent book. Highly recommended if you want to learn assembly language on LINUX.

I don't even know how to start... I think all of us had that feeling when you read a good book or watch a good movie and you never want that good thing to end. I feel sad because I finished the book, it's like a good thing from my life gone now. The best book I ever read. I love the style of writing. The author is starting slowly from the very beginning explaining how you can put your daily actions on a paper and afterwards in a program with very nice examples. As the title suggests you will learn how to develop assembly language programs for Intel architecture x86 using NASM assembler and Insight debugging tool on Linux (Ubuntu). If you use a newer version of Ubuntu or other linux OS, you can easily find on the internet how to install Insight. For me the best chapter in the book was Memory Addressing which I read many times just to be sure I will master the memory addressing. The way he explains the things I never met, not even in college, easy to understand with diagrams and good examples all the time. Another thing which I love is that always when I was reading something and not understood entirely, next page he was explaining exactly that thing! I could fill pages writing good things about this book. At the end of the book you will have a final test and like Mr. Duntemann is saying, "Pull that off, and you can take a bow: You will be an assembly language programmer!", I would like to take a bow in front of you Mr. Duntemann, you are awesome! I made a very small program which will review Mr. Duntemann's book and I was afraid to post it because I don't know how the text will be formatted so I took a screenshot. If you know assembly you will know the result, if not, you will have to read the book and come back one day :)

Awesome book. Easy read. I learned assembly with ease.

Good entry level book on Assembly Language.

This is a great book to begin your computer programming career. Before you go off and hack around in java, php, ruby etc. read this so you understand how computers work, what's going on in Linux, how stack frames work, how arguments are passed are in your programs, etc

Awesome book that starts the reader at the beginning of how CPU's and memory works. I had trouble visualizing memory addressing, registers, and where everything is until I bought this book. This book does a great job of breaking the CPU architecture and registers down in an understandable format.

I read this book 2 years ago to compare it with Randall Hyde's Art of Assembly Language which I had read 7 years earlier. Both books have strengths. Art of Assembly Language is stronger in terms of structure and description of concepts. It is more high level and provides the reader with a clearer general outline of assembly language and computer program design related to CPU's. The benefit of the Art of Assembly language is that you may gain insights into how to optimize a computer program regardless of programming language. Assembly Language Step-by-Step is much, much more detailed and is designed primarily for someone who actually intends to use assembly language. It is more of a hands-on, real-world book than the Art of Assembly language. Based on the amount of detail, you may not come away with a clear overview, but you will have knowledge about using assembly language in a very direct and concrete way. The details that are provided can give you tremendous understanding about the different ways software code can be set up to do things. It is a book that gives credibility to the old saying that there are thousands of ways to write the same code. I used managed programming languages for 11 years straight. In the early years, I have seen these managed languages run slower in some areas compared to similar solutions in other languages not based on a managed paradigm. A knowledge of system characteristics and how managed languages transform into system representations can definitely help in producing more efficient solutions. Books like this one can be a benefit to anyone regardless of chosen language or to better inform their understanding of computers in conversation or use.

[Download to continue reading...](#)

Assembly Language Step-by-Step: Programming with Linux The Linux Programming Interface: A Linux and UNIX System Programming Handbook CompTIA Linux+ Powered by Linux Professional

Institute Study Guide: Exam LX0-103 and Exam LX0-104 (Comptia Linux + Study Guide) Python Programming: Python Programming for Beginners, Python Programming for Intermediates, Python Programming for Advanced C++: The Ultimate Crash Course to Learning the Basics of C++ (C programming, C++ in easy steps, C++ programming, Start coding today) (CSS,C Programming, ... Programming,PHP, Coding, Java Book 1) Python Programming: The Complete Step By Step Guide to Master Python Programming and Start Coding Today! (Computer Programming Book 4) A Communication from Sir Charles Brisbane, K.C.B. Governor of Saint Vincent: To the House of Assembly of That Colony, Enclosing Lord Bathurst's ... Assembly; and a Letter Depicting the Alarm Gun Digest Book of Automatic Pistols Assembly/Disassembly (Gun Digest Book of Firearms Assembly/Disassembly) The Gun Digest Book of Firearms Assembly/Disassembly Part V - Shotguns: Shotguns Pt.5 (Gun Digest Book of Firearms Assembly/Disassembly: Part 5 Shotguns) The Gun Digest Book of Tactical Weapons Assembly/Disassembly (Gun Digest Book of Firearms Assembly/Disassembly) Easy Linux For Beginners: A Complete Introduction To Linux Operating System & Command Line Fast! CompTIA Linux+ Guide to Linux Certification Python: Programming: Your Step By Step Guide To Easily Learn Python in 7 Days (Python for Beginners, Python Programming for Beginners, Learn Python, Python Language) C++ and Python Programming: 2 Manuscript Bundle: Introductory Beginners Guide to Learn C++ Programming and Python Programming C++ and Python Programming 2 Bundle Manuscript. Introductory Beginners Guide to Learn C++ Programming and Python Programming JAVASCRIPT: Easy JavaScript Programming For Beginners. Your Step-By-Step Guide to Learning JavaScript Programming (JavaScript Series) A Practical Guide to Linux Commands, Editors, and Shell Programming (3rd Edition) Linux System Programming: Talking Directly to the Kernel and C Library Beginning Linux Programming Hacking University: Learn Python Computer Programming from Scratch & Precisely Learn How the Linux Operating Command Line Works: 2 Manuscript Bundle

[Contact Us](#)

[DMCA](#)

[Privacy](#)

[FAQ & Help](#)