Concurrent in Ada
Alan Burns and Andy Wellings
Second edition
A major feature of the Ada programming language is the facilities it provides for concurrent
programming. In this book, Alan Burns and Andy Wellings provide a thorough and self-contained
account of concurrent programming in Ada, and show users how to harness the full power of the
language. Starting with an overview of the nonconcurrent features of Ada, the authors examine in
detail the uses of concurrent programming and the inherent difficulties in providing interprocess
communication. They introduce the Ada tasking model, and explain system programming, real-time
issues, distribution, object-oriented programming, and reuse. This is the first book to deal with
concurrent features in the new Ada standard, and it offers practical advice to both programmers
working with embedded systems and those interested more broadly in the development of
programming languages. Many otherwise inaccessible issues are probed in depth, making this book
invaluable to professional software engineers and advanced students of programming alike. Every
Ada programmer will find it essential reading and a primary reference work.

Book Information

Paperback: 442 pages
Publisher: Cambridge University Press; 2 edition (March 13, 1998)
Language: English
ISBN-10: 052162911X
Product Dimensions: 6 x 1.1 x 9 inches
Shipping Weight: 1.4 pounds
Average Customer Review: 4.8 out of 5 stars See all reviews (6 customer reviews)
Best Sellers Rank: #2,926,483 in Books (See Top 100 in Books) #38 in Books > Computers &
Technology > Programming > Languages & Tools > Ada #3033 in Books > Textbooks >
Computer Science > Networking #3406 in Books > Textbooks > Computer Science > Software
Design & Engineering

Customer Reviews

Fully harnessing multi-core processors requires concurrent programming, yet most mainstream
languages have little support or do so with poor portability. Well-written concurrent Ada programs
can be extremely portable and support multi-core and multiple processors automatically. This book
explores, in-depth, the concurrent programming ("tasking") part of the Ada language. The authors
then show how the concurrent programming facilities of Ada are augmented by the language to
support real-time programming. These standard real-time facilities are state-of-the-art, extending, for example, beyond the POSIX real-time facilities in their expressive power. The book first establishes the necessary foundation for understanding concurrent programming by explaining the possible problems (e.g., deadlocks and race conditions) and highlighting some of the mechanisms traditionally used in concurrent programming, such as semaphores and message-passing. The tasking part of Ada is then covered in full detail, necessarily including interactions with some other parts of the language, such as exceptions, but also including interactions with the Ada object-oriented language facilities. The implementations of a number of reusable concurrency abstractions serve to illustrate the use of the tasking features. The authors next lay another foundation, this time for scheduling in real-time systems, and show how Ada directly supports the common approaches. Both fixed-priority and dynamic-priority dispatching are supported by Ada and these are covered in detail. Ada 2005 also defines a number of time-related abstractions, such as a monotonic clock and timing events, and these are also covered completely.

Download to continue reading...

Object-Oriented Programming Ada 95: The Lovelace Tutorial Ada 2012 Rationale: The Language --
The Standard Libraries (Lecture Notes in Computer Science)