C Concurrency In Action

Human lives are getting increasingly entangled with technology, especially comp- ing and electronics. At each step we take, especially in a developing world, we are dependent on various gadgets such as cell phones, handheld PDAs, netbooks, me- cal prosthetic devices, and medical measurement devices (e.g., blood pressure m- itors, glucometers). Two important design constraints for such consumer electronics are their form factor and battery life. This translates to the requirements of reduction in the die area and reduced power consumption for the semiconductor chips that go inside these gadgets. Performance is also important, as increasingly sophisticated applications run on these devices, and many of them require fast response time. The form factor of such electronics goods depends not only on the overall area of the chips inside them but also on the packaging, which depends on thermal ch- acteristics. Thermal characteristics in turn depend on peak power signature of the chips. As a result, while the overall energy usage reduction increases battery life, peak power reduction in?uences the form factor. One more important aspect of these electronic equipments is that every 6 months or so, a newer feature needs to be added to keep ahead of the market competition, and hence new designs have to be completed with these new features, better form factor, battery life, and performance every few months. This extreme pressure on the time to market is another force that drives the innovations in design automation of semiconductor chips.

???????IEEE?????????—POSIX??(????Pthreads??),????????????????????????,????????????????????????????????????:????

????????????,????????????C++???????,????????????,????????????????????????,??????????????,????????????

Software Engineer's Reference Book provides the fundamental principles and general approaches, contemporary information, and applications for developing the software of computer systems. The book is comprised of three main parts, an epilogue, and a comprehensive index. The first part covers the theory of computer science and relevant mathematics. Topics under this section include logic, set theory, Turing machines, theory of computation, and computational complexity. Part II is a discussion of software development methods, techniques and technology primarily based around a conventional view of the software life cycle. Topics discussed include methods such as CORE, SSADM, and SREM, and formal methods including VDM and Z. Attention is also given to other technical activities in the life cycle including testing and prototyping. The final part describes the techniques and standards which are relevant in producing particular classes of application. The text will be of great use to software engineers, software project managers, and students of computer science.

C?C++????
??????????????? ??????C++11?? ?????C++11??????????????????C++??
?? ?C++ Primer, 5th Edition ????????????C++????????????????????????
??? ????&?????
????????C++11????????????????????????????????? ?????????????????????????????????
???????????????C++11???? ?????????????????????????? ??
??????????C++?????????C++?????????????????????????????????????C++?????????? #???? GOTOP .
???C++11??Bjarne
Stroustrup?C++????????????????????????????C++???????????????????????? ?The C++ Programming Language,
Fourth Edition??C++??(????????????????????????)?????????????????????Stroustrup?????????C++11????????????????
????????????C++???????????ISO???????????????

??C++11???????

??????? ?????????????????????????? ??????????????for?????move???Unicode??
?Lambda?????????????????????????????(variadic template)?????(template alias)??????????? ??????

????????????

?????????????????(scope)?????(storage)?????? ????(modularity)????????(namespace)?????????(exception handling)
?C++????(???class?class?????template)?????????????????????????????(generic programming)
?????????(container)????????(iterator)?????(utility)???????I/O?locale???(numerics)? ?C++?????????????????
???????????(????????C++98?????)????????C++11?????????????????????C++11???????????? #???? GOTOP
Information Inc.
Many of the normal concerns faced by application developers are amplified by the challenges of web-scale concurrency, real-time performance expectations, multi-core support, and efficiently consuming services without constantly managing I/O blocks. Although it's possible to solve most of these issues with existing languages and frameworks, Go is designed to handle them right out of the box, making for a more natural and productive coding experience. Developed at Google

for its own internal use, Go now powers dozens of nimble startups, along with name brands like Canonical, Heroku, SoundCloud, and Mozilla, who rely on highly performant services for their infrastructure. Go in Action introduces the unique features and concepts of the Go language, guiding readers from inquisitive developers to Go gurus. It provides hands-on experience with writing real-world applications including web sites and network servers, as well as techniques to manipulate and convert data at incredibly high speeds. It also goes in-depth with the language and explains the tricks and secrets that the Go masters are using to make their applications perform. For example, it looks at Go's powerful reflection libraries and uses real-world examples of integration with C code. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.

??????Java???57????????????,????10?,??????:??????????????????????????C??????????????????

Collects in four chapters single monographs related to the fundamental advances in parallel computer systems and their developments from different points of view (from computer scientists, computer manufacturers, end users) and related to the establishment and evolution of grids fundamentals, implementation and deployment.

?????????,???????C?????,???????C?????????????.?????????,??????????????????????????,????????????????

API.

"This book should be on every C++ programmer's desk. It's clear, concise, and valuable." Rob Green, Bowling Green State University This bestseller has been updated and revised to cover all the latest changes to C++ 14 and 17! C++ Concurrency in Action, Second Edition teaches you everything you need to write robust and elegant multithreaded applications in C++17. You choose C++ when your applications need to run fast. Well-designed concurrency makes them go even faster. C++ 17 delivers strong support for the multithreaded, multiprocessor programming required for fast graphic processing, machine learning, and other performance-sensitive tasks. This exceptional book/course unpacks the features, patterns, and best practices of production-grade C++ concurrency. C++ Concurrency in Action, Second Edition is the definitive guide to writing elegant multithreaded applications in C++. Updated for C++ 17, it carefully addresses every aspect of concurrent development, from starting new threads to designing fully functional multithreaded algorithms and data structures. Concurrency master Anthony Williams presents examples and practical tasks in every chapter, including insights that will delight even the most experienced developer. Inside: Full coverage of new C++ 17 features Starting and managing threads Synchronizing concurrent operations Designing concurrent code Debugging multithreaded applications This book/course is written for intermediate C and C++ developers. No prior experience with concurrency required. Anthony Williams has been an active member of the BSI C++ Panel since 2001 and is the developer of the just::thread Pro extensions to the C++ 11 thread library. A thorough presentation of C++ concurrency

capabilities. Maurizio Tomasi, University of Milan Highly recommended for programmers who want to further their knowledge of the latest C++ standard. Frédéric Flayol, 4Pro Web C++ The guide contains snippets for everyday use in your own projects and to help take your concurrency C++ skills from the Padawan to the Jedi level. Jura Shikin, IVI Technologies NARRATED BY LISA FARINA AND DIANA GARDINER.

Specifying and implementing dynamical systems with the situation calculus. Modeling and implementing dynamical systems is a central problem in artificial intelligence, robotics, software agents, simulation, decision and control theory, and many other disciplines. In recent years, a new approach to representing such systems, grounded in mathematical logic, has been developed within the AI knowledge-representation community. This book presents a comprehensive treatment of these ideas, basing its theoretical and implementation foundations on the situation calculus, a dialect of first-order logic. Within this framework, it develops many features of dynamical systems modeling, including time, processes, concurrency, exogenous events, reactivity, sensing and knowledge, probabilistic uncertainty, and decision theory. It also describes and implements a new family of high-level programming languages suitable for writing control programs for dynamical systems. Finally, it includes situation calculus specifications for a wide range of examples drawn from cognitive robotics, planning, simulation, databases, and decision theory, together with all the implementation code for these examples. This code is available on the book's Web site.

?????Go?????????????????????Go?????????????????Go???????????JavaScript?Ruby?Python?Java?C++?????????????
??????Go????????????????????????????????????
?????????Go???Go?????????????
????????????Go??
?????????????????????????goroutine?channel??????????Go??
???????Go???????????reflection???????????????unsafe?????????????????cgo????Go?C??????
???????????Go??http://gopl.io/??????go get???????????? #???? GOTOP Information Inc.

This text takes complicated and almost unapproachable parallel programming techniques and presents them in a simple, understandable manner. It covers the fundamentals of programming for distributed environments like Internets and Intranets as well as the topic of Web Based Agents.

????15???????????????GDI+????????????;?????????????????????????????????Web???,??12?13??????ADO.NET?WinForms??????
????

Provides information on building concurrent applications using Java.

?????:????
????:????????

Concurrent programming has become a required discipline for all programmers. Multi-core processors and the increasing demand for maximum performance and scalability in mission-critical applications have renewed interest in functional languages like Erlang that are designed to handle concurrent programming. Erlang, and the OTP platform, make it possible to deliver more robust applications that satisfy rigorous uptime and performance requirements. Erlang and OTP in Action teaches you to apply Erlang's message passing model for concurrent programming--a completely different way of tackling the problem of parallel programming from the more common multi-threaded approach. This book walks you through the practical considerations and steps of building systems in Erlang and integrating them with real-world C/C++, Java, and .NET applications. Unlike other books on the market, Erlang and OTP in Action offers a comprehensive view of how concurrency relates to SOA and web technologies. This hands-on guide is perfect for readers just learning Erlang or for those who want to apply their theoretical knowledge of this powerful language. You'll delve into the Erlang language and OTP runtime by building several progressively more interesting real-world distributed applications. Once you are competent in the fundamentals of Erlang, the book takes you on a deep dive into the process of designing complex software systems in Erlang. Purchase of the print book comes with an offer of a free PDF, ePub, and Kindle eBook from Manning. Also available is all code from the book.

C++ Concurrency in ActionManning Publications

With the new C++ Standard and Technical Report 2 (TR2), multi-threading is coming to C++ in a big way. TR2 will provide higher-level synchronization facilities that allow for a much greater level of abstraction, and make programming multi-threaded applications simpler and safer. Concurrent programming is required if programmers are to take advantage of the multi-core microprocessors increasingly available from Intel and others. The new standard for C++ has extensions to the language that make concurrent programming more accessible to regular developers. As a guide and reference to the new concurrency features in the upcoming C++ Standard and TR2, this book is invaluable for existing programmers familiar with writing multi-threaded code in C++ using platform-specific APIs, or in other languages, as well as C++ programmers who have never written multithreaded code before.

????16?,??:“?????”?“?????”?“??????”?“???????”?“????????????”??

????Tim Peierls?Joshua Bloch?Joseph Bowbeer?David Holmes?Doug Lea

C++ Concurrency in Action, Second Edition is the definitive guide to writing elegant multithreaded applications in C++. Updated for C++ 17, it carefully addresses every aspect of concurrent development, from starting new threads to designing fully functional multithreaded algorithms and data structures. Concurrency master Anthony Williams presents examples and practical tasks in every chapter, including insights that will delight even the most experienced developer. --

Provided by publisher.

?????C++??????????????STL??????????????????????????????

Concurrent C is a superset of C that provides parallel programming facilities such as those for the declaring and creating processes, for process synchronization and interaction, and for process termination and abortion. Concurrent C was designed for the effective utilization of multiprocessors and multicomputers. Concurrent C, as a compile-time option, also works with C++, an object-oriented superset of C.

????#?C?????

?????C++??????,C++??????????????????,??????????????????,???C++??????????????????

????:Richard Helm,Ralph Johnson,John Vlissides ????:???,??,???

Intelligent agents are one of the most important developments in computer science of the past decade. Agents are of interest in many important application areas, ranging from human-computer interaction to industrial process control. The ATAL workshop series aims to bring together researchers interested in the core/micro aspects of agent technology. Speci?cally, ATAL addresses issues such as theories of agency, software architectures for intelligent agents, methodologies and programming languages for r-lizing agents, and software tools for applying and evaluating agent systems. One of the strengthsoftheATALworkshopseriesisitsemphasisonthesynergiesbetweentheories, languages, architectures, infrastructures, methodologies, and formal methods. This year s workshop continued the ATAL trend of attracting a large number of high quality submissions. In more detail, 71 papers were submitted to the ATAL 2000 workshop, from 21 countries. After stringent reviewing, 22 papers were accepted for publication and appear in these proceedings. As with previous workshops in the series, we chose to emphasize what we perceive asimportantnewthemesinagentresearch. Thisyear sthemeswerebothassociatedwith the fact that the technology of intelligent agents and multi-agent systems is beginning to migrate from research labs to software engineering centers. As agents are deployed in applications such as electronic commerce, and start to take over responsibilities for their human users, techniques for controlling their autonomy become crucial. As well, the availability of tools that facilitate the design and implementation of agent systems becomes an important factor in how rapidly the technology will achieve widespread use.