1. Record Nr. UNINA9911006790603321

Autore	Hillar Gaston C
Titolo	C# 2008 and 2005 threaded programming : beginner's guide : exploit the power of multiple processors for faster, more responsive software / / Gaston C. Hillar
Pubbl/distr/stampa	Birmingham, U.K., : Packt Pub., 2009
ISBN	1-62198-890-2 1-282-00879-X 9786612008795 1-84719-711-6
Edizione	[1st ed.]
Descrizione fisica	1 online resource (416 p.)
Disciplina	005.133
Soggetti	C# (Computer program language) Threads (Computer programs)
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	Cover; Table of Content; Preface; Chatper 1: Taking Advantage of Multiprocessing and Multiple Cores; Mono-processor systems: The old gladiators; Single core: Only one warrior to fight against everybody; Doing a tiny bit of each task; The performance waterfall; Multi-processor systems: Many warriors to win a battle; Estimating performance improvements; Avoiding bottlenecks; Taking advantage of multiple execution cores; Scalability; Load balancing: Keeping everybody happy; Operating systems and virtual machines; Parallelism is here to stay; Summary; Chatper 2: Processes and Threads Processes-any running programTime for action - Coding a simple CPU-intensive loop; Time for action - Changing the cores available for a process; Relating processes to cores; Time for action - Changing a process priority; Linear code problems in multiprocessing systems; Time for action - Running many processes in parallel; Time for action - Testing parallelism capabilities with processes; Time for action - Using the Process Explorer; Threads-Independent parts of a process; Time for action - Listing threads with Process Explorer Time for action - Analyzing context switches with Process

ExplorerMultiple threads in servers; Multiple threads in clients; Summary; Chatper 3: BackgroundWorker-Putting Threads to Work; RTC: Rapid thread creation; Time for action - Breaking a code in a single thread; Time for action - Defining the work to be done in a new thread; Asynchronous execution; Time for action - Understanding asynchronous execution step-by-step; Synchronous execution; Showing the progress; Time for action - Using a BackgroundWorker to report progress in the UI; Cancelling the job
Time for action - Using a BackgroundWorker to cancel the jobTime for action - Using a BackgroundWorker to detect a job completed; Time for action - Working with parameters and results; Working with multiple BackgroundWorker components; Time for action - Using many BackgroundWorker components to break; the code faster; BackgroundWorker and Timer; BackgroundWorker creation on the fly; Time for action - Creating BackgroundWorker components in run-time; Summary; Chatper 4: Thread Class-Practical Multithreading in Applications; Creating threads with the Thread class
Time for action - Defining methods for encryption and decryptionTime for action - Running the encryption in a new thread using the; Thread class; Decoupling the UI; Creating a new thread; Retrieving data from threads; Sharing data between threads; Time for action - Updating the UI while running threads; Sharing some specific data between threads; A BackgroundWorker helping a Thread class; Time for action - Executing the thread synchronously; Main and secondary threads; Passing parameters to threads; Time for action - Using lists for thread creation on the fly I
Time for action - Using lists for thread creation on the fly II

Sommario/riassunto

Exploit the power of multiple processors for faster, more responsive software.