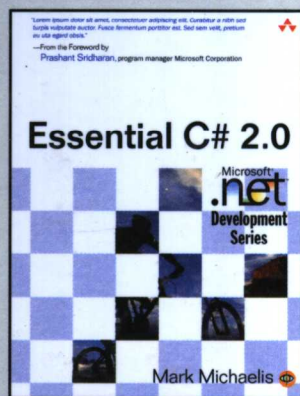


[美] Mark Michaelis 著

Essential C# 2.0

(英文版)



**这本参考指南在工作时应该放在伸手可及的地方，在外
出时应该带在背包中。现在就开始阅读吧！**

——Prashant Sridharan 微软公司项目经理

TP312/Y286

2007.



典藏原版书苑

Essential C# 2.0

(英文版)

[美] Mark Michaelis 著

江苏工业学院图书馆
藏书章

人民邮电出版社

北京

图书在版编目 (CIP) 数据

Essential C# 2.0: 英文 / (美) 米凯利斯 (Michaelis, M.)
著. —北京: 人民邮电出版社, 2007.12
(典藏原版书苑)
ISBN 978-7-115-16687-6

I. E… II. 米… III. C 语言—程序设计—英文 IV. TP312

中国版本图书馆 CIP 数据核字 (2007) 第 127248 号

版 权 声 明

Original edition, entitled Essential C# 2.0, 1st Edition, 0321150775 by Mark Michaelis, published by Pearson Education, Inc, publishing as Addison Wesley Professional, Copyright © 2007 by Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

China edition published by PEARSON EDUCATION ASIA LTD., and POSTS & TELECOMMUNICATIONS PRESS Copyright © 2007.

This edition is manufactured in the People's Republic of China, and is authorized for sale only in People's Republic of China excluding Hong Kong, Macau and Taiwan.

仅限于中华人民共和国境内 (不包括中国香港、澳门特别行政区和中国台湾地区) 销售。本书封面贴有 Pearson Education (培生教育出版集团) 激光防伪标签。无标签者不得销售。

典藏原版书苑

Essential C# 2.0 (英文版)

-
- ◆ 著 [美] Mark Michaelis
责任编辑 付 飞
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销
 - ◆ 开本: 800×1000 1/16
印张: 47
字数: 1 056 千字 2007 年 12 月第 1 版
印数: 1—3 000 册 2007 年 12 月北京第 1 次印刷

著作权合同登记号 图字: 01-2007-2134 号

ISBN 978-7-115-16687-6/TP

定价: 95.00 元

读者服务热线: (010)67132705 印装质量热线: (010)67129223

内容提要

本书以大量实例，详尽地阐述了 C# 2.0 的核心特性，指导方针和最佳实践。书中代码采用了专门的格式，并采用“知识地图”的形式概括出每一章的主要内容。

本书全面讲解了 C# 语言，读者从中能够学到：C# 基本数据类型、值类型、引用类型、类型转换和数组；运算符、控制流、循环、条件逻辑和顺序编程；方法、参数、异常处理和结构化编程；类、遗传、结构接口和面向对象编程；结构良好的类型、运算符重载、命名空间和垃圾回收；泛型、集合和迭代子；映射、属性和声明性编程；线程、同步和多线程模式；互操作性和不安全代码；C# 通用语言基础结构（CLI）。

本书内容全面系统，并经过精心编排，相信无论是初学编程的新手、有其他语言编程经验的开发者或是专业 C# 程序员，阅读本书都将受益匪浅。

Praise for *Essential C# 2.0*

"Essential C# 2.0 pulls off a very difficult task. The early chapters are comprehensible by beginning developers, while the later chapters pull no punches and provide the experienced developer with the detailed information they need to make the most of C# 2.0. Starting with the first chapter, Mark has successfully interwoven tidbits of information useful to even the most advanced developer while keeping the book approachable."

—Chris Kinsman, chief architect, Vertafore, Microsoft Regional Director

"How refreshing! This book deals with C# thoroughly, rather than skimming over the whole .NET framework. It is valuable to newcomers and professionals alike."

—Jon Skeet, C# MVP

"Essential C# 2.0 is a one-stop shop for an experienced programmer looking to ramp up on one of the hottest languages around today. Mark delivers an intelligent and detailed tour of C#, providing newcomers to the language with a solid foundation of skill on which to build their next generation of applications."

—Stephen Toub, technical editor, MSDN Magazine

"This book provides complete, up-to-date coverage of all the programming constructs in C#. Masterfully organized, it allows beginning programmers to get on board and leads more experienced programmers into the world of structured programming. Because of its unwavering focus on the essential programming constructs of C#—such as generics, delegates, and much more—this book is indispensable. For programmers who want to solve their day-to-day programming issues using the latest features this modern programming language has to offer, this book is indispensable."

—Narendra Poflee, IT integration specialist, Itron Inc.

"Essential C# 2.0 is an ideal book for all programmers interested in C#. If you are a beginner, you will quickly learn the basics of C# programming and become familiar with the concepts. The flow of the text is easy to follow and does a great job of not repeating concepts that have already been covered. For the experienced programmer, this book has priceless nuggets embedded within its pages, making it a great read for programmers who are already familiar with C#. This will be a book that I will keep next to my computer for years to come."

—Michael Stokesbary, software engineer, Itron Inc.



About the Author

Mark Michaelis is an enterprise software architect at Itron Corporation and a trainer for Pluralsight. Mark holds a B.A. in philosophy from the University of Illinois and an M.S. in computer science from the Illinois Institute of Technology. He is a Microsoft MVP for the Visual Studio Team System, and he serves on several Microsoft software design review teams, including C# and VSTS. Mark speaks at developer conferences, both nationally and internationally, and has written several articles and books.

When not bonding with his computer, Mark is busy with his family, exercising outdoors, or traveling the globe, with particular affection for Africa. Mark lives in Spokane, Washington, with his wife Elisabeth, and three children, Benjamin, Hanna, and Abigail.

Acknowledgments

NO BOOK CAN BE published by the author alone, and I am extremely grateful for the multitude of people who helped me with this one.

The order in which I thank people is not significant, except for those that come first. By far, my family has made the biggest sacrifice to allow me to complete this. Benjamin, Hanna, and (most recently) Abigail often had a Daddy distracted by this book, but Elisabeth suffered even more so. She was often left to take care of things, holding the family's world together on her own. Thanks honey!

Many technical editors reviewed each chapter in minute detail to ensure technical accuracy. I was often amazed by the subtle errors that these folks still managed to catch:

Paul Bramsman
Doug Dechow
Gerard Frantz
Anson Horton
Angelika Langer
Nicholas Paldino
Narendra Poflee
Jon Skeet
Michael Stokesbary
John Timney
Stephen Toub

In particular, Michael was a huge help in editing the technical content and serving as a sounding board as I was putting the material together, not

to mention his invaluable friendship. I am also especially grateful to the C# MVPs (Nicholas and John), who know the language in certain areas second only to those on the C# team.

Thanks also to all those at Addison-Wesley, for their patience in working with me in spite of my frequent focus everywhere except on the manuscript. Thanks to:

Sheri Cain
Jessica D'Amico
Curt Johnson
Joan Murray
Stephane Nakib
Ann Wells

I especially appreciated Ann's frequent guidance and encouragement, starting back when the book was first suggested.

Prashant Sridharan, from Microsoft's Developer Division, was the one who got me started on this, and he provided me with an incredible jumpstart on the material. Thanks Prashant!



Foreword

FOR THOSE IN THE INDUSTRY who have watched the C programming language develop to embrace new concepts such as object programming (C++), ubiquity across platforms (Java), and highly structured models that form the basis of C#, it is clear that the language is well positioned to hold its own as a dominant development language.

This book has everything a developer needs to know to get up and running with the language. Many concise code samples, the use of mind maps to provide context, and the interesting way Mark contrasts C# with other languages make *Essential C# 2.0* stand out in a sea of C# books.

I have known Mark for a while now. His extensive involvement in the development of C# principles, together with his prior experience as an author of books on C++, COM+, and C#, uniquely qualify him to write the book that I believe will be a seminal reference for C#.

Further, Mark's teaching experience allows him to combine the world of technical skill with that of practical, intuitive tutorial. It begins with principles of object programming in C#, as it is a reference to verify advanced multithreading techniques in a complex system. The reader, novice or expert, will enjoy his casual style and method of eloquently communicating challenging topics.

This reference and tutorial should be on the shelf next to your workstation and carried in your backpack. Pick it up and read it! I'm confident

that whether you are new or experienced with the language, you will learn something in this book that will induce an “aha!” moment!

—*Prashant Sridharan*
Microsoft Corporation
March 2006

前 言

纵观软件工程的历史，编写计算机程序的方法论经历了几次变革，每一次都在原来的基础上提高代码的组织性并降低其复杂性。本书将引领读者经历同样的发展过程。

开篇的几章引领读者回顾**顺序程序设计**，即语句按照它们的执行顺序编写。这种模式的问题在于，随着需求的增长，复杂性将呈指数级增长。为了降低复杂性，出现了代码块的方法，由此产生了**结构化程序设计模型**，允许在一个程序中的多个位置调用同一个代码块，而不需重复代码。即使在这种构架下，程序也很快变得笨拙而需要进一步改进，由此产生了面向对象程序设计，这将在第5章讨论。在后续章节中，读者将进一步学习编程方法，诸如基于接口编程和尚未完全成熟的通过属性进行的声明性编程（将在第4章讨论）。

本书有三个主要功能。

- 全面覆盖 C# 语言知识，超越了一般性的指南，为读者开始高效软件开发工程打下坚实基础；
- 对于已经熟悉 C# 的读者，可以通过本书来深入理解复杂的编程范例和最新的 C# 2.0 的主要特性；
- 可长期参考，即使在精通 C# 2.0 以后，仍会有所帮助。

成功掌握 C# 2.0 的关键在于尽可能快地着手编写代码。马上开始编写软件吧，不要等到成为理论“专家”以后。作为迭代开发的拥护者，我希望本书能使读者在第二章结尾的时候就能开始编写基本的 C# 代码，即使是初学者。

本书并未囊括 C# 所有相关主题，如 ASP.NET、ADO.NET、smart client development（智能客户端开发）、distributed programming（分布式编程）等。虽然这些主题和 .NET 框架相关，但要真正掌握它们需要专门的书籍。阅读本书能使读者准备好所有这些领域的相关知识。本书专注于讲解 C# 和基础类库中的类型。

本书的目标读者

本书最大的挑战是如何既不使高级开发者觉得无聊，又不让初学者看到 assembly、link、chain、thread 和 fusion 等术语而不知所云，让这本书看起来更像是为铁匠而不是程序员写的。本书主要面向有经验而又想多掌握一种语言的开发者。不过，这本书经过了精心的设计，相信对各个阶段的开发者都是很有价值的。

- **初学者：**本书能帮助编程新手成功地从入门级程序员过渡到 C# 开发者，顺利完成所遇到的任何 C# 编程工作。这本书不仅教授语法，同时也通过良好的编程实践来提供训练，这些实践能使读者在整个编程生涯中受益匪浅。
- **结构化语言的程序员：**就像学习外语最好要沉浸其中一样，学习一种计算机语言最有效的方式就是在了解所有的复杂性之前就开始使用它。按照这种方式，熟悉结构化语言的读者会很轻松地理解这本书开篇的指南，并且到了第 4 章结尾，这些读者就能够驾轻就熟地编写基本控制程序了。但是，成为卓越 C# 开发者的关键并不是记住语法。要实现从简单程序到企业级开发的转变，C# 开发者必须本能地考虑对象和它们之间的联系。为此，第 5 章开篇介绍了类和面向对象开发。传统的结构化语言如 C、COBOL 和 FORTRAN 等仍扮演很重要的角色，但已不如从前了，所以软件工程师应该熟悉面向对象开发。C# 是进行这种转变的理想语言，因为它是以面向对象开发为一个核心原则来设计的。
- **基于对象和面向对象开发者：**C++ 和 Java 程序员，以及许多有经验的 Visual Basic 程序员就属于此类。他们中的大多数已经完全适应了那些分号和大括号。第 1 章的一段简短代码集中体现了这一点，C# 与他们早已熟悉的 C 和 C++ 风格很相似。
- **专业 C# 程序员：**对这些已经精通 C# 的人来说，本书是对那些不太常用语法的方便的参考材料，同时还提供了对那些语言细节和不常涉及的微妙之处的解析。最重要的是，书中提出了编写健壮和可维护代码的指导方针和模式。此外，本书还可以用于向其他人传授 C# 语言。

本书针对 C#2.0 做了大量增订，其中一些最重要的有：

- 分部类 (Partial class)，参见第 5 章；
- 全局命名空间限定词 (Global Namespace Qualifier)，参见第 9 章；
- 属性的 getter 和 setter 方法的不同访问修饰符，参见第 5 章；
- 匿名方法 (Anonymous method)，参见第 13 章；
- 泛型 (Generic)，参见第 11 章；
- 迭代子 (Iterator) 相关主题，参见第 12 章。

如果对这些主题还不太熟悉，书中对它们作了详尽的阐述。第 17 章中关于指针的论述，即使是对那些高级 C# 开发者来说也是很有价值的——有经验的 C# 开发者也常常不能很好地理解指针。

本书特色

本书致力于详尽地阐述 C#2.0 的核心特性。为了帮助读者真正理解 C# 语言的内涵，在讲解每种特性时都辅以大量实例。每个概念都辅以指导方针和最佳实践，以确保代码能够正确编译，避免可能的缺陷并获得最大的可维护性。

为使本书易于阅读，书中代码采用了专门的格式，并采用知识地图的形式概括出每一章的主要内容。

代码示例

书中的绝大部分代码都能在任何通用语言基础结构 (Common Language Infrastructure, CLI) 的

实现中运行，包括 Mono、Rotor 和 Microsoft .NET 平台。Platform- 或 vendor-specific 库很少用到，除非涉及只和那些平台相关的重要概念(如适当地处理单线程 Windows 用户接口)。明确需要 C# 2.0 compliance 的代码都在“Appendix C: C# 2.0 Topics”中列出。

下面列出文中的一段程序清单作为示例。

```

class CommentSamples
{
    static void Main()
    {
        Single-Line Comment
        string firstName; // Variable for storing the first name
        string lastName; // Variable for storing the last name

        System.Console.WriteLine("Hey you!");

        Delimited Comment Inside Statement
        System.Console.Write /* No new line */ (
            "Enter your first name: ");
        firstName = System.Console.ReadLine();

        System.Console.Write /* No new line */ (
            "Enter your last name: ");
        lastName = System.Console.ReadLine();

        /* Display a greeting to the console
           using composite formatting. */ } Delimited Comment
        System.Console.WriteLine("Your full name is {0} {1}.",
            firstName, lastName);
        // ...
    }
}

```

格式如下。

- 注释使用斜体 (italics)。

```

/* Display a greeting to the console
   using composite formatting. */

```

- 关键字使用粗体 (bold)。

```

static void Main()

```

- 突出显示的是特定代码片段，可能是修改原来程序清单中的代码得到的，或示范了文中描述的思想。

```

    System.Console.Write /* No new line */ (

```

- 突出显示可能是整行，也可能是一行中的几个字符。

```

    System.Console.WriteLine(
        "Your full name is {0} {1}.",

```

- 在不完全的程序清单中，用省略号表示忽略的与书中主题无关的代码。

```

    // ...

```

- 控制台输出表示特定程序清单的输出，出现在此程序清单之后。

```
>HeyYou.exe
Hey you!
Enter your first name: Inigo
Enter your last name: Montoya
```

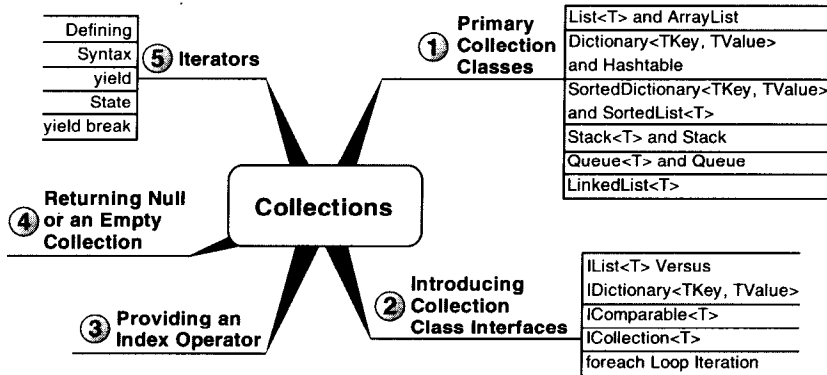
程序中用户输入的部分使用斜体。

虽然提供完整的代码示例可以方便读者将它们拷贝到自己的程序中，但这样做会使读者不能集中精力学习特定主题。这样，读者需要经过调试才能将代码示例加入到自己的程序中。省略的部分最主要的是错误检查，如异常处理。代码示例中也没有明确包含使用的系统状态，读者需要自己设想所有示例的状态。

代码示例可以从 <http://mark.michaelis.net/EssentialCSharp> 下载。

知识地图

每一章开篇都提供一个知识地图，提供了每章内容的概览。举一例如下（摘自第 12 章）。



知识地图的中央是每章的主要内容，周围的分支是其下的高级主题。知识地图使读者可以很容易地了解本章中有哪些高级主题及其更具体的内容，并避开那些不感兴趣的特定内容。

特别提示

根据读者的经验水平，特定的代码段和提示有助于更好地理解正文。

- **Beginner Topic**（初级话题）为入门级程序员提供解释说明。
- **Advanced Topic**（高级话题）使有经验的程序员将注意力集中在适合他们的内容上。
- **Callout note**（警示）在警示框中强调关键原则，使读者容易认识到它们的重要性。
- **Language Contrast**（语言对比）帮助熟悉其他编程语言的读者认识到 C# 与这些编程语言的关键不同。

本书的结构

宏观上看，软件工程是关于复杂性管理的，因此我用 1~4 章来介绍结构化编程语言，使读者能

够马上写出具有简单功能的代码。第 5~9 章介绍 C#语言的面向对象架构,初学者在进行本书余下的高级内容之前应先专注于彻底理解这一部分。

第 11~13 章介绍额外降低复杂性的架构,处理实际上为所有现代编程语言所需的模式。由此产生了关于反射和属性的动态规划,被广泛运用于线程和互操作性中,在后面的章节会提及。

本书最后一章讨论通用语言基础结构,描述与开发平台相关的 C#。将这一章放在书末使因为这些内容并非 C#特性,并区别于本书其它部分的语法与编程风格。不过,本章适合随时阅读,可能在第 1 章结束后立即阅读最为合适。

下面介绍每章主要内容:

- **第 1 章 Introducing C#:** 介绍第 1 个 C# 程序之后,本章继续对其进行剖析,使读者熟悉本书,对 C#程序有一个感性的认识,并展示了如何编译和调试程序的细节。此外,还涉及了 C# 程序的执行及其中间语言 (intermediate language)。
- **第 2 章 Data Types:** 程序需要处理数据才能正常运行。本章介绍 C#的简单数据类型,包括值类型和引用类型,同时介绍了类型之间的转化和对数组的支持。
- **第 3 章 Operators and Control Flow:** 为了利用计算机的循环能力,需要懂得如何在程序中加入循环和条件逻辑。本章还介绍了 C#运算符、数据转换和预处理程序指令。
- **第 4 章 Methods and Parameters:** 本章细致研究方法及其参数,包括 passing by value、passing by reference 和通过参数返回的数据。C#不支持缺省参数,本章解释这样做的原因,并介绍如何实现同样的功能。
- **第 5 章 Classes:** 前面已经介绍了类的基本构造块,本章将这些结合成完整功能的类型。类通过为对象定义模版,构成了面向对象技术的核心。
- **第 6 章 Inheritance:** 虽然对许多开发者来说,继承是编程的基础,但 C#有一些独特构造,例如新的修饰符 (modifier.)。本章详细论述继承语法 (inheritance syntax),包括覆盖 (overriding)。
- **第 7 章 Interfaces:** 本章论述接口是怎样用来定义“无版本冲突的 (versionable)”类间交互协定的。C#包含显式和隐式的接口成员实现,这使 C#能够达到其他大多数语言无法支持的封装水平。
- **第 8 章 Value Types:** 虽然不像定义引用类型那样普遍,但定义作为 C#基本数据类型的值类型有时还是必要的。本章描述如何定义结构,并揭示它们可能引入的特性。
- **第 9 章 Well-Formed Types:** 本章讨论更高级类型定义,解释如何执行+和 cast 等运算符,以及如何将多个类封装到一个库中。此外,本章还示范了命名空间和 XML 注释的定义,并讨论了有关垃圾回收器的类的定义。
- **第 10 章 Exception Handling:** 本章在第 4 章基础上进一步介绍了异常处理,描述异常是如何随支持自定义异常的层次结构而产生的。本章还包括一些异常处理的最佳实践。
- **第 11 章 Generics:** 泛型 (Generic) 可能是 C# 1.0 所缺少的核心特性。本章详细阐述了这个新特性。
- **第 12 章 Collections:** 有了泛型, .NET 1.1 中所有的集合类都可以用它们的等效泛型代替。本章回顾了集合类和定义集合类一般行为的接口。在基本运行时冲突最小的情

况下, C# 2.0 提供了利用迭代子更容易地创建集合类的语法。迭代子提供了定义类中数据如何循环的清晰语法。

- **第 13 章 Delegates and Events:** 委托 (Delegate) 通过定义处理代码中事件的模式, 首先将 C# 同它的前辈区分开。这实际上排除了编写投票例程的需要。经过封装的委托, 也就是事件, 是通用语言运行时的基本构造之一。本章还介绍了匿名方法和其他 C# 2.0 特性。
- **第 14 章 Reflection and Attributes:** 上世纪 80 年代末, 面向对象编程成为编程结构中范式转变 (Paradigm shift) 的基础。同样, 属性促进了声明式编程和嵌入元数据, 是一种新范式的开始。本章介绍属性, 并讨论如何通过映射来检索, 还阐述通过基本类库中的序列化框架的文件输入、输出。
- **第 15 章 Multithreading:** 大多数现代编程需要利用线程来在保证及时响应并发任务的同时执行长线任务。随着程序的日趋复杂, 必须特别注意在复杂环境中保护数据。编写多线程应用程序是非常复杂的。本章讨论如何运用线程, 并提供避免危害多线程程序的问题的最佳实践。
- **第 16 章 Multithreading Patterns:** 在上一章的基础上, 本章示范了一些能够简化线程代码显式控制的内建线程模式。
- **第 17 章 Platform Interoperability and Unsafe Code:** 既然 C# 是一种相对年轻的语言, 那么用其他语言编写的代码也就远远多于 C# 代码。为用这些代码, C# 支持互操作性——通过平台调用 (P/Invoke) 来调用非托管代码 (unmanaged code)。此外, C# 还支持指针和直接内存操作。虽然带有指针的代码需要特殊运行权限, 但同时也提供了与传统 C 语言编程接口的完全互操作能力。
- **第 18 章 The Common Language Infrastructure:** C# 本质上是基于通用语言基础结构编写的高效编程语言。本章深入探讨 C# 程序如何与基本运行时相关, 以及 C# 规范。
- **Appendix A Downloading and Installing the C# Compiler and the CLI Platform:** 本附录指导读者安装 C# 编译器和运行代码的平台 (.NET 或 Mono)。
- **Appendix B Complete Source Code Listings:** 在某些章节的示例中, 如果列出完整的源代码会占用过长的篇幅。为了使它们对读者来说完整可用, 本附录收录了第 3 章、第 11 章、第 12 章、第 14 章和第 17 章中示例的完整代码。
- **Appendix C C# 2.0 Topics:** 本附录提供了书中 C# 2.0 主题的快速索引, 专为 C# 1.0 程序员设计, 帮助他们快速查询 2.0 特性。

我希望这本书能成为读者掌握 C# 专业技能的重要资源, 并在读者继续探索 C# 深层技术和内部工作机制时提供参考。



Contents

1	Introducing C#	1
	Hello, World	2
	<i>Compiling and Running the Application</i>	3
	<i>Managed Execution and the Common Language Infrastructure</i>	4
	C# Syntax Fundamentals	7
	C# Keywords	7
	Type Definition	9
	Main	10
	Statements and Statement Delimiters	12
	Whitespace	13
	Working with Variables	14
	Data Types	15
	Declaring a Variable	16
	Assigning a Variable	16
	Using a Variable	18
	Console Input and Output	18
	Getting Input from the Console	18
	Writing Output to the Console	20
	Comments	22
	Common Intermediate Language and ILDASM	25