信息素养文库 · 高等学校信息技术系列课程规划教材

C语言程序设计

◎主　编　马　杰　刘　艳　杨　磊
◎副主编　刘　啸　蔡　键　刘一秀　顾海霞
◎主　审　周晓云

南京大学出版社

信息素养文库 · 高等学校信息技术系列课程规划教材

C语言程序设计

◎ 主　编 马　杰　刘　艳　杨　磊
◎ 副主编 刘　啸　蔡　键　刘一秀　顾海霞
◎ 主　审 周晓云

【微信扫码】
本书导学，领你入门

南京大学出版社

图书在版编目(CIP)数据

C语言程序设计 / 马杰, 刘艳, 杨磊主编. — 南京 : 南京大学出版社, 2018.1
(信息素养文库)
高等学校信息技术系列课程规划教材
ISBN 978-7-305-19634-8

Ⅰ. ①C… Ⅱ. ①马… ②刘… ③杨… Ⅲ. ①C语言—程序设计—高等学校—教材 Ⅳ. ①TP312.8

中国版本图书馆 CIP 数据核字(2017)第 284477 号

出版发行　南京大学出版社
社　　址　南京市汉口路 22 号　　邮　编　210093
出 版 人　金鑫荣

丛 书 名　信息素养文库·高等学校信息技术系列课程规划教材
书　　名　C语言程序设计
主　　编　马杰　刘艳　杨磊
责任编辑　陈亚明　王南雁　　编辑热线　025-83597482

照　　排　南京南琳图文制作有限公司
印　　刷　南京大众新科技印刷有限公司
开　　本　787×1092　1/16　印张 18　字数 445 千字
版　　次　2018 年 1 月第 1 版　2018 年 1 月第 1 次印刷
ISBN 978-7-305-19634-8
定　　价　44.80 元

网址：http://www.njupco.com
官方微博：http://weibo.com/njupco
官方微信号：njupress
销售咨询热线：(025) 83594756

前　言

一个人接受科技教育的最大收获，是那些能够受用一生的通用智能工具。

——George Forsythe

随着社会信息化程度的不断加深，计算机已然成为现代人必不可少的工具，或者说是学习生活中不可或缺的好伙伴、好助手。收集资料、深入学习、提高工作效率和质量等，很多时候取决于个人对计算机的掌握程度。

对于生活在高度信息化的当今社会的一名大学生，要真正学好计算机，让计算机更好地为我所用，仅是从广度上了解计算机世界、能够简单地使用计算机是远远不够的。

《C语言程序设计》课程是国内外高等学校所开设的，继《大学计算机基础》之后的一门重要的计算机基础课程。通过该课程的学习：

1. 可以掌握一种高级程序设计语言的基本语法。本书详尽的列出了C语言的相关语法知识，并用实例、图示等生动地讲解各知识点的原理。

2. 可以深入地学习并探究利用计算机解决问题的方法。本书从简单实例到复杂实例，都试图从计算机解决问题的局限性出发，讲解程序的基本方法，进而有效地利用计算机高效地解决问题。

3. 可以用来很好地备考全国计算机等级考试。本书以全国等级考试为指导，紧随全国等考系统的最新变化，由原来的Visual C++6.0到现在的Visual C++2010；本书在第一章特别介绍了Visual C++2010的操作和使用过程，内容精练，结构合理，便于自学，可以极大地减轻读者学习C语言的困难，是备考学员较好的选择。

本书主要由以下12章内容构成：

第1章　程序设计概述，介绍了计算机思维和程序设计的关系以及算法的描述方法，详细讲解了编译环境操作过程，重点掌握流程图的算法描述形式。

第2章　数据类型和运算符，从一个程序出发，介绍了一个C语言程序的基本构成，数据类型、变量、常量和表达式算术符的用法以及数学表达方法和计算机表达方式的不同，重点掌握计算机的思维方式。

第3章　数据的输入和输出，介绍了C语言程序和外界进行信息交流的方式：输入和输出语句，重点解释了printf和scanf的使用。

第4章　选择结构，介绍了选择结构的思想，讲解了关系运算符、逻辑运算符以及条件运算符的使用，重点说明了单分支、双分支和多分支结构的用法以及switch语句的使用。

第5章　循环结构程序设计，介绍了循环结构和三种循环语句的语法构成，从计数循

环、条件循环和循环嵌套讲解了三种循环语句的使用，以及 break 和 continue 语句对循环的控制作用。

第 6 章　函数，介绍了函数的定义、声明和调用的方式，分析了函数传值调用以及函数返回值的使用方法，重点讲解了递归函数的定义和执行过程，以及变量作用域的相关知识点。

第 7 章　数组，介绍了一维数组、二维数组的定义、引用以及初始化的方法，通过排序算法强化了数组的使用，详细讲解了函数中数组做出参数的定义和调用方式。

第 8 章　指针，介绍了指针的概念以及指针变量定义、初始化和易用的方法，通过函数的调用强化指针的使用。

第 9 章　指针和数组，介绍了指针和数组的关系，重点从地址和值的角度分析了指针运算符的使用方式，讲解了指针数组、数组指针、指针函数和函数指针这两对易混概念。

第 10 章　字符串，介绍了字符、字符串、字符数组三个概念的区别以及字符串的存储和输入输出的相关内容，分析了字符串的处理函数，又重点解释了指针数组在处理字符串问题的优势和不同，详细描述字符串基本操作函数实现。

第 11 章　结构体和共用体，介绍了结构体、共同体、枚举类型的定义以及变量、数组、指针的定义和使用方式，重点解释了链表的相关操作。

第 12 章　文件，介绍了文件操作的主要两种模式：读和写的方法，以及其中使用的主要操作函数。

本书由马杰、刘艳、杨磊担任主编，编写组成员有刘啸、蔡键、刘一秀、顾海霞等。周晓云教授担任主审，并给予了具体指导。此外，为了加强学生的实践能力，本书配套了实践教程，由顾海霞，刘一秀，周晓云主编，南京大学出版社出版。本书还配套有不少网络资源，内容包括导学、习题解答、附录，其他资源等，覆盖相关章节，能够让学习者随时随地用手机观看。这些网络资源以二维码的形式在书中呈现，无需下载与注册，只需用微信扫描即可查阅。

本书适用于不同层次包括计算机专业和非计算机专业，零基础和有一些基础的读者。读者可根据自身情况选读书中内容。

由于作者水平有限，书中难免会有错误和不足之处，敬请读者批评指正。

编　者

2017 年 12 月

目 录

第1章 程序设计概述

1.1 计算思维与程序设计

计算思维，一个看似遥远又抽象的概念，其中却蕴含着丰富的人生智慧。在我们的生活中，小到洗衣做饭，大到公司决策，都与计算思维息息相关、紧密相连。计算思维究竟是什么？如何更好地掌握与运用这一能力？

思维是人脑对客观事物的一种概括的、简洁的反应，它反映客观事物的本质和规律。科学思维是指理性认识及其过程，即经过感性阶段获得的大量材料，通过整理和改造，形成概念、判断和推理，从而反映事物的本质和规律。现阶段我们把科学思维分为三类：理论思维、实验思维和计算思维。

计算思维的概念是由美国卡内基·梅隆大学的周以真教授首先提出的。她认为计算思维能够将一个问题清晰、抽象地描述出来，并将问题的解决方案表示为一个信息处理的流程。它是一种解决问题切入的角度。计算思维包含了数学性思维和工程性思维，而其最重要的思维模式就是抽象话语模式。

运用计算机科学的思想、方法和技术进行问题求解、系统设计，以及人类行为理解等涵盖计算机科学广度的一系列思维活动，这就是计算思维。计算思维的核心是算法思维。

周以真教授于2006年在《美国计算机学会通讯》上发表了《计算思维》(Computational Thinking，也称计算性思维)一文，将计算思维作为一种基本技能和普适思维方法提出。它的运用将引导计算机教育工作者、研究者和实践者去推动社会变革，不仅仅限于计算机领域，如当前各个行业领域中面临的大数据问题，就需要依赖于计算思维来挖掘有效内容，这意味着计算机科学将从前沿变得更加基础和普及。

计算科学这个领域提供的思维模式，对所有的领域、职业都是适用的，都是能够从中受益的。对于学生而言，学一点算法、计算机编程，抽象化的这种技巧，对以后从商、搞法律、学医或者是自己创业，都会比那些没有学过计算机科学的人要更强，要更加有优势。这是因为学习抽象的语言和算法，你就会有一种新的解决问题的技能。所以通过计算机程序设计课程培养初步的计算能力，了解经典的一系列算法，是培养初步的计算思维的有效途径。

1.2 算法

1.2.1 算法简介

之前，我们提到计算思维的核心是算法思维。那什么是算法呢？

广义地说，为解决一个问题而采取的方法和步骤，就称为算法。不要认为只有“计算”的问题才有算法，任何问题的解决步骤都可称为算法。

我们更关心的计算机算法可以分为两大类别：数值运算算法和非数值运算算法。数值运算算法的目的是求解数值，例如求方程的根、求最大公约数等。非数值运算涉及的领域十分广泛，最常见的是用于事务管理领域，例如检索、人事管理等。

计算机算法有如下特性：

1. 有穷性：一个算法在执行有穷步骤之后必须结束。也就是说，一个算法，它所包含的计算步骤是有限的。

2. 确定性：算法的每一个步骤必须要确切地定义。即算法中所有有待执行的动作必须严格而不含混地进行规定，不能有歧义性。

3. 输入：算法有零个或多个的输入，即在算法开始之前，最初给出的量。

4. 输出：算法有一个或多个的输出，即与输入有某个特定关系的量，简单地说就是算法的最终结果。

5. 可操作性：算法上描述的操作在计算机上都是可以实现的。

1.2.2 如何描述算法

算法是为了解决实际问题的，问题简单，算法也简单；问题复杂，算法也相应复杂。为了便于交流和算法处理，往往需要将算法进行描述，即算法的表示。为了表示一个算法，可以用不同的方法。常用的表示方法有：自然语言、传统流程图、结构化流程图、伪代码、PAD 图等。这里我们主要介绍自然语言表示、传统流程图表示和伪代码表示。

1. 自然语言表示

所谓自然语言，就是自然地随文化演化的语言，如英语、汉语等。通俗地讲，自然语言就是我们平时口头描述的语言。对于一些简单的算法，可以采用自然语言来口头描述算法的过程。

【例 1.1】 求 5! 的结果，用自然语言描述算法

可以设两个变量：p 为被乘数，i 为乘数。将每一次的乘积放在被乘数变量中。使用循环思路来求结果，算法可描述为：

S1：使 p=1。

S2：使 i=2。

S3：将 p×i，乘积仍放在变量 p 中，可表示为：p= i×p。

S4：使 i 的值加 1，即 i=i+1。

S5：如果 i 不大于 5，返回重新执行步骤 S3 以及其后的步骤 S4 和 S5；否则，算法结束。

最后得到 p 的值就是 5! 的值。

随着需求的发展，很多算法都比较复杂，很难用自然语言描述，同时自然语言的表述繁琐难懂，不利于发展和交流。因此，需要用其他的方法来进行表示。

2. 流程图表示

流程图是一种用图形表示算法流程的方法，其由一些图框和流向线组成，如图 1－1 所示。

其中，图框表示各种操作的类型，图框中的说明文字和符号表示该操作的内容，流向线表示操作的前后次序。流程图的最大优点是简单直观、便于理解，在计算机算法领域有着广泛的应用。

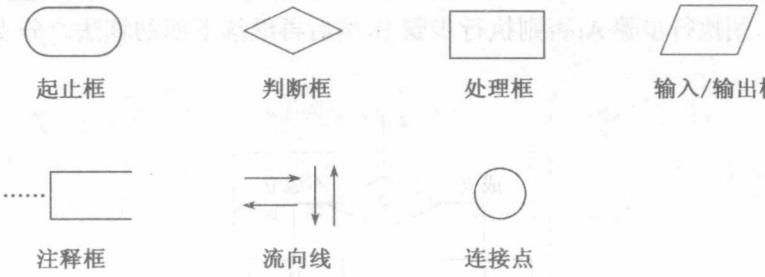

图 1－1　流程图的图元

【例 1.2】　求 5! 的结果，用流程图描述算法。

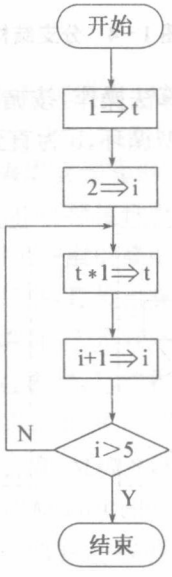

图 1－2　求 5! 的流程图

在实际使用中，算法中的各种流程往往可以用如下三种基本流程结构组合而成，这三种基本流程结构是：顺序结构、分支结构和循环结构。

顺序结构是最简单的一种流程结构，即一个接一个进行算法步骤的处理，如图 1－3 所示。

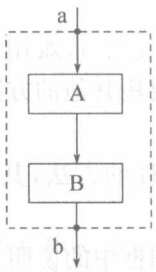

图 1-3　顺序结构

分支结构常用于根据某个条件来决定算法的走向，如图 1-4 所示。这里先判断条件 P，如果 P 成立，则执行步骤 A；否则执行步骤 B，然后再继续下面的算法。分支结构有时也称为选择结构。

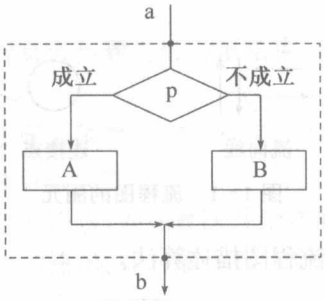

图 1-4　分支结构

循环结构常用于需要反复执行的算法操作，按循环的方式，可以分为当型循环结构和直到型循环结构，如图 1-5 所示，a 为当型循环，b 为直到型循环。

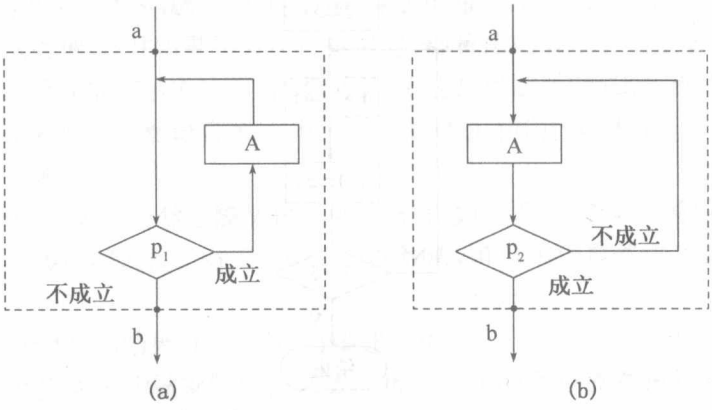

图 1-5　两种循环结构

当型循环和直到型循环结构的区别：当型循环结构先对条件进行判断，然后再执行循环体，一般采用 while 语句来实现，直到型循环结构先执行循环体，然后再对条件进行判断，一般采用 do…while 语句来实现。

3. 伪代码表示

伪代码是用介于自然语言和计算机语言之间的文字和符号来描述算法。它如同一篇文章一样，自上而下地写下来。每一行（或几行）表示一个基本操作。它不用图形符号，因此书写方便、格式紧凑，也比较好懂，也便于向计算机语言算法（即程序）过渡。适用于设计过程中需要反复修改时的流程描述。

【例 1.3】 求 5! 的结果，用伪代码描述算法。

```
对变量 t 赋值 1
对变量 i 赋值 2
while(i<=5){
    使 t=t×i
    使 i=i+1
}
输出 t 的值
```

在使用伪代码时，描述应该结构清晰、代码简单、可读性好，这样才能够更有利于算法的表示；否则，将适得其反，让人很难懂，就失去了伪代码的意义了。

1.3 C语言概述

1.3.1 C语言的出现和历史背景

C语言是国际上广泛流行的计算机高级语言，它适合作为系统描述语言，既可以用来编写系统软件，也可以用来编写应用软件。

早期的操作系统软件主要是用汇编语言编写的（包括 UNIX 操作系统在内）。由于汇编语言依赖于计算机硬件，程序的可读性和可移植性都比较差，所以为了提高系统软件的可读性和可移植性，最好改用高级语言。但是一般的高级语言难以实现汇编语言的某些功能（汇编语言可以直接对硬件进行操作，例如对内存地址的操作、位操作等）。人们希望找到一种兼具一般高级语言和低级语言优点的语言，于是，C语言就在这种情况下应运而生了。

C语言是在B语言的基础上发展起来的，它的根源可以追溯到 ALGOL60。1960 年出现的 ALGOL60 是一种面向问题的语言。1963 年英国剑桥大学推出了 CPL（Combined Programming Language）语言。CPL 语言在 ALGOL60 的基础上接近硬件一些，但规模比较大，难以实现。1967 年英国剑桥大学的 Martin Richards 对 CPL 语言做了简化，推出了 BCPL（Basic Combined Programming Language）语言。1970 年美国贝尔实验室的 Ken Thonpson 以 BCPL 语言为基础，又做了进一步简化，设计出了很简单的而且很接近硬件的 B 语言（取 BCPL 的第一个字母）。

1971 年在 PDP11/20 上实现了 B 语言，UNIX 操作系统，此时的 B 语言过于简单，功能有限。1972 年至 1973 年间，贝尔实验室的 D.M.Ritchie 在 B 语言的基础上设计出了 C 语言（取 BCPL 的第二个字母）。C 语言既保持了 BCPL 和 B 语言的优点（精炼，接近硬件），又克服了它们的缺点（过于简单，数据无类型等）。最初的 C 语言只是为描述和实现 UNIX 操作系统

提供一种工作语言而设计的。1973 年，Ken Thompson 和 D.M. Ritchie 合作把 UNIX 的 90% 以上用 C 语言改写(UNIX 第 5 版，原来的 UNIX 操作系统是 1969 年由美国的贝尔实验室的 Ken Thompson 和 D.M. Ritchie 开发成功的，是用汇编语言编写的)。

后来，C 语言多次做了改进，但主要还是贝尔实验室内部使用。直到 1975 年 UNIX 第 6 版发布后，C 语言的突出优点才引起了人们的普遍注意。1977 年出现了不依赖于具体机器的 C 语言编译程序“可移植 C 语言编译程序”，使 C 语言移植到其他机器时所需做的工作大大简化了，这也推动了 UNIX 操作系统迅速地在各种机器上实现。例如 VAX、AT&T 等计算机系统都相继开发了 UNIX。随着 UNIX 的日益广泛使用，C 语言也迅速得到推广。C 语言和 UNIX 可以说是一对孪生兄弟，在发展过程中相辅相成。1978 年以后，C 语言先后移植到大、中、小、微型计算机上，已独立于 UNIX 和 PDP 了。C 语言从此风靡全世界，成为世界上应用最广泛的几种语言之一。

以 1978 年发布的 UNIX 第 7 版中的 C 语言编译程序为基础，Brian W. Kernighan 和 Dennis M. Ritchie(合称 K&R)合著了影响深远的名著 *The C Programming Language*，这本书中介绍的 C 语言成为后来广泛使用的 C 语言版本的基础，它成为事实上的 C 标准。1983 年，美国国家标准化协会(ANSI)根据 C 语言问世以来各种版本对 C 语言的发展和扩充，公布了第一个 C 语言标准草案(83 ANSI C)。ANSI C 比原来的 C 语言有了很大的发展。K&R 在 1988 年修改了他们的经典著作 *The C Programming Language*，按照即将公布的 ANSI C 重新写了该书。1989 年，ANSI 公布了一个完整的 C 语言标准——X 3.159.1989，简称 C89。1990 年，国际标准化组织 ISO(Internatiomal Standard Organization)接受 C89 为 ISO C 的标准(ISO9899—1990)，通称 C90。C90 与 C89 基本相同。1999 年，ISO 又修订了 C 语言标准，简称 C99。目前流行的 C 语言编译系统大多是以 C89 为基础进行开发的，并未实现 C99 所建议的功能。不同版本的 C 语言编译系统所实现的语言功能和语法规则又略有差别，因此读者应了解所用的 C 语言编译系统的特点(可参阅有关手册)。本书基本上以 C89 为基础并结合当前常用的 C 编译系统的情况进行介绍。

1.3.2 C语言的特点

早期的 C 语言主要是用于 UNIX 系统。由于 C 语言的强大功能和各方面的优点逐渐为人们认识，到了八十年代，C 开始进入其他操作系统，并很快在各类大、中、小和微型计算机上得到了广泛使用，成为当代最优秀的程序设计语言之一。C 语言主要有以下特点：

1. 语言简洁，使用方便灵活

C 语言是现有程序设计语言中规模最小的语言之一，而小的语言体系往往能设计出较好的程序。C 语言的关键字很少，ANSI C 标准一共只有 32 个关键字，9 种控制语句，压缩了一切不必要的成份。C 语言的书写形式比较自由，表达方法简洁，使用一些简单的方法就可以构造出相当复杂的数据类型和程序结构。

2. 可移植性好

用过汇编语言的读者都知道，即使是功能完全相同的一种程序，对于不同的单片机，必须采用不同的汇编语言来编写。这是因为汇编语言完全依赖于单片机硬件。而现代社会中新器件的更新换代速度非常快，也许我们每年都要跟新的单片机打交道。如果每接触一种新的单片机就要学习一次新的汇编语言，那么也许我们将一事无成，因为每学一种新的汇编

语言，少则几月，多则几年，那么我们还有多少时间真正用于产品开发呢？

C语言是通过编译来得到可执行代码的，统计资料表明，不同机器上的C语言编译程序80%的代码是公共的，C语言的编译程序便于移植，从而使在一种单片机上使用的C语言程序可以不加修改或稍加修改即可方便地移植到另一种结构类型的单片机上去。这大大增强了我们使用各种单片机进行产品开发的能力。

3. 表达能力强

C语言具有丰富的数据结构类型，可以根据需要采用整型、实型、字符型、数组类型、指针类型、结构类型、联合类型、枚举类型等多种数据类型来实现各种复杂数据结构的运算。C语言还具有多种运算符，灵活使用各种运算符可以实现其他高级语言难以实现的运算。

4. 表达方式灵活

利用C语言提供的多种运算符，可以组成各种表达式，还可采用多种方法来获得表达式的值，从而使用户在程序设计中具有更大的灵活性。C语言的语法规则不太严格，程序设计的自由度比较大，程序的书写格式自由灵活。程序主要用小写字母来编写，而小写字母是比较容易阅读的，这些充分体现了C语言灵活、方便和实用的特点。

5. 可进行结构化程序设计

C语言是以函数作为程序设计的基本单位的，C语言程序中的函数相当于汇编语言中的子程序。C语言对于输入和输出的处理也是通过函数调用来实现的。各种C语言编译器都会提供一个函数库，其中包含许多标准函数，如各种数学函数、标准输入输出函数等。此外，C语言还具有自定义函数的功能，用户可以根据自己的需要编制满足某种特殊需要的自定义函数。实际上C语言程序就是由许多个函数组成的，一个函数即相当于一个程序模块，因此C语言可以很容易地进行结构化程序设计。

6. 可以直接操作计算机硬件

C语言具有直接访问单片机物理地址的能力，可以直接访问片内或片外存储器，还可以进行各种位操作。

7. 生成的目标代码质量高

众所周知，汇编语言程序目标代码的效率是最高的，这就是为什么汇编语言仍是编写计算机系统软件的重要工具的原因。但是统计表明，对于同一个问题，用C语言编写的程序生成代码的效率仅比用汇编语言编写的程序低10%~20%。

尽管C语言具有很多的优点，但和其他任何一种程序设计语言一样也有其自身的缺点，如不能自动检查数组的边界、各种运算符的优先级别太多、某些运算符具有多种用途等。但总体来说，C语言的优点远远超过了它的缺点。经验表明，程序设计人员一旦学会使用C语言，就会对它爱不释手。

1.3.3　VC++2010简介和C语言程序结构

1. 用什么工具写代码

程序设计是门交流的艺术，和我们交流的对象是计算机。我们要做的是在代码里面说清楚想要计算机做出怎样的操作。其实写代码就像平时写文章一样，只是在电脑上写一些文本内容，那用什么工具来写代码呢？平时我们在Windows中写文章，可以用记事本、Word、UltraEdit等文本编辑工具。在C语言程序的实际开发中，为了提高开发效率，一般会

使用可视化开发工具平台，可视化开发工具平台的好处是对程序开发的过程进行智能化的支持，如语法检查、代码自动缩进等。VC++2010 作为成熟的 C 语言可视化开发平台，系统稳定、开发效率高，并且是各级考试中选择的考试环境平台，是我们一开始接触 C 语言程序设计的第一选择，如图 1－6 所示。

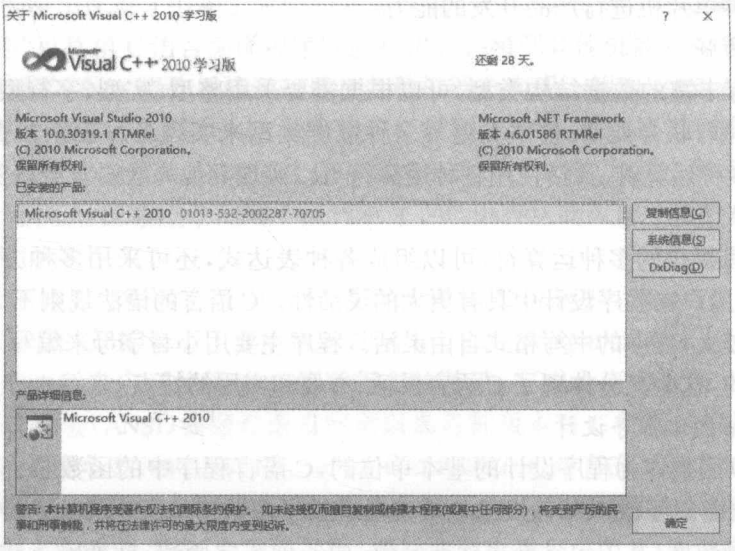

图 1－6　VC++2010

2. 使用 VC++2010 设计一个 C 程序

(1) 进入 VC++2010 熟悉界面环境

可以通过以下方法进入 VC++2010 可视化集成开发平台。

打开 windows 开始菜单，点击“所有程序”，在程序列表中查找“Microsoft Visual Studio 2010 Express”，运行其中的程序“Microsoft Visual C++2010 Express”。

VC++2010 的界面如图 1－7 所示。

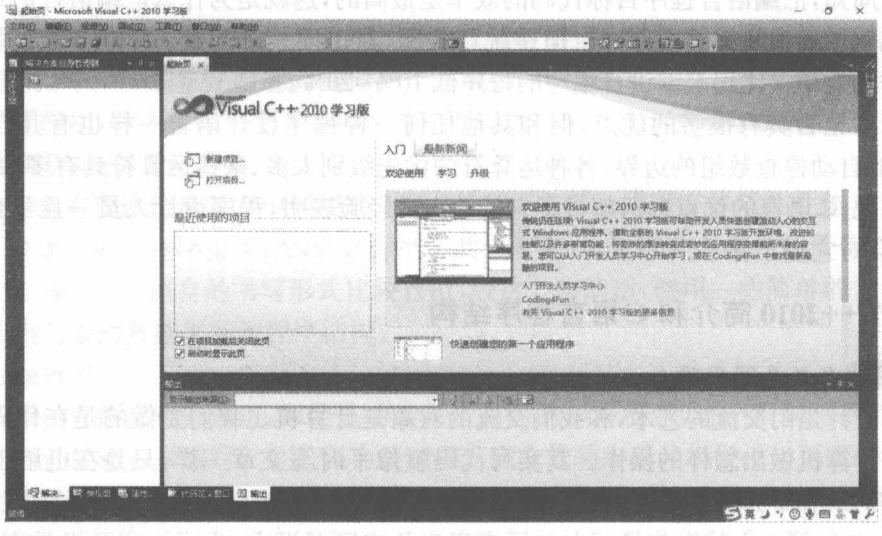

图 1－7　VC++2010 界面组成