

21 世纪高等学校计算机**基础**实用规划教材

基于自然语言思想的递进 C/C++教程 (上册 C)

李春庚 编著



清华大学出版社

21 世纪高等学校计算机**基础**实用规划教材

基于自然语言思想的递进 C/C++教程（上册 C）

李春庚 编著

清华大学出版社
北京

内 容 简 介

本书以自然语言“选词造句,连句成段,多段成章”的渐进学习思想贯穿全书,简捷深入地阐述了 C 语言的概念和语法。将 C 语言中的类型和量值(常量和变量)理解为“名词”;各种运算符理解为“动词”;构成分支结构的 if...else、构成循环结构的 while 等理解为“连词”。名词、动词和连词的有机组合就形成 C 语言的语句。而将数组和结构体等类型和变量理解为集合名词,是基本名词的横向发展,将指针、位、函数和文件理解为名词的纵向发展。

全书例题围绕几个问题展开,要么用不同方法解决,要么不断完善和深化;同时用一个综合实例,融合 C 语言的各个知识点,让 C 语言的学习更接近于实际开发。每章的习题都是例题的变形和深化,以加强学生对知识的理解。

本书适合作为普通高校 C 语言课程的教材,也可用作培训教材及自学教材使用。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

基于自然语言思想的递进 C/C++教程.上册,C/李春庚编著. —北京:清华大学出版社,2009.9
(21 世纪高等学校计算机基础实用规划教材)

ISBN 978-7-302-20391-9

I. 基… II. 李… III. C 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 101095 号

责任编辑:梁 颖

责任校对:时翠兰

责任印制:李红英

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:三河市春园印刷有限公司

经 销:全国新华书店

开 本:185×260 印 张:13.25 字 数:302 千字

版 次:2009 年 9 月第 1 版 印 次:2009 年 9 月第 1 次印刷

印 数:1~4000

定 价:24.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系
调换。联系电话:(010)62770177 转 3103 产品编号:032488-01

编审委员会成员

(按地区排序)

清华大学

周立柱 教授
覃 征 教授
王建民 教授
刘 强 副教授
冯建华 副教授

北京大学

杨冬青 教授
陈 钟 教授
陈立军 副教授

北京航空航天大学

马殿富 教授
吴超英 副教授
姚淑珍 教授

中国人民大学

王 珊 教授
孟小峰 教授
陈 红 教授

北京师范大学

周明全 教授

北京交通大学

阮秋琦 教授

北京信息工程学院

孟庆昌 教授

北京科技大学

杨炳儒 教授

石油大学

陈 明 教授

天津大学

艾德才 教授

复旦大学

吴立德 教授

吴百锋 教授

杨卫东 副教授

同济大学

苗夺谦 教授

徐 安 教授

张惠娟 副教授

华东理工大学

邵志清 教授

华东师范大学

杨宗源 教授

应吉康 教授

上海大学

陆 铭 副教授

东华大学	乐嘉锦	教授
	孙莉	副教授
浙江大学	吴朝晖	教授
	李善平	教授
南京大学	骆斌	教授
	黄强	副教授
南京航空航天大学	黄志球	教授
	秦小麟	教授
南京理工大学	张功萱	教授
南京邮电学院	朱秀昌	教授
苏州大学	龚声蓉	教授
	陈建明	副教授
江苏大学	宋余庆	教授
武汉大学	何炎祥	教授
华中科技大学	刘乐善	教授
中南财经政法大学	刘腾红	教授
华中师范大学	叶俊民	教授
	王林平	副教授
	魏开平	副教授
国防科技大学	赵克佳	教授
	肖依	副教授
中南大学	陈松乔	教授
	刘卫国	教授
湖南大学	林亚平	教授
	邹北骥	教授
西安交通大学	沈钧毅	教授
	齐勇	教授
长安大学	巨永峰	教授
哈尔滨工业大学	郭茂祖	教授
吉林大学	徐一平	教授
	毕强	教授
山东大学	孟祥旭	教授
	郝兴伟	教授
中山大学	潘小轰	教授
厦门大学	冯少荣	教授
云南大学	刘惟一	教授
电子科技大学	刘乃琦	教授
	罗蕾	教授
重庆邮电学院	王国胤	教授
西南交通大学	曾华燊	教授
	杨燕	副教授

出版说明

随着我国改革开放的进一步深化,高等教育也得到了快速发展,各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度,通过教育改革合理调整和配置了教育资源,优化了传统学科专业,积极为地方经济建设输送人才,为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是,高等教育质量还需要进一步提高以适应经济社会发展的需要,不少高校的专业设置和结构不尽合理,教师队伍整体素质亟待提高,人才培养模式、教学内容和方法需要进一步转变,学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007年1月,教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》,计划实施“高等学校本科教学质量与教学改革工程(简称‘质量工程’)”,通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容,进一步深化高等学校教学改革,提高人才培养的能力和水平,更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中,各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势,对其特色专业及特色课程(群)加以规划、整理和总结,更新教学内容、改革课程体系,建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上,经教育部相关教学指导委员会专家的指导和建设,清华大学出版社在多个领域精选各高校的特色课程,分别规划出版系列教材,以配合“质量工程”的实施,满足各高校教学质量和教学改革的需要。

本系列教材立足于计算机公共课程领域,以公共基础课为主、专业基础课为辅,横向满足高校多层次教学的需要。在规划过程中体现了如下一些基本原则和特点。

(1) 面向多层次、多学科专业,强调计算机在各专业中的应用。教材内容坚持基本理论适度,反映各层次对基本理论和原理的需求,同时加强实践和应用环节。

(2) 反映教学需要,促进教学发展。教材要适应多样化的教学需要,正确把握教学内容和课程体系的改革方向,在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生的知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点,保证质量。规划教材把重点放在公共基础课和专业基础课的教材建设上;特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现教学质量和教学改革成果的教材。

(4) 主张一纲多本,合理配套。基础课和专业基础课教材配套,同一门课程可以针对不同层次、面向不同专业的多本具有各自内容特点的教材。处理好教材统一性与多样化,基本教材与辅助教材、教学参考书,文字教材与软件教材的关系,实现教材系列资源配套。

(5) 依靠专家,择优选用。在制定教材规划时依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主题。书稿完成后要认真实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平教材编写梯队才能保证教材的编写质量和建设力度,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校计算机基础实用规划教材

联系人:魏江江 weijj@tup.tsinghua.edu.cn

前言

1. 本书编写背景

C 语言是应用面广、效率高的结构化高级计算机语言；C++ 语言是在 C 语言基础上发展起来的、面向对象的高级计算机语言，它不但继承了 C 语言的所有优点、兼容 C 语言的所有语法，更增加了安全性高、适应性强和编程效率高的特点。通常将 C 语言和 C++ 语言合并，泛称 C++ 语言。C++ 语言也成为全世界多数大学理工类专业计算机编程训练的首选教学语言。

怎样才能通过教和学，高效率地掌握 C++ 语言呢？当前有以下两种教学模式。

(1) 先学习 C 语言，然后学习 C++ 语言。该教、学模式下，知识内容掌握得比较深入，但耗费的时间比较长，效率不高。另外，容易带来“结构化”和“面向对象”两种编程思想的对立。因为在 C 语言的教材中，通常只讲结构化的编程，而 C++ 教材通常强调面向对象的优点和其与结构化思想的不同，忽略了“结构化”和“面向对象”的联系，及“面向对象”是如何从“结构化”发展而来的。

(2) 模糊 C 语言和 C++ 语言的界限，直接学习 C++ 语言。这种教、学模式效率比较高，耗费的时间较短，但教、学效果往往不够理想。因为面向对象编程中，程序语句排列的先后顺序和程序执行的顺序不一致，具有“超文本的跳跃性”，对于没有结构化程序设计基础的初学者理解难度较大。

为了克服以上两种教学模式的不足，我们寻着计算机编程语言产生发展的脉络，提出了“基于自然语言思想的递进 C/C++ 教学”思路。因为计算机编程语言从机器语言到汇编语言再到高级语言的发展过程，就是越来越逼近人类自然语言的过程；从 C 语言到 C++ 语言的发展过程，就如同人类自然语言不断丰富的发展过程，它是一个递进的补充、完善过程。

人类自然语言是由词构成短语，多个词或短语构成句子，多个句子就形成段落，多个段落就是文章。基于人类自然语言的思想，我们将 C/C++ 语言中的类型、常量和变量看做名词，将各种运算符看做动词，将 if...else、switch...case、while 等看做连词，表达式就是短语，程序语句就是句子，程序模块就是段落，整个工程的程序就是文章。而数组、结构体、函数和指针是名词的发展；语句嵌套、函数的递归等是句式的发展；沿着这样的理解思路，C++ 中的类自然是 C 中结构体类型的发展。这样就明晰了从 C 发展到 C++ 的过程，找到了贯穿 C 和 C++ 的一致性知识框架体系，使 C 语言和 C++ 语言无缝衔接，使学生在 学习过程中，不去区分 C 和 C++，更多地学习 C 和 C++ 的递进性、一致性，而在课程结束

后,可以深入地理解 C 和 C++ 的不同,领略 C++ 对 C 的补充和增强,深入理解 C 语言和 C++ 语言的知识内容,真正做到融会贯通,培养编程实践能力。

本书讲稿已经在大连海事大学的教学中使用,得到良好的教学效果。

2. 本书特色

(1) 以自然语言的思想,讲授计算机语言,形象易懂。

(2) 篇幅短小,中心突出,练习题通常是例题的发展和变化,少而精,非常适合于课堂教学和初学者自学。

(3) 仅用几个例题,贯穿本书始末,将学习的各种语法递进加入例题,使之不断完善,丰富,一步步接近实际开发。

(4) 引入四则运算数学方法,讲解二维数组指针难点,使语言的学习成为简单的推导,深入而准确。

(5) 本书提出表达式、函数是复合的变量,常量是变量的“瞬间快照”等观点,蕴涵了“泛化”的思想,潜移默化训练学生的思维方式,为 C++ 的泛型程序设计打下思想基础。

感谢大连海事大学信息科学技术学院计算机系“C/C++ 语言课程组”的史一民老师、刘宁老师、刘亚清老师和于纯妍老师给予的大力支持,感谢遥感信息研究所的安居白教授给予的大量帮助。

感谢读者选择使用本书,欢迎您对本书内容提出批评和修改意见,我们将不胜感激。
联系方式: licgsh@163.com, liangying@tup.tsinghua.edu.cn。

李春庚

2009 年 5 月于大连海事大学

目 录

第 1 章 概述	1
1.1 自然语言和计算机语言	1
1.2 C 语言及 C++ 语言	2
1.3 简单的 C/C++ 程序及其运行方法(环境的使用)	2
1.3.1 C/C++ 程序开发运行环境	2
1.3.2 格式化输出输入函数 printf() 和 scanf()	5
习题	8
第 2 章 基本的 C/C++ 语言	9
2.1 C/C++ 语言中的名词——类型、量值(常量和变量)	9
2.1.1 整型和整型量值	9
2.1.2 浮点型和浮点量值(常量和变量)	11
2.1.3 字符型和字符量值(常量和变量)	12
2.1.4 字符串常量	13
2.2 C/C++ 语言中的动词——运算符、短语—表达式和特殊动词性关键字	14
2.2.1 赋值运算符和赋值表达式	14
2.2.2 算术运算符和算术表达式	15
2.2.3 逻辑运算符和逻辑表达式	17
2.2.4 关系运算符和关系表达式	18
2.2.5 其他运算符和表达式	19
2.2.6 不同类型量值的不同运算的混合	21
2.3 C/C++ 语言中的连词及句子——分支、循环和顺序结构	24
2.3.1 顺序结构	25
2.3.2 if...else——分支结构	25
2.3.3 switch...case——分支结构	27
2.3.4 ?...:——分支结构	29
2.3.5 for——循环结构	29
2.3.6 while——循环结构	31

2.3.7	do...while——循环结构	31
2.3.8	goto...label...if——循环结构	32
2.3.9	break——终止循环或开关	34
2.3.10	continue——终止本次循环	35
2.4	C/C++ 语言中的句子小结	35
2.5	用 C/C++ 语言编写程序——段落	36
2.6	用 C/C++ 语言编写复杂段落——语句嵌套	36
2.6.1	复杂段落——语句嵌套的含义	36
2.6.2	嵌套语句的示例	37
2.6.3	复杂段落——语句嵌套的小结	40
2.7	总结	42
	习题	43
第 3 章 C/C++ 语言的横向名词性扩充		44
3.1	C/C++ 语言的横向名词性扩充(1): 同类变量的集合——数组	44
3.1.1	一维数组	44
3.1.2	二维数组	48
3.1.3	n 维数组	52
3.1.4	字符数组	52
3.2	C/C++ 语言的横向名词性扩充(2): 自定义类型——typedef	58
3.2.1	简单类型的自定义	58
3.2.2	同类集合类型的自定义——变量数组类型的定义	58
3.3	C/C++ 语言的横向名词性扩充(3): 异类集合类型和异类集合变量—— 结构体型(值)、共用体型(值)、枚举型(值)	59
3.3.1	结构体类型和结构体变量	59
3.3.2	共用体类型和共用体变量	62
3.3.3	枚举类型和枚举变量	62
3.4	C/C++ 语言的横向名词性扩充(4): 异类集合类型的同类集合类型—— 结构体数组	65
3.5	各种横向扩充类型的融合使用	68
	习题	71
第 4 章 C/C++ 语言的纵向名词性扩充(1): 指针		72
4.1	简单指针类型和指针量值(常量和变量)	72
4.1.1	变量的三层含义	72
4.1.2	变量的指针和指针类型的变量	73
4.1.3	与指针相关的动词补充——& 和 * 运算符	73

4.1.4	指针和指针变量示例	74
4.2	同类集合变量——数组的指针和对应的指针变量	76
4.2.1	一维数组的指针和操作一维数组的指针变量——本质上是指向 单个变量的指针变量	76
4.2.2	二维数组的各级指针	80
4.2.3	操作二维数组的各种指针变量——本质上是指向简单变量和一维 数组的指针变量	85
4.2.4	m 维数组的指针和指向 m 维数组的指针变量	89
4.3	指向指针变量的指针量值——二级指针	89
习题		97
第 5 章	C/C++ 语言的纵向名词性扩充(2): 函数	98
5.1	函数的构成	98
5.1.1	函数的定义	98
5.1.2	函数的声明和调用	99
5.1.3	函数的参数和返回值	99
5.2	不同参数和不同返回值形式的各种函数	100
5.2.1	无返回值函数	100
5.2.2	参数或返回值是异类集合量值的函数	101
5.2.3	参数是简单指针量值的函数	102
5.2.4	异类集合类型的指针量值作为函数的参数或返回值	105
5.2.5	函数的参数是和一维数组名对应的指针量值	108
5.2.6	函数的参数是和二维数组名对应的指针量值	109
5.2.7	返回值是指针量值的函数	111
5.3	函数的多级调用	114
5.3.1	函数的嵌套调用	114
5.3.2	函数的递归调用	116
5.4	函数的指针和指向函数的指针变量	117
5.4.1	函数的指针	117
5.4.2	指向函数的指针变量	118
5.5	main 函数的参数	121
5.6	动态内存管理函数	122
习题		124
第 6 章	名词的空间域、时间域和物理存储位置	126
6.1	C/C++ 程序的逻辑结构	126
6.2	名词性的类型和量值在各逻辑结构上的空间域	127

6.2.1	模块域.....	127
6.2.2	嵌套模块的层域.....	128
6.2.3	文件域.....	130
6.2.4	工程域.....	132
6.2.5	工作区域.....	134
6.3	名词性的类型和量值在各逻辑结构上的时间域	134
6.3.1	模块内 static	134
6.3.2	文件内的 static	135
6.3.3	关于 static 的总结	136
6.4	变量的物理存储位置	136
	习题.....	137
第 7 章 编译预处理.....		138
7.1	简单替换宏——符号常量	138
7.2	复杂替换宏——宏函数	140
7.3	批量的嵌入——文件包含	142
7.4	条件编译	143
7.4.1	#ifndef... #else... #endif	143
7.4.2	#ifdef... #else... #endif	148
7.4.3	#if... #else... #endif	149
7.4.4	assert 故障诊断	149
7.4.5	字符串替换——#	150
7.4.6	数字链接——##	150
	习题.....	151
第 8 章 C/C++语言的纵向名词性扩充(3): 宏观扩充——文件		152
8.1	输入输出流和文件的指针	152
8.1.1	输入输出流.....	152
8.1.2	文件的指针.....	153
8.2	文件的打开和关闭	153
8.2.1	文件的物理形式.....	153
8.2.2	文件的打开.....	153
8.2.3	文件的关闭.....	155
8.3	文件的顺序读写	155
8.3.1	向文件中顺序输出(写)字符.....	155
8.3.2	从文件中顺序输入(读)字符.....	156
8.3.3	顺序多个数据读写.....	158

8.3.4	文件的格式化输入和输出	161
8.3.5	文件的字符串读写	161
8.4	文件内的定位和随机读写	163
8.4.1	检测文件位置指针	163
8.4.2	置文件位置指针于文件头	163
8.4.3	置文件位置指针于指定位置	163
8.5	文件的出错控制	166
8.5.1	检查是否到达文件尾 <code>feof()</code>	167
8.5.2	检查当前文件操作是否出错 <code>ferror()</code>	167
8.5.3	清除文件错误标识 <code>clearerr()</code>	167
	习题	167
第9章	C/C++语言的纵向名词性扩充(4): 微观扩充——位和位域	168
9.1	位运算	168
9.1.1	位运算符和位运算	168
9.1.2	位运算举例	170
9.2	位域	171
	习题	173
附录	习题答案	174
	参考文献	195

1.1 自然语言和计算机语言

自然语言有着悠久的历史,它的一切都是在人类漫长的历史中自然发展、约定俗成的。它的构成单位由小到大可以粗略地分为字、词、短语、句子、段落和篇章等,在英文中没有字的概念,最小的单位是词(如不需要,可统称为字词)。人学习语言的过程是先学习简单的字词,通常是名词、动词和连词,如车、雨、乘、如果……那么……等,这些字词已经在漫长的历史中被固定,无法改变,如果非要改变,那么就无法和别人交流沟通。对简单的字词有了一定的掌握后,两个以上的字词就构成短语,如出租车、下雨等;短语有一定自由度,可以适当变化。在短语的基础上,配合连词可以说出“如果下雨,那么就乘出租车”的句子,句子的灵活性就更大了。当然,如果一个人掌握了很多的句子,就可以将句子加以组合,形成段落和篇章。随着社会的发展和社会实践的需要,人们是可以不断创造出新的字词,使语言不断丰富。这是我们每个人学习自然语言的过程。但是必须明确,熟练掌握一种自然语言,并不是说就可以写出优美的文章。就好像可以熟练地使用汉语,而又有多少人可以用汉语写出优美的文章呢!因为文章中不但要有正确的字词、短语和句子,更要有深邃的思想,而思想不属于语言本身,只是以语言作为表达的载体。思想属于某个专门的领域,需要对该领域知识的深刻理解和掌握。这就是说,语言只是工作于社会中的基本技能,在此基础上,需要对某一领域的掌握和理解,然后更要将语言恰当地应用于该领域中,用语言表达该领域的思想,解决该领域的问题。另外,语言是非常依赖实践的,只是听别人说,而自己不说,只是阅读别人写的文章,自己不写,是永远不可能掌握语言的。

计算机语言的构成及学习过程和人类的自然语言是非常相似的,也包含“名词”,在计算机语言中通常叫“类型”和“量值”,“量值”包括“常量”和“变量”;“动词”,在计算机语言中叫“运算符”;“连词”,在计算机语言中叫控制流程;构成的“短语”在计算机语言中叫表达式;用表达式构成的“句子”在计算机语言中叫语句;相当于自然语言的“段落”叫程序段或模块;和“篇章”相对应的就是程序了。计算机语言发展到今天,已经有了和“著作”相对应的单位,那就是“工程”。学习计算机编程语言的过程与学习自然语言的过程是一致的。先要学会类型的定义,量值的声明,表达式的创建,程序语句的书写,以及模块和程序的编制。完全类似的是,并不是掌握了计算机语言就成为了编程高手。计算机的程序中同样必须蕴涵着思想,用计算机的术语叫“算法”,它不属于计算机语言本身,它属于要解决专业的领域。只有熟练掌握了计算机语言,又真正理解了所要解决的专业问题,才能编写出可以解决问题的、有一定规模的计算机程序。当然,对于很多类似的问题有通用的算法,这

是算法课要学习的内容。程序设计课讲授一种计算机语言的基本格式和语法规则,如何用该种计算机语言和计算机进行交流,该课程通常不涉及复杂的算法,只是讲授语言的规则和思路。如果和汉语语文课相类比的话,程序设计课相当于小学到初中的语文,强调的是字词的正确,语句的通顺,意思简单,重点不放在主题的确立和文章的布局谋篇。

1.2 C 语言及 C++ 语言

计算机语言有好多种,各有特点,适合不同的应用需要。C 和 C++ 语言蕴涵了所有计算机语言的思想和方法,所以,全世界的大学里在做计算机语言的基础训练时,都是用 C 或 C++ 语言。C 语言是结构化程序设计思想的代表; C++ 语言是面向对象程序设计思想的代表。面向对象程序设计思想是建立在结构化程序设计思想基础上的, C 语言是 C++ 语言的子集,所有 C 语言的关键字、表达式、语句、函数等在 C++ 语言环境中都能够被正确识别。从 C 语言开始,像学习自然语言那样,通过递进地学习,可以很自然地过渡到 C++ 语言。熟练掌握 C 和 C++ 语言后,理论上可以很容易掌握其他各种计算机语言。本书的后面章节中,通常说“C/C++ 语言”,而不强调 C 语言,因为 C 语言是 C++ 语言的一部分。

1.3 简单的 C/C++ 程序及其运行方法(环境的使用)

1.3.1 C/C++ 程序开发运行环境

怎样才能使用计算机编写一段 C/C++ 语言的程序并使之运行起来,输出结果呢?

(1) 要安装一套用来开发 C/C++ 程序的软件。用来开发 C/C++ 程序的软件很多,通常叫做程序开发环境,如 Microsoft Visual C++, Turbo C/C++, Bland C/C++ 等。本书以 Microsoft Visual C++ 6.0 为例来讲解,它是当前使用广泛、功能强大、操作方便的微型计算机 Windows 平台 C/C++ 开发环境。

(2) 在“我的电脑”或“资源管理器”中创建工作文件夹,以后编写的程序和结果都存储在此处,这是管理计算机文件的好习惯,可以有效避免找不到自己需要的文件。

(3) 运行 Microsoft Visual C++。选择“文件”→“新建”命令,打开“新建”对话框。在“工程”选项卡中选定工程类型为 Win32 Console Application,在“工程”文本框中输入一个和实际问题相关的工程名,在“位置”文本框中指定自己创建的工作文件夹,选择“创建新工作区”单选按钮,如图 1.1 所示。单击“确定”按钮,在打开的对话框中选择 An empty project 单选按钮,如图 1.2 所示。单击“完成”按钮,并在打开的“新建”对话框“工程”选项卡中单击“确定”按钮,完成新工程的创建。

(4) 再选择“文件”→“新建”命令,打开“新建”对话框,从“文件”选项卡中选定文件类型为 C++ Source File,在“文件”文本框中输入一个和问题相关的文件名,如图 1.3 所示。单击“确定”按钮完成源文件的创建,如图 1.4 所示。

注意: 文件名不一定要和工程名相同,但仅有一个源文件的小的工程,文件名通常和工程名相同。

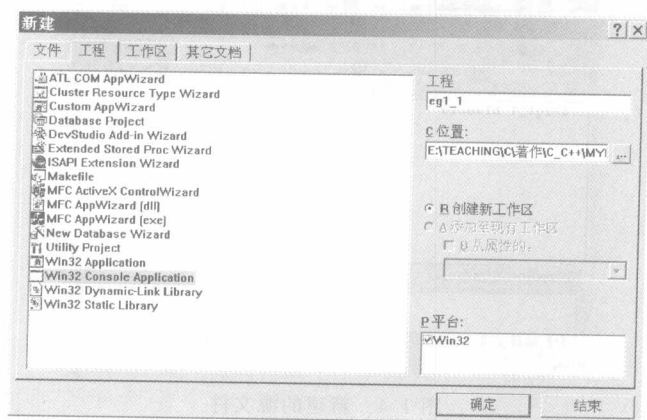


图 1.1 “新建”对话框“工程”选项卡

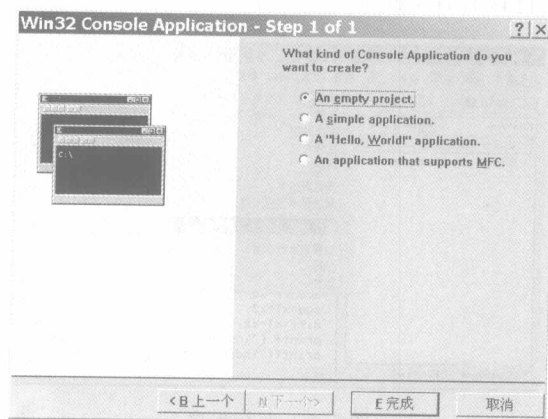


图 1.2 创建一个空工程

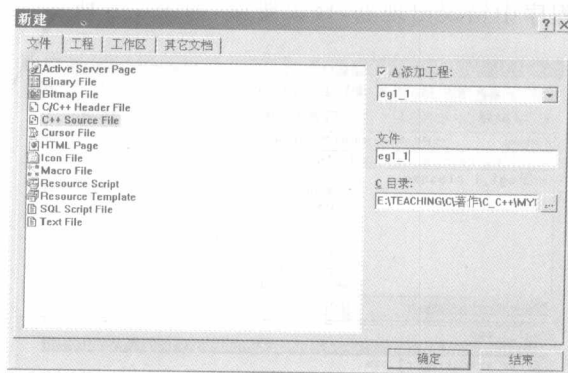


图 1.3 “新建”对话框“文件”选项卡