



普通高等教育“十二五”规划教材
PUTONG GAODENG JIAOYU“SHIERWU”GUIHUAJIAOCAI

C语言程序设计

实训及考试指导

C YUYAN CHENGXU
SHEJI SHIXUN JI KAOSHI
ZHIDAO

主编 唐国纯



普通高等教育“十二五”规划教材
PUTONG GAODENG JIAOYU “SHIERWU” GUIHUA JIAOCAI

C语言程序设计 实训及考试指导

C YUYAN CHENGXU
SHEJI SHIXUN JI KAOSH
ZHIDAO

主 编 唐国纯
副主编 别凤兰

内 容 简 介

本书可作为《C 语言程序设计教程》配套的学习、实训和考试指导。本书课程体系和课程内容立足当前学生的需求和职业能力培养,分为四个部分:C 语言程序设计实践指导、全国计算机等级考试二级 C 语言指导、蓝桥杯软件大赛 C 语言组辅导和 C 语言应用扩展。本书适合作为理工类专业学生学习 C 语言程序设计课程的辅助教材,同时也适合参加全国计算机等级考试和全国 C 语言组的蓝桥杯软件大赛的考生使用,也可作为自学者学习 C 语言程序设计的辅导书。

图书在版编目(CIP)数据

C 语言程序设计实训及考试指导 / 唐国纯 主编. —北京:中国传媒大学出版社, 2013. 7

ISBN 978 - 7 - 5657 - 0745 - 2

I. ①C… II. ①唐… III. ①C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 152239 号

C 语言程序设计实训及考试指导

作 者: 唐国纯

责任编辑: 李 莉 穆会荣

责任印制: 曹 辉

封面设计: 雨 & 寒

出 版 人: 蔡 翔

出版发行: 中国传媒大学出版社

社 址: 北京市朝阳区定福庄东街 1 号 邮编: 100024

电 话: 65450532 或 65450528 传真: 010 - 65779405

网 址: <http://www.cucp.com.cn>

经 销: 全国新华书店

印 刷: 北京市科星印刷有限责任公司

开 本: 787 × 1092 毫米 1/16

印 张: 19.5

字 数: 487 千字

版 次: 2013 年 8 月第 1 版 2013 年 8 月第 1 次印刷

ISBN 978 - 7 - 5657 - 0745 - 2/TP · 0745

定价: 37.00 元

前 言

C 语言是一种编程灵活、特色鲜明的程序设计语言，是电子信息专业及部分理工专业类学生的基础课程。学好这种语言可以为后续的相关课程打好坚实的基础。C 语言除了学习必须的基本知识，如概念、方法和语法规则之外，更重要的是进行实训，以提高学习者的动手和编程能力，培养学生的自主学习能力，养成自觉学习的意识，这才是学习 C 语言的最终目的。结合多年来的教学、等级考试和大赛辅导经验，根据学生的学习情况，为配合教学过程，培养学生动手能力和职业能力，特编写了该《C 语言程序设计实训及考试指导》一书，以在实训、二级考试和全国蓝桥杯 C 语言组软件大赛过程中给学生提供帮助。

本书课程体系和课程内容立足当前学生的需求和职业能力培养，分为四个部分：C 语言程序设计实践指导、全国计算机等级考试二级 C 语言指导、蓝桥杯软件大赛 C 语言组辅导和 C 语言应用扩展。第一部分 C 语言程序设计实践指导分为 15 个章节实训和 5 个综合实训两部分，目的是为了让学生编写规范的代码，采用随堂实训，使学生掌握该部分内容的知识要点，并能灵活运用在实训中，培养学生的动手能力和职业编程能力。为提高学生的学习积极性，同时为培养软件开发的合作精神，综合实训可以使学生从软件开发的角角度思考问题、解决问题。形式采用小组合作，各小组设定项目组长和成员，一般为 5 人一组，由项目组长组织人员、进行分工和协调工作，负责开发过程中遇到的疑难问题的解决，并进行开发的全程控制。第二部分为全国计算机等级考试二级 C 语言指导，通过该部分的学习，学生可顺利通过二级 C 语言考试并获取证书。第三部分为蓝桥杯软件大赛 C 语言组辅导，该部分相对于前两部分来说要求更高，可使优秀学生挑战自己，脱颖而出，成为企业争夺的优秀学子。第四部分为 C 语言应用扩展，虽然简短，但指明了 C 语言的应用领域，进一步激发学生的学习积极性，通过实训，让学生初步掌握 Linux 下的 C 语言编程过程。

本书由唐国纯担任主编，由别凤兰担任副主编。本书体系新颖，具有基础性、实用性和系统性等特点，可以指导学生按照由浅入深、从低到高的训练过程，逐步提高他们的编程动手能力和实践能力。本书在编写的过程中也参考了许多文献及成果，在此，对本书参考书籍的作者、因特网上信息的提供者及作者，表示深深的敬意及诚挚的感谢。

由于编者水平有限，加之时间仓促，错漏之处在所难免，敬请广大读者、专家批评指正。

编 者

2013 年 5 月

目 录

第一部分 C语言程序设计实践指导	1
第1章 C语言概述 / 1	
实训一 Visual C++ 中的C语言的运行环境和运行过程 / 1	
实训二 TC 中的C语言的运行环境和运行过程 / 8	
第2章 简单C语言程序——顺序结构 / 11	
实训三 简单的C程序设计和基本数据处理 / 11	
第3章 分支结构程序设计 / 15	
实训四 选择结构程序设计 / 15	
第4章 循环结构程序设计 / 20	
实训五 循环结构程序设计（一） / 20	
实训六 循环结构程序设计（二） / 24	
第5章 数组 / 27	
实训七 一维数组的应用 / 27	
实训八 二维数组的应用 / 32	
实训九 字符数组的应用 / 34	
第6章 模块化程序设计 / 37	
实训十 函数参数的传递 / 37	
实训十一 嵌套与递归调用的实现 / 44	
第7章 编译预处理 / 46	
实训十二 存储类型和编译预处理 / 46	
第8章 指针 / 50	
实训十三 指针的应用 / 50	
第9章 构造数据类型 / 58	
实训十四 结构体和共用体 / 58	
第10章 文件 / 65	
实训十五 文件的应用 / 65	
综合实践训练 / 76	
实训十六 C语言 socket 实现主机与客户端异步通信 / 76	

实训十七	汇款系统	/ 82
实训十八	图书管理系统	/ 86
实训十九	歌曲信息管理系统	/ 98
实训二十	通讯录管理系统	/ 109

第二部分 全国计算机等级考试二级C语言指导 118

全国计算机等级考试二级 C 语言考试大纲	/ 118
二级 C 语言公共基础知识	/ 120
二级 C 语言公共基础知识综合练习	/ 142
二级 C 语言考试重点介绍	/ 150
真题训练	/ 154
理论习题	/ 209

第三部分 蓝桥杯软件大赛C语言辅导 236

第三届“蓝桥杯”软件大赛——决赛规则	/ 236
真题训练	/ 240
理论习题	/ 257
编程题	/ 266

第四部分 C语言应用扩展 298

一、C 语言的应用领域	/ 298
二、实训：Linux 下 C 语言使用、编译与调试实验	/ 298

参考文献 303

第一部分 C 语言程序设计实践指导

在学习 C 语言程序设计的过程中，上机实训是十分重要的环节，通过实训，可以加深对 C 语言功能特征、语法规则、程序编译与运行等基本概念和基本方法的理解和运用。通过上机调试程序，使学生能及时发现程序编制中出现的错误并找到改正方法，提高学生的独立编程能力和编程技巧，为 C 语言在后续课程中的应用打下坚实的基础。

第 1 章 C 语言概述

实训一 Visual C++ 中的 C 语言的运行环境和运行过程

实训目的：

- (1) 掌握 C 程序设计编程环境 Visual C++，掌握运行一个 C 程序设计的基本步骤，包括编辑、编译、连接和运行。
- (2) 掌握 C 语言程序设计的基本框架，能够编写简单的 C 语言程序。
- (3) 了解程序调试的思想，能找出并改正 C 语言程序中的语法错误。

实训内容：

- (1) 在“我的电脑”上新建一个文件夹，用于存放 C 语言程序。
- (2) 调试示例，在屏幕上显示一个短句“Hello World!”。

源程序

```
#include <stdio.h>
void main()
{
    printf("Hello World!\n");
}
```

运行结果：

Hello World!

基本步骤：（要求熟练掌握）

① 启动 VC ++。“开始”→“程序”→“Microsoft Visual Studio 6.0”→“Microsoft Visual C++6.0”进入 VC ++ 编程环境，如图 1-1 所示。

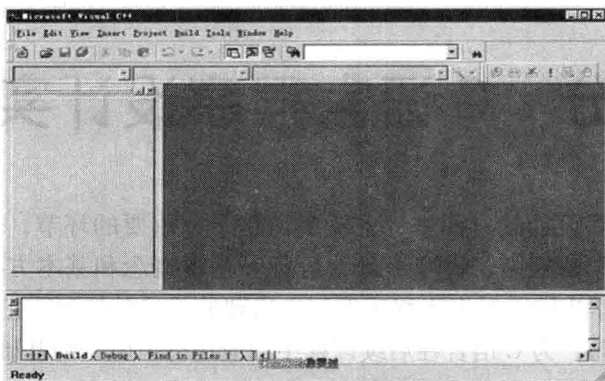


图 1-1 VC6 的集成开发环境窗口

图 1-1 式样的窗口从大体上可分为四部分。上部：菜单和工具条；中左：工作区（workspace）视图显示窗口，这里将显示处理过程中与项目相关的各种文件种类等的信息；中右：文档内容区，是显示和编辑程序文件的操作区；下部：输出（Output）窗口区，程序调试过程中，进行编译、链接、运行时输出的相关信息将在此处显示。

【注意】 由于系统的初始设置或者环境的某些不同，可能导致所启动的 VC6 初始窗口式样与图中有所不同，也许会出现没有 Workspace 窗口或 Output 窗口，这时可通过“View→Workspace”菜单选项的执行，使中左处的工作区窗口显现出来；而通过“View→Output”菜单选项的执行，又可使下部的输出区窗口得以显现。当然，如果不想看到这两个窗口，可以点击相应窗口的“×”按钮来关闭窗口。

② 创建工程并输入源程序代码。为了把程序代码输入给计算机，需要使用 VC6 的编辑器来完成。用 VC6 编写并处理的任何程序都与工程有关（都要创建一个与其相关的工程），而每一个工程又总与一个工程工作区相关联。工作区是对工程概念的扩展。一个工程的目标是生成一个应用程序，但很多大型软件往往需要同时开发数个应用程序，VC 开发环境允许用户在一个工作区内添加数个工程，其中有一个是活动的（缺省的），每个工程都可以独立进行编译、连接和调试。选择菜单 File 下的 New 项，会出现一个如图 1-2 所示的选择界面。

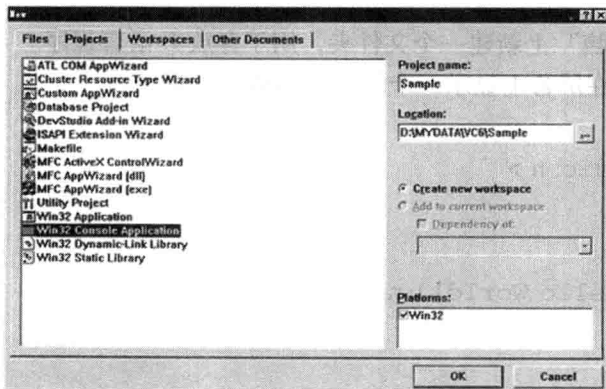


图 1-2 新建一个名为 Sample 的工程（同时自动创建一工作区）

在属性页中选择 Projects 标签后, 会看到近 20 种工程类型, 我们只需选择其中最简单的一种: “Win32 Console Application”, 而后往右上处的 “Location” 文本框和 “Project name” 文本框中填入工程相关信息所存放的磁盘位置 (目录或文件夹位置) 以及工程的名字, 设置到此时的界面信息如图 1-2 所示。在 “Location” 文本框中填入如 “D: \myData \VC6”, 这是假设你准备在 D 磁盘的 \ myData \VC6 文件夹即子目录下存放与工程工作区相关的所有文件及其相关信息, 当然也可通过点击其右部的 “...” 按钮去选择并指定其他文件夹即子目录位置。“Project name” 文本框中填入如 “Sample” 的工程名 (注意, 名字根据工程性质确定, 此时 VC6 会自动在其下的 Location 文本框中用该工程名 “Sample” 为用户建立一个同名字目录, 随后的工程文件以及其他相关文件都将存放在这个目录下)。

③ 选择 OK 按钮进入下一个选择界面。这个界面主要是询问用户想要构成的工程类型, 此时的界面如图 1-3 所示。

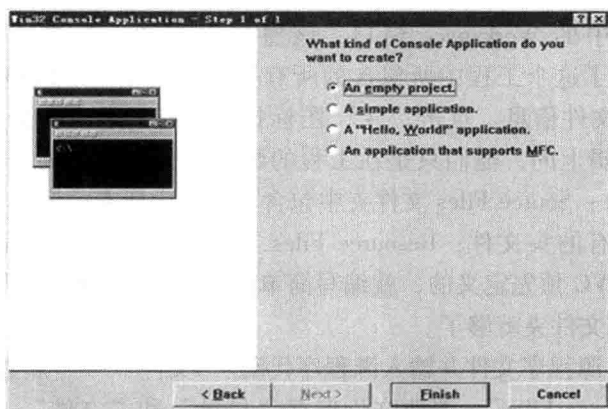


图 1-3 选择创建工程类型

【说明】 选择 “An empty project” 项将生成一个空的工程, 工程内不包括任何东西。选择 “A simple application” 项将生成包含一个空的 main 函数和一个空的头文件的工程。选 “A” Hello World! “application” 项与选 “A simple application” 项没有什么本质的区别, 只是需要包含有显示出 “Hello World!” 字符串的输出语句。选择 “An application that supports MFC” 项的话, 可以利用 VC6 所提供的类库来进行编程。

学习 C 语言, 在这里我们一般选择 “An empty project” 项。单击 Finish 按钮, 这时 VC6 会为你生成一个小型报告, 报告的内容是刚才所有选择项的总结, 并且询问你是否接受这些设置。如果接受选择 OK 按钮, 否则选择 Cancel 按钮。我们选 OK 从而可进入到真正的编程环境下。此时的界面情况如图 1-4 所示。

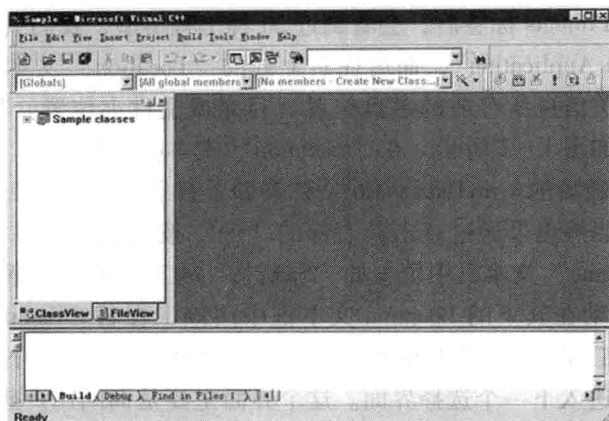


图 1-4 创建工程 Sample 后的 VC6 集成开发环境窗口

【注意】 屏幕中的 Workspace 窗口，该窗口中有两个标签 ClassView 和 FileView。ClassView 标签中列出了这个工程中所包含的所有类的有关信息。FileView 标签中列出了这个工程所包含的所有文件信息。点击“+”图标打开所有的层次，会发现有三个逻辑文件夹（逻辑文件夹是逻辑上的，他们只是在工程的配置文件中定义的，在磁盘上并没有物理量存在这三个文件夹）：Source Files 文件夹中包含了工程中所有的源文件；Header Files 文件夹中包含了工程中所有的头文件；Resource Files 文件夹中包含了工程中所有的资源文件。这三个逻辑文件夹是 VC 预先定义的，就编写简单的单一源文件的 C 程序而言，我们只需要使用 Source Files 一个文件夹就够了。

④ 工程中新建 C 源程序文件并输入源程序代码。“文件”→“新建”，单击“文件”选项卡，选择“C ++ Source Files”，修改文件保存“目录”和“文件”（文件名），单击“确定”，或者选择菜单 Project 中子菜单 Add To Project 下的 new 项，在出现的对话框的 Files 标签（选项卡）中，选择“C ++ Source File”项，然后在右方的“File”一栏中为程序起个名字，可以是任意字母和数字的组合，但其中不能有空格，并且最后一定要以“.c”结尾，否则系统将按 C ++ 扩展名“.cpp”保存。我们取名为 Hello（其他遵照系统隐含设置，此时系统将使用 Hello.cpp 的文件来保存所键入的源程序），此时的界面情况如图 1-5 所示。

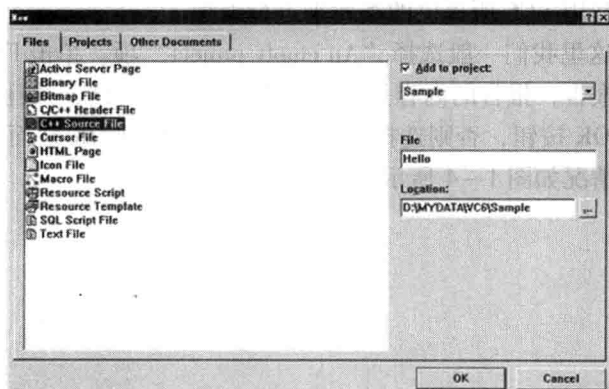


图 1-5 在工程 Sample 中新建一名为 Hello.cpp 的 C 源程序文件

选择 OK 按钮，进入输入源程序的编辑窗口。

通过键盘输入你所需要的源程序代码：（注意：源程序一定要在英文状态下输入，即字符标点都要在半角状态下，同时注意大小写，一般都用小写）

```
#include <stdio.h>
void main()
{
    printf("Hello World!\n");
}
```

这时可通过 Workspace 窗口中的 FileView 标签，看到 Source Files 文件夹下文件 Hello.cpp 已经被加了进去，此时的界面情况如图 1-6 所示。

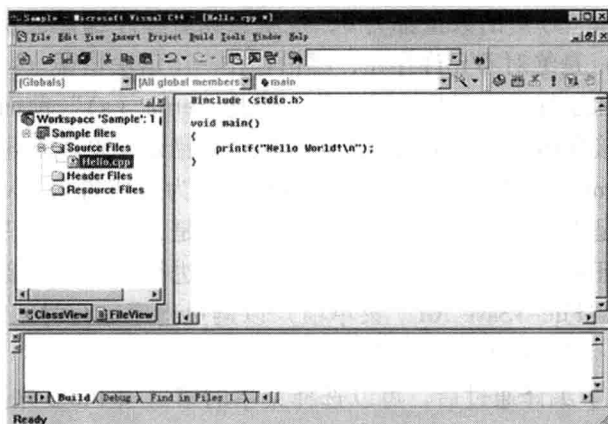


图 1-6 在 Hello.cpp 编辑 C 源程序代码

同时在 Workspace 窗口的 ClassView 标签中的 Globals 文件夹下，也可以看到我们刚才所键入的 main 函数。

【说明】 不创建工程，也可直接编辑 C 语言程序源代码。选“Files”标签，再选择“C++ Source File”，其界面与图 1-5 相似（仅 Add to project 是暗淡的、无法选择），后续操作则与前述相同。最简单的做法是：直接使用工具栏上的新建文件按钮“”新建一空白文件，紧接着单击工具栏上的保存按钮“”保存此空文件。同时要注意，当输入结束后，保存文件时，可指定扩展名“.c”，否则系统将按 C++ 扩展名“.cpp”保存。这种方式新建的 C 源程序文件在编译时，会提示用户，要求允许系统为其创新一个默认的工程（含相应的工作区）。

⑤ 编译 (*.obj)，检查语法错误。C 语言源程序文件是不能直接被计算机所执行的，它需要通过编译（Compile）和连接（Link）两个步骤，才能生成可被计算机直接执行的“可执行文件”。整个过程如图 1-7 所示，请牢记这张图。

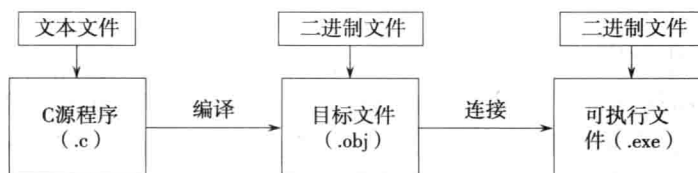


图 1-7 C 程序的编译连接过程

C 语言源程序文件是文本文件，它的后缀名为“.c”，它不能被计算机直接执行；经过编译后，生成目标文件，它是二进制文件，后缀名为“.obj”，它也不能被计算机直接执行；再经过连接（有的书籍称为链接），生成可执行文件，它是二进制文件，后缀名为“.exe”，它可以被计算机直接执行。选择执行菜单第一项 Compile，此时将对程序进行编译。若编译中发现错误（error）或警告（warning），将在 Output 窗口中显示出它们所在的行以及具体的出错或警告信息，可以通过这些信息的提示来纠正程序中的错误或警告（注意，错误是必须纠正的，否则无法进行下一步的链接；而警告则不然，它并不影响进行下一步）。当没有错误与警告出现时，Output 窗口所显示的最后一行应该是：“Hello. obj - 0 error (s), 0 warning (s)”。其中，“error (s)”代表语法错误，“warning (s)”代表警告。语法错误（error (s)）与警告（warning (s)）的数量都为 0，也就意味着程序编译成功。在实际编程当中，语法错误（error (s)）是绝对不允许存在，哪怕存在一个语法错误（error (s)），程序都无法进入下一步的连接和运行，也就是编译失败。因此“error (s)”前面的错误必须是“0”，才能进入下一步的连接和运行，否则就要改正程序中的错误，并重新进行编译，直到编译成功。而对于警告（warning (s)），则可以不必要理会，因为造成警告（warning (s)）的原因比较复杂，并不一定是因为语法错误。需要特别指出的是，编译成功只能保证程序在语法上是正确的，并不能保证程序是无错的。注意，在对程序进行编译、链接和运行前，最好先保存自己的工程（使用“File→Save All”菜单项）以避免程序运行时系统发生意外而使自己之前的工作付诸东流，应让这种做法成为自己的习惯与素质。

⑥连接（*.exe）。编译通过后，可以选择菜单的第二项 Build 来进行连接生成可执行程序。在连接中出现的错误也将显示到 Output 窗口中。连接成功后，Output 窗口所显示的最后一行应该是：“Sample. exe - 0 error (s), 0 warning (s)”。

⑦运行。最后就可以运行（执行）我们所编制的程序了，选择 Execute 项（该选项前有一个深色的感叹号标志“!”，实际上也可通过单击窗口上部工具栏中的深色感叹号标志“!”来启动执行该选项），VC6 将运行已经编好的程序，执行后将出现一个结果界面，使得用户可以浏览输出结果，按下任一个键盘按键时可返回到集成界面的编辑窗口。

⑧关闭程序工作区。执行“File→Close Workspace”菜单项，待系统询问是否关闭所有的相关窗口时，回答“是”，则结束了一个程序从输入到执行的全过程，回到了刚刚启动 VC6 的那个初始画面。

⑨打开文件。“File→Open”查看 C 源文件、目标文件和可执行文件的存放位置。

源文件在保存目录下，目标文件和可执行文件在“保存目录\Debug”中。

【说明】 创建工程工作区之后，系统将创建出一个相应的工作区文件（.dsw），用来存放与该工作区相关的信息；另外还将创建出的其他几个相关文件：工程文件（.dsp）以及选择信息文件（.opt）等。

⑩及时备份。

- 完全备份。对于刚才工作的工程 Sample 而言，只需将 D:\myData\VC6 下的文件夹 Sample 复制到 U 盘或打包成一个文件后放到自己的邮箱。需要在其他计算机上继续完成该工程时，将该文件夹复制到该计算机的硬盘上，进入 VC6，通过“File→Open Workspace”菜单项将该工程打开即可。

- 只备份 C 源程序文件。对于刚才工作的工程 Sample 而言，工程非常简单，没有什么

专门的设置，因此，仅备份其中的 C 源程序 Hello.cpp 就足矣。需要在其他计算机上继续完成该程序时，只需将备份的程序复制到该计算机的硬盘上，进入 VC6，根据前面的讲述，新建 Win32 Console Application，然后通过“Project→Add to Project→Files”菜单项将 Hello.cpp 添加到新建的工程中。

最简单的做法是：直接使用工具栏上的文件打开按钮“”打开 Hello.cpp。

⑪正式发布。前面所说的编译、连接过程都是 Debug 类型的，也就是说，当 VC6 在进行这些工作时将加入一些调试信息，致使编译连接后生成的代码很庞大，效率也降低。如果确信程序已经完美无缺或者是要正式发布，就应该选择菜单 Build 中的 Batch Build 项，产生如图 1-7 所示的对话框，其中的两个选项分别代表编译的代码形式。如果选择第一项 Release，那么生成的就是最终代码，其运行效率会增高。

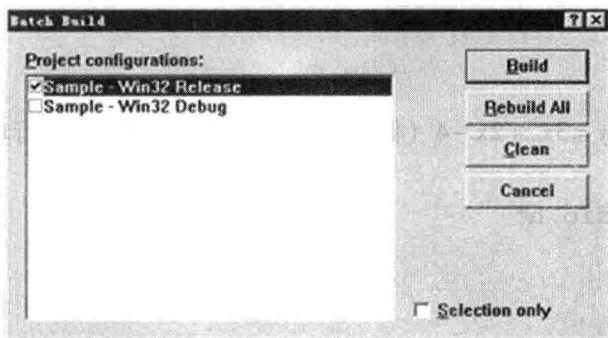


图 1-7 选择可执行程序类型

选择 Sample - Win32 Release 项，再进行 Build 或 Rebuild All 就会在工程所在的目录下产生一个新的目录 release，在 release 目录下生成的可执行程序代码规模小，执行效率高，是我们最后的产品。

【说明】 在 FileView 中对着一个工程单击鼠标右键后，关联菜单中有一个“Clean”命令，在此特地要解释一下它的功能：VC6 在建立（Build）一个工程时，会自动生成很多中间文件，如预编译头文件、程序数据库文件等。这些中间文件加起来的大小往往有数兆，很多人在开发一个软件期间会使用办公室或家里的数台机器，如果不把这些中间文件删除，在多台机器之间使用软盘拷贝工程就很麻烦。“Clean”命令的功能就是把 VC6 生成的中间文件全部删除，避免了手工删除时可能会出现误删或漏删的问题。另外，在某些情况下，VC6 编译器可能无法正确识别哪些文件已被编译过了，以致于在每次建立工程时都进行完全重建，很浪费时间，此时使用“Clean”命令删除掉中间文件就可以解决这一问题。

实训报告要求：

将以上各题的源程序、运行结果，在实训中遇到的问题和解决问题的方法，以及实训过程中的心得体会，写在实训报告上。

实训二 TC 中的 C 语言的运行环境和运行过程

实训目的:

- (1) 熟悉 TC 运行环境, 进入 Turbo C 集成环境, 熟悉 Turbo C 主菜单下各选择项的功能及功能键的使用。
- (2) 练习简单 C 语言程序的编写。
- (3) 熟练掌握 C 语言程序的上机运行步骤。

实训内容:

1. 实训要求

编程求计算函数 $y = 3x^2 + 2x - 4$ (假设 $x = 2$) 的值, 熟悉 TC 运行环境。

2. 源代码

```
#include "stdio.h"
void main( )
{
    int x=2,y;
    y=3*x*x+2*x-4;
    printf("\ny = %d",y);
}
```

在 TC 环境中输入上面的代码后, 先编译、连接, 再保存, 然后再运行。

具体操作步骤如下所示:

- (1) 启动 TC 集成环境。
- (2) 编辑一个新文件。按 F10, 将亮条移到 File 处按回车键, 再选 New 后按回车键 (为简化起见, 以后用 File→New 的形式表示上述操作), 出现子窗口, 在其中输入要创建的文件名, 输入文件名后按回车键, 出现编辑窗口, 即可在此窗口中输入和修改源程序, 如图 1-8 所示。
- (3) 保存文件。按 F10, 选择菜单 File→Save, 即可以保存当前文件到外存。
- (4) 编译文件。按 F10, 选择菜单 Compile→Compile to obj, 就可以对程序进行编译。在此过程中可能会出现错误信息, 并进行修改, 修改完后存盘再进行编译, 直至成功。
- (5) 运行文件。按 F10, 选择 Run→Run 命令运行文件, 如图 1-9 所示。

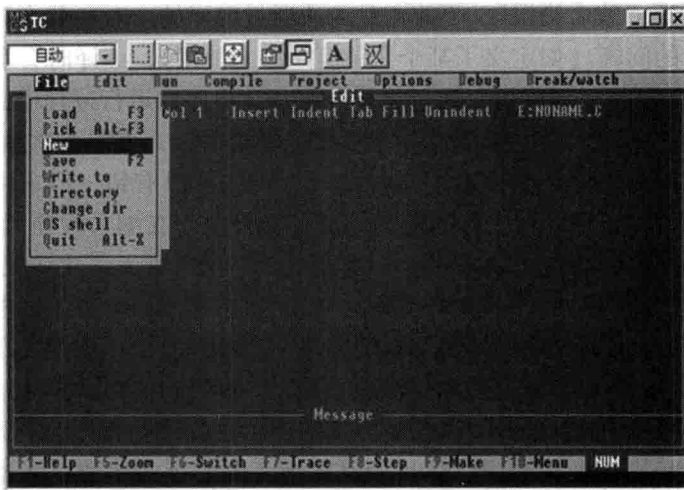


图 1-8 TC 的 File→New 窗口



图 1-9 TC 的 Run→Run 命令窗口

(6) 查看运行结果。按 F10，选择菜单 Run→User screen 即可以查看程序运行结果。

【注意】 若编译没有成功，则程序不能运行；若编译成功，则可以运行程序。但若运行结果不正确，则表明程序有逻辑错误，需要对程序进行调试和修改，直至程序运行结果正确。

用户以后每次上机运行一个 C 程序都要经过上述操作步骤②~⑥，因此，需要深刻理解和熟练掌握这些步骤。

3. 学会调入一个已存在的 C 源文件

按 F10，选择菜单 File→Load，出现子窗口，在其中输入已存在的文件名，按回车键后即可调入一个已存在的文件。

4. 学会退出 Turbo C

暂时退出：按 F10，选择菜单 File→OS shell。此时按 Exit 即可以返回主菜单。

永久退出：按 F10，选择菜单 File→Quit。

【说明】 C 程序的错误类型包括：①语法错误。语法错误是指不符合 C 语言的语法规定。例如，左右括号数量不匹配、语句最后漏了分号等。这些错误会在编译时被发现并指

出。这些错误都属于“致命错误”，不改正是无法通过编译的。对一些在语法上有轻微毛病但不影响程序运行的问题（如定义了某个变量但始终未使用），编译时会发出“警告”，“警告”不属于“致命错误”，程序可以通过编译。但是在程序的运行过程中，这些“警告”可能造成一些运行时的错误。

②逻辑错误。程序没有语法错误，可以正常运行，但结果不对。例如，要求 a 和 b 的和，可是却写成了“ $a - b$ ；”。语法上没有错，但求出的却是 a 和 b 的差。这类错误可能是由于设计算法时的错误，也可能是由于在编写程序时出现的疏忽所致。这类错误计算机是无法检查出来的。

③运行错误。有时程序既无语法错误，又无逻辑错误，但程序不能正常运行或结果不对，多数情况下，这是由于数据不正确，或数据类型不合适造成的。例如，要求 a/b ，如果 b 的值为非 0 则程序不会有问题，而如果 b 的值为 0 就会造成溢出（over-flow）错误。初学者常犯的错误是认为“编译和连接”都正确，程序就应该没有问题，怎么会结果不对呢？“编译和连接”都正确，只能说明程序没有语法和拼写上的错误，但在算法（逻辑）上有没有错，还得看结果对不对。反过来讲，无论让你设计一个什么样的程序，你都只写以下几行，则“编译和连接”肯定都正确，但能实现设计的要求吗？

```
#include <stdio.h>
void main()
{
    printf("Hello World!\n");
}
```

事实上，程序设计重点完全不是修正编译和连接过程中的错误。相对而言，这种工作基本没有技术含量，程序设计的主要工作是设计正确的算法。

实训报告要求：

将以上各题的源程序、运行结果，在实训中遇到的问题和解决问题的方法，以及实训过程中的心得体会，写在实训报告上。