

重 庆 市 高 等 院 校 规 划 教 材
普通高等院校应用型本科计算机基础教育系列教材

C语言程序设计实训教程

—— 上机指导与等级考试辅导

C YUYAN CHENGXU SHEJI SHIXUN JIAOCHENG
—— SHANGJI ZHIDAO YU DENGJI KAOSHI FUDAO

重庆市教育委员会
重庆市高校计算机基础教育研究会 组编

主 编 张 涛 陈立志
副主编 高占国 杨永斌 邹永贵



重庆大学出版社
<http://www.cqup.com.cn>

重 庆 市 高 等 院 校 规 划 教 材
普通高等院校应用型本科计算机基础教育系列教材

C语言程序设计实训教程

—— 上机指导与等级考试辅导

C YUYAN CHENGXU SHEJI SHIXUN JIAOCHENG
—— SHANGJI ZHIDAO YU DENGJI KAOSHI FUDAO

重庆市教育委员会
重庆市高校计算机基础教育研究会 组编

主 编 张 涛 陈立志
副主编 高占国 杨永斌 邹永贵



重庆大学出版社

内容简介

本教材作为《C 语言程序设计实例教程》教材的配套书籍,本书主要有两方面的内容:C 语言实验环境和上机练习题、C 语言知识点及 C 语言等级考试讲解和练习。

C 语言实验环境和上机练习题部分按照《C 语言程序设计实例教程》教材内容的顺序,精心设计了程序验证性实验、程序填空性实验和程序设计性实验三种模式的实验。程序设计性实验介绍了相关程序的设计思路和主要步骤,旨在提高学生的实际动手能力。C 语言知识点及 C 语言等级考试讲解部分按照各章节的知识点进行归纳和整理,同时通过实际的等级考试试题的讲解来加深学生的理解和应用。

本教材适合作为高等教育应用型本科计算机基础 C 程序设计课程的配套教材,也可作为全国计算机等级考试培训教材和复习参考书。

图书在版编目(CIP)数据

C 语言程序设计实训教程:上机指导与等级考试辅导/
张涛,陈立志主编. —重庆:重庆大学出版社,2009. 8
(普通高等院校应用型本科计算机基础教育系列教材)
ISBN 978-7-5624-4998-0

I. C… II. ①张…②陈… III. C 语言—程序设计—高等学校—教学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 138027 号

重庆市高等院校规划教材
普通高等院校应用型本科计算机基础教育系列教材

C 语言程序设计实训教程 ——上机指导与等级考试辅导

重庆市教育委员会
重庆市高校计算机基础教育研究会 组编

主 编 张 涛 陈立志

副主编 高占国 杨永斌 邹永贵

责任编辑:王海琼 钟 渊 版式设计:王海琼

责任校对:谢 芳 责任印制:赵 晟

*

重庆大学出版社出版发行

出版人:张鸽盛

社址:重庆市沙坪坝正街 174 号重庆大学(A 区)内

邮编:400030

电话:(023) 65102378 65106042(24 小时录音电话)

传真:(023) 65103686 65105565

网址:<http://www.cqup.com.cn>

邮箱:fxk@cqup.com.cn (营销中心)

全国新华书店经销

重庆科情印务有限公司印刷

*

开本:787×1092 1/16 印张:10.75 字数:268千

2009 年 8 月第 1 版 2009 年 8 月第 1 次印刷

印数:1—3 000

ISBN 978-7-5624-4998-0 定价:16.00 元

本书如有印刷、装订等质量问题,本社负责调换
版权所有,请勿擅自翻印和用本书
制作各类出版物及配套用书,违者必究

前言

本教材是与《C 语言程序设计实例教程》相配套的实验指导和等级考试辅导书。本书的目的有两个方面:一是通过一系列的基础实验和综合性实验使学生巩固所学的知识,掌握 C 语言程序设计的基本技能;二是方便学生复习迎考计算机等级考试中的二级 C 语言考试。通过对知识点、考点的梳理,实际的等考试题的解析来讲解相应的知识点和考点,使学生对二级 C 语言考试有直观的体验,有的放矢地准备等级考试。本书以实训为主线,针对性强,突出 C 语言程序设计基本技能训练,突出 C 语言等级考试复习要点,通过实训使学生熟练掌握程序设计和答题方法和技巧,使学生能顺利通过考试。

本书由以下两部分组成:

(1) C 语言上机实训

上机实训中第一章介绍了 C 语言上机的常用编程环境,包括 Turbo C 编程环境和 Visual C++ 6.0 编程环境。第二章按照主教材的章节安排相应的基础性实验,包括程序验证性实验、程序填空性实验和程序设计性实验 3 种类型的实验。上课教师可以根据教学情况选择搭配上机实训题。第三章通过 3 个综合性实验来提高学生的 C 语言的程序设计技能,灵活应用 C 语言的数组和指针等数据类型。

(2) C 语言等级考试辅导

在这部分中按照主教材的章节梳理相应的知识点和考点;通过以前的真题来解析知识点的考核方式和应试技巧,针对性强,具有很强的实战性;每一章都配有相应的等级考试练习题供学生训练。

本教材由张涛、陈立志担任主编;邹永贵、高占国、杨永斌担任副主编。张涛编写了 C 语言上机实训部分;陈立志编写了 C 语言等级考试辅导的第 2—8 章及第 10 章,杨永斌编写了 C 语言等级考试辅导的第 1 章,邹永贵编写了 C 语言等级考试辅导的第 9 章及 C 语言上机实训的综合性实验部分。高占国对本书的结构和内容提出了宝贵的建议,在本书的编写过程中,杨秀清、黄海龙、黄世权参加了收集整理资料的工作,郑先鋒、王丽艳参加了本教材相关程序上机验证工作,在此表示感谢。全书由张涛、陈立志统稿、校稿。

由于时间仓促,书中难免存在不足之处,恳请广大读者提出宝贵意见, E-mail: enigmaer@163.com。

编 者

2008 年 12 月

目录

第1部分 C语言上机实训	1
第1章 C语言上机过程及编程环境	2
1.1 C语言程序上机过程	2
1.2 Turbo C 2.0 开发环境	3
1.3 Visual C++6.0 开发环境	13
第2章 基础实验	19
2.1 C语言概述	19
2.2 数据类型、运算符与表达式	21
2.3 数据的输入和输出	24
2.4 C程序控制结构	27
2.5 数组	31
2.6 函数	37
2.7 指针	41
2.8 结构体、共用体和枚举类型	46
2.9 文件	51
第3章 综合实验	56
3.1 综合实验1	56
3.2 综合实验2	57
3.3 综合实验3	58
第2部分 计算机等级考试C语言辅导	61
第1章 C语言概述	62
1.1 知识要点复习	62
1.2 等级考试题解析	63
1.3 等级考试题练习	63
第2章 数据类型、常量、变量、运算符与表达式	65
2.1 知识要点复习	65
2.2 等级考试题解析	68

C 语言程序设计实训教程

2.3 等级考试题练习	73
第3章 数据的输入和输出	74
3.1 知识要点复习	74
3.2 等级考试题解析	74
3.3 等级考试题练习	75
第4章 C 程序控制结构	76
4.1 知识要点复习	76
4.2 等级考试题解析	79
4.3 等级考试题练习	81
第5章 数组	83
5.1 知识要点复习	83
5.2 等级考试题解析	84
5.3 等级考试题练习	85
第6章 函数	87
6.1 知识要点复习	87
6.2 等级考试题解析	89
6.3 等级考试题练习	98
第7章 指针	100
7.1 知识要点复习	100
7.2 等级考试题解析	102
7.3 等级考试题练习	111
第8章 结构体、共用体和枚举类型	113
8.1 知识要点复习	113
8.2 等级考试题解析	117
8.3 等级考试题练习	122
第9章 文件	127
9.1 知识要点复习	127
9.2 等级考试题解析	130
9.3 等级考试题练习	131
第10章 C 语言上机考试典型试题	133
10.1 编程题	133
10.2 填空题	137
10.3 改错题	142
附录1 全国高等学校(重庆考区)非计算机专业计算机等级考试 C 语言笔试试题 样卷及解答	148
附录2 全国高等学校(重庆考区)非计算机专业计算机等级考试 C 语言上机试题 样卷及解答	160
参考文献	163

第 1 部分 C 语言上机实训

第 1 章 C 语言上机过程及编程环境

1.1 C 语言程序上机过程

C 程序开发过程与其他高级语言源程序开发过程一样,都必须先经过编辑、编译和连接过程,最后生成可执行代码后才能运行与调试。程序开发过程基本步骤如下:

①编辑:编辑是将编写好的 C 程序输入到计算机中,生成磁盘文件的过程。磁盘文件名字要用扩展名 .c,该文件称为 C 语言的源程序。例如:File.c。

②编译:编译是通过编译器将程序的源代码转换成为机器代码的过程,这种形式的代码称为目标代码,即机器语言指令。这时产生的文件扩展名为 .obj。在这个过程中,如果程序有语法错误,编译给出出错类型、出错位置等信息,供用户参考。例如:将源文件 File.c 编译后,生成目标文件 File.obj。

③连接:编译过后生成的目标文件仍然不是一个可执行的程序,因为目标代码只是一个一个的程序块,需要相互连接成为一个适应操作系统环境的程序整体。为了把它转换为可执行程序,必须进行连接。

连接是生成可执行文件的一个过程,它将用户程序生成的多个目标代码文件和系统提供的库文件(.lib)中某些代码连接在一起。经过这种连接处理,生成一个可执行文件,存储这个可执行文件的扩展名为 .exe。例如:File.obj 文件经连接处理后,生成可执行文件 File.exe。

④运行与调试:运行与调试的目的是验证程序功能的正确性。当执行结果与程序应该得到的结果不符时,则说明程序的算法有错误,这类错误一般称为逻辑错误。因此必须通过调试的手段查处逻辑错误,并在源程序中加以修改,再重复以上过程,直到程序运行正确为止。运行可执行文件的方法很多,一般在编译系统中通过选择菜单项的方式实现。

程序在被编译、连接的过程中,难免会出现错误,这时系统会给出致命错误和警告错误的提示,这时需要返回上一步过程中,对源程序进行修改,直至没有错误提示为止。对于程序中存在的致命错误必须要进行修改,否则系统不能生成可执行文件。当程序中存在警告错误时,虽然系统也会生成可执行文件,但是一般要求修改警告错误后再去运行可执行文件,因为有的警告错误会造成结果的错误。

C 程序整个开发过程如图 1.1 所示。

目前的 C 语言开发环境对以上流程的各个操作进行了集成,形成了所谓的 IDE(Integrated Development Environment),即集成开发环境。用户的编辑、编译、连接、运行调试都是

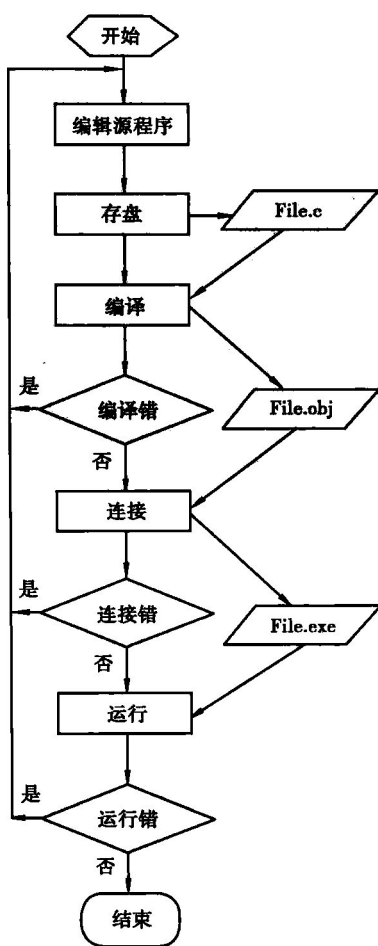


图 1.1 C 程序开发过程

在一个集成界面的环境下实现的,从而方便了用户的使用。下面介绍两种常用的集成开发环境的基本使用。

1.2 Turbo C 2.0 开发环境

Turbo C 2.0 是 Borland 公司推出的一个集程序编辑、编译、连接、调试为一体的 C 程序开发软件,简称 TC2.0。它具有速度快、效率高、功能强、使用方便等优点。用户在这个集成环境下,可以利用内部的编辑器进行全屏幕编辑,也可以利用窗口功能进行编译、连接、调试、运行、环境设置等工作。由于 TC2.0 是在 DOS 操作系统下运行的,其简单易于掌握的特点给初学者带来了很大的方便,故 TC2.0 依然是 C 语言教学的很好选择。

1) TC2.0 的启动

如果你的计算机系统已经安装了 Turbo C 2.0,则在 DOS 命令状态下输入命令:

TC 或 TC filename. c,其中 filename 是用户需要进行编辑、编译、连接、运行的 C 程序的

文件名。在输入命令 TC 的情况下,该文件名将在进入集成环境后再指定。

如果是在 Windows 环境下启动 TC,只需要找到 TC. exe 文件或它的快捷方式双击即可。进入 TC2.0 集成环境后,首先在屏幕上显示 TC2.0 的主菜单窗口,如图 1.2 所示。

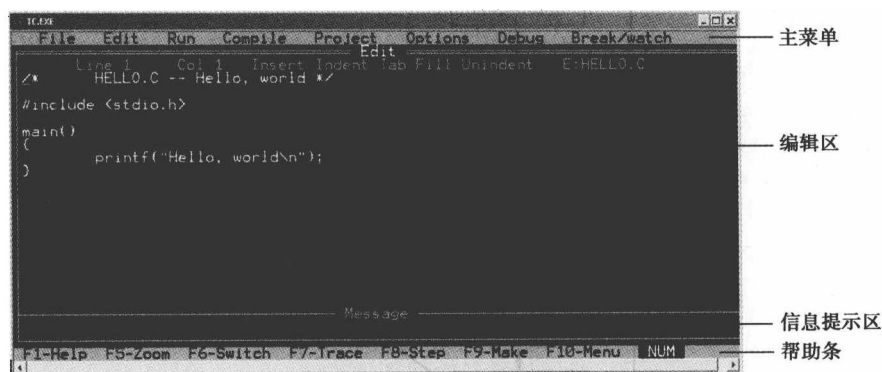


图 1.2 TC2.0 集成开发环境

2) TC2.0 的菜单

从图 1.2 可以看出,在主菜单中,有 8 个菜单条目,每一个菜单条目的功能如下:

- File 处理文件(包括装入、存盘、选择、建立、换名写盘),目录操作(包括列表、改变工作目录),退出系统以及调用 DOS。
- Edit 建立、编辑源文件。
- Run 控制运行程序。如果程序已经编辑连接好,且 Debug/Source Debugging 以及 Option/Compiler/Code Generation/Obj Debug Information 开关设置为 ON,则可以用此菜单初始化调试阶段。
- Compile 编译并生成目标程序与可执行文件。
- Project 允许说明程序中包含哪些文件的管理条目。
- Options 可以选择集成环境任选项(如存储模式、编译时的任选项、诊断及连接任选项)以及定义宏;也可以记录 Include, Output 及 Library 文件目录,保存编译任选项和从配置文件加载任选项。
- Debug 检查、改变变量的值,查找函数程序运行时查看调用栈。选择程序编译时是否在执行行代码中插入调试信息。
- Break/watch 增加、删除、编辑监视表达式,以及设置、清除、执行至断点。

特别要指出的是,除了 Edit 菜单项外,每一个菜单项以对应一个子菜单。选择 Edit 项目后,只是进入编辑器。

为了从主菜单中选择所需要的功能,可以用以下两种方式:

- 按 F10 键后,可以看到屏幕上部主菜单中的某个条目处处于高亮状态,此时,利用左、右光标移动键(←与→)移到所要选择的菜单条目位置处,然后按回车键,即出现相应的子菜单。
- 直接按 Alt + 主菜单条目中的首字母(分别为 F, E, R, C, P, O, D, B),此时就会出现相应的子菜单。例如,按 Alt + F 键表示选择文件子菜单(File)。

当出现子菜单时,其中某个条目是高亮度的,此时可以利用上、下光标移动键(↑与↓)来移动该高亮度线,从而选择所需要的功能。在主菜单或通过主菜单调用的任意一个子菜单中,按 Esc 键后将直接返回到活动窗口。

下面简要介绍各子菜单的功能。

(1) 文件(File)菜单

当选中 File 子菜单后,在“File”下方将出现一个子窗口,如图 1.3 所示。在这个子窗口中,有的条目右边还标出了实现该功能的热键。

小知识

所谓“热键”,是指为执行菜单中某一固定功能而设置的键。通过热键来实现某种功能,比通过菜单选择更简单直接,但要求用户熟记这些热键。例如,为了选择“文件子菜单(File)”,除了通过主菜单选择以外,还可以直接用热键 Alt + F 来选择。



图 1.3 文件菜单中的子菜单

下面简要说明各项的功能:

- Load(加载) 装入一个文件。当给定的文件名中有文件名通配符(*或?)时,将进行列表选择。
- Pick(选择) 将最近装入编辑窗口的 8 个文件列成表,供用户选择,选择后又装入编辑器,光标置在上次修改过的地方。若选了“…LOAD FILE…”,屏幕上将出现“LOAD FILE NAME”提示框。
- New(新文件) 新建一个编辑文档,装入编辑器的缺省文件名为 NONAME. C。
- Save(存盘) 将编辑器中的内容存盘。若文件名为 NONAME. C,而又要存盘,编辑器会询问是否要改名。
- Write to(存盘) 把编辑器中的内容写入指定的文件中。若该文件已经存在,则导致重写。
- Directory(目录) 显示目录与所需文件列表。热键 F4 改变匹配符,选择文件名后,将该文件装入编辑器。
- Change dir(改变当前目录) 显示当前目录,或改变当前驱动器与目录。

• Os shell(暂时退出) 暂时退出 Turbo C,转到 DOS 状态,在 DOS 状态下用 Exit 命令又可返回 Turbo C。此功能对于在想运行 DOS 命令但又不想退出 Turbo C 时非常有用。

• Quit(退出) 退出 Turbo C,返回到 DOS 状态。

(2) 编辑(Edit)菜单

调用内部编辑器。在编辑器中按 F10 键可返回主菜单(或 Alt + 所需主菜单命令的首字母),但此时编辑器中的内容仍保持在屏幕上。在主菜单中按 Esc 或 E 键即可回到编辑器(按 Alt + E 键也可,且在任何时候都起作用)。

进入编辑窗口后,编辑窗口的名字是高亮度的,表示它是活动窗口。窗口的顶部第一行是状态行,给出有关正在被编辑文件的信息,如当前光标所在的行、列;编辑模式(插入/改写);正在编辑文件的文件名等。当需要编辑修改时,在编辑窗口中灵活地使用光标移动键以及编辑命令即可达到预期的效果。常用的编辑命令见表 1.1 所示。

表 1.1 TC2.0 常用编辑键

功能键	命令功能
Home	将光标移到行首
End	将光标移到行尾
Ins	插入/改写两种状态的切换
Del	删除光标所在的字符
PgUp	向上翻页正文
PgDn	向下翻页正文
Ctrl + Y	删除光标所在的行
Ctrl + T	删除光标所指的一个单词
Ctrl + U	放弃操作
Ctrl + KB	设置(开始)块头标志
Ctrl + KK	设置(结尾)块尾标志
Ctrl + KV	块移动到光标处
Ctrl + KC	块拷贝到光标处
Ctrl + KY	块删除
Ctrl + KH	取消块定义

(3) 运行(Run)菜单

当选中 Run 菜单后,在“Run”下方将出现一个子菜单窗口,如图 1.4 所示,其中也列出了对应的热键。

下面简要说明各项的功能:

• Run(运行) 运行当前程序。

• Program reset(程序重启动) 中止当前调试,释放分配给程序的空间,关闭已打开的



图 1.4 运行菜单中的子菜单

文件。

- Go to cursor(执行到) 使程序从执行长条运行到编辑窗口中光标所在行。若光标所在行不含可执行代码语句,则显示一个 Esc 框作警告。
- Trace into(跟踪进入) 运行当前函数中的下一个语句。若此语句不含调试器可访问的函数调用,则停在下一条可执行语句上;但若此语句含有调试器可访问的函数调用,则停在函数定义的开始。
- Step over(单步执行) 执行当前函数的下一语句,即使遇到调试语句可访问的函数调用,也不会跟踪进入下一级函数中。
- User screen(用户屏幕) 切换到用户屏幕。

(4) 编译(Compile)菜单

当选中 Compile 子菜单后,在“Compile”下方将出现一个子菜单窗口,如图 1.5 所示。

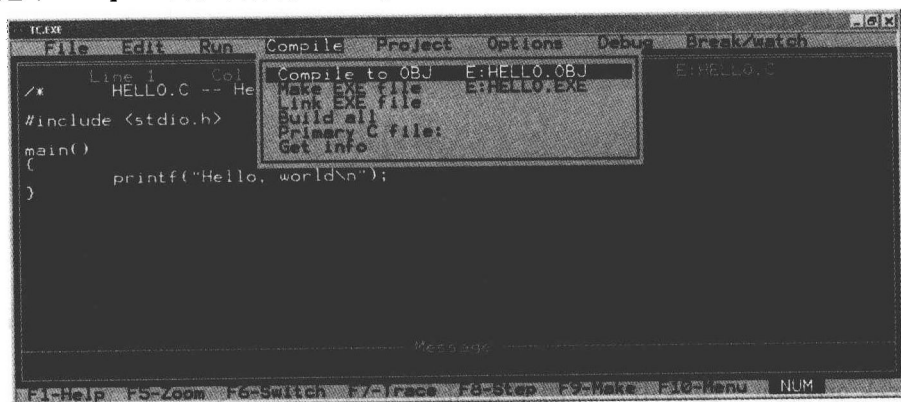


图 1.5 编译菜单中的子菜单

下面简要说明各项的功能:

- Compile to obj(编译生成目标代码) 本命令将一个.c 源文件编译成.obj 文件,同时显示生成的文件名。obj 文件由源.c 文件名产生;或在没有指定文件名时,由上次装入编辑器的文件名产生。Turbo C 在编译时弹出一个窗口,用于显示编译结果。在编译/组装

(Make)完后,按任意键将清除编译窗口。此时若发现有错误,则转到消息窗口的的第一个错误处(有亮度标志)。

本命令的热键为 Alt + F9。

- **Make exe file(生成执行文件)** 调用本命令来生成 .EXE 文件,并显示所生成的 .EXE 文件名。 .EXE 文件名是依次由下列文件名产生的:Project/Project name 说明的文件名;或 Project c file 说明的文件名;或上次装入窗口的文件名。

本命令的热键为 F9。

- **Link EXE file(连接执行文件)** 把当前文件与库文件(既可以是缺省的,也可以是定义在当前项目文件中的)连接在一起,生成 .EXE 文件。

- **Built All(建立所有文件)** 重建项目中的所有文件。本命令类似于 Compile/Make EXE File,只是它是无条件执行的,而 Compile/make EXE File 只重建那些非过时的文件。本命令首先将所有的 Project 文件中的 .obj 的日期与时间置为 0,然后再组装(Make)。这样,若用户因 Ctrl + Break 键中断了 Build All 命令,只要用 Compile/Make EXE File 即可恢复。

- **Primary C file(主 C 文件)** 当编译多个 .H 头文件单个 .C 文件时,Primary C File 命令是很有用的(但并非是必要的)。若在编译过程中发现错误,包含错误的文件(.C 或 .H)将被自动装入编辑器,可对其进行修改。但必须注意,.H 文件只有在已将 Option/Environment/Message Tracking 缺省设置改为 All File 时才能自动装入,而原缺省设置不会自动加载 .H 文件。即使 .C 文件不在编辑器,但只要一按热键 Alt + F9, C 主文件即被重新编译。

- **Get Info(获得信息)** Compile/Get Info 开辟一窗口,给出如下信息:

源文件;

与当前文件相联系的目标文件名;

当前源文件名;

当前源文件字节数;

程序退出码;

可用空间。

(5) 项目(Project)菜单

当选中 Project 子菜单后,在“Project”下方将出现一个子菜单窗口,如图 1.6 所示。

下面简要说明各项功能:

- **Project name** 选择一个包含将要编译连接的文件名的 Project 文件,项目名也将是以后要建立的 .EXE 或 MAP 文件名。典型的项目文件具有 .PRJ 扩展名。

- **Break make on** 提供用户说明中止 Make 的缺省条件,如警告(Warnings)、错误(Errors)、致命错误(Fatal Error)。

- **Auto dependencies(自动依赖)** 这是一个开关,当置为 On 时,项目组装(Project-Make)自动检查每个项目表中在磁盘上有相应 .C 文件的那些 .OBJ 文件的源文件的日期/时间信息与 .OBJ 文件的依赖关系。所谓自动依赖关系检查是指:项目组装打开 .OBJ 文件,寻找包含在源代码的那些文件的有关信息。此时,把每个组成 .OBJ 文件的日期/时间信息与 .OBJ 中的进行比较,若不同,则重新编译 .C 源文件。若 Auto dependencies 开关置为



图 1.6 项目菜单中的子菜单

Off,则不进行这种检查。

- Clear project(清除 project) 清除项目文件名,重置消息窗口(Message Window)。
- Remove message(删除信息) 该命令把错误信息从消息窗口中清除掉。

(6) 配置(Option)菜单

当选中 Option 菜单后,在“Option”下方将出现一个子菜单窗口,如图 1.7 所示。



图 1.7 配置菜单中的子菜单

下面简要说明各项的功能:

- Compile(编译器) 本命令将产生一个子菜单,为用户提供说明硬件配置、存储模式、调试技术、代码优化、诊断消息控制以及宏定义等。各菜单条目如下:

Model:选择存储模型。

Define:打开一个宏定义框。

Code generation:代码生成。

Optimization:优化用户代码。

Source:处理源代码。

Errors:处理和响应诊断信息。

Names:改变代码、数据等。

其中每一个条目又对应一个子菜单,供用户选择各种功能。详细介绍请参看 Turbo C 的用

户手册。

- **Linker(连接器)** 本命令将产生有关连接器的设置。它包括以下内容:

Map file:选择映射文件的类型 On/Off,缺升值为 Off。

Initialize segments:段初始化 On/Off,缺升值为 Off。

Default libraries:缺省库 On/Off,缺升值为 On。

Graphics library:图形库 On/Off,缺升值为 On。

Warn duplicate symbols:警告重复字符 On/Off,缺升值为 On。

Stack warning:堆栈警告 On/Off,缺升值为 On。

Case-sensitive link:大小写区别连接 On/Off,缺升值为 On。

- **Environment(环境设置)** 本命令将产生编译环境的设置。它包括以下内容:

Message tracking:消息跟踪 Current File/All Files/Off,缺升值为 Current File。

Keeping message:保存消息 Yes/No,缺升值为 No。

Config auto save:配置自动保存 On/Off,缺升值为 On。

Edit auto save:编辑自动保存 On/Off,缺升值为 Off。

Backup files:备份文件 On/Off,缺升值为 On。

Tab size:制表键大小,缺升值为 8。

Zoomed window:放大窗口 On/Off,缺升值为 Off。

Screen size:选择屏幕显示行数。

- **Directories(目录)** 本命令告诉 T C 到哪里去寻找编译连接所需的文件,生成的可执行文件放到何处,在哪里查找配置文件。具体内容如下:

Include directories:包含目录,如 C:\TURBOC\INCLUDE。

Library directories:库目录,如 C:\TURBOC\LIB。

Output directory:输出目录。

Turbo C directory:Turbo C 目录。

Pick file name:pick 文件名。

Current pick file:当前 pick 文件。

- **Arguments(参数)** 本设置允许用户给出运行程序命令行。

- **Save options(保存任意项)** 将选择的编辑器、连接环境、调试和 project 任选项保存到一个配置文件中(缺省文件名为 TCCONFIG.TC)。启动时,Turbo C 再到 T C 目录中去寻找同样的文件。

- **Retrieve options(恢复任选项)** 加载以前用 Option/Save options 命令保存的配置文件。

(7) 调试(Debug)菜单

当选中 Debug 菜单后,在“Debug”下方将出现一个子菜单窗口,如图 1.8 所示。

下面简要说明各项功能:

- **Evaluate(计算)** 计算变量或表达式值,并显示其结果。

- **Call stack(调用栈)** 本命令显示一个调用栈的弹出窗口。调用栈显示程序运行到正在运行的函数时调用的函数序列。其中主函数 Main 在栈底,正在运行的函数在栈顶。



图 1.8 调试菜单中的子菜单

调用函数的每一项显示了函数名以及传递给它的参数值。

- Find function(查找函数定义) 显示编辑窗口每一函数的定义。只有在调试阶段才能使用本命令。
- Refresh display(刷新显示器) 如果编辑屏幕被重写,使用本命令可以恢复当前屏幕的内容。
- Display swapping(显示转换) 本命令提供 3 种选择:On(缺省)、Always 和 None。
- Source debugging(源代码调试) 本命令提供 3 种选择 On(缺省)、Standalone 和 None。

(8) 中断/监视(Break/watch)菜单

当选中 Break/watch 菜单后,在“Break/watch”下方将出现一个子菜单窗口,如图 1.9 所示。使用本命令可以控制断点和监视表达式。



图 1.9 中断/监视菜单中的子菜单

下面简要说明各项的功能:

- Add watch(增加监视表达式) 向监视窗口插入一个监视表达式。
- Delete watch(删除监视表达式) 从监视窗口中删除当前监视表达式。
- Edit watch(编辑监视表达式) 选择本命令后,调试器弹出一个含有当前监视表达式拷贝的窗口。