

高等院校计算机应用系列教材

计算机基础与 C 语言程序 设计实验指导 (第四版)

焉德军 刘明才 主 编
王 鹏 辛慧杰 副主编

清华大学出版社

北 京

内 容 简 介

本书是《计算机基础与C语言程序设计(第四版)》的配套实验教材。全书共分4篇：第一篇是C语言程序设计实验指导，介绍了如何使用Visual C++ 2010进行程序的编辑、编译和错误处理，同时结合主教材的内容提供了11个实验，给出了实验目的、实验内容及程序提示；第二篇是《计算机基础与C语言程序设计(第四版)》习题解答；第三篇是全国计算机等级考试二级C介绍，包括全国计算机等级考试大纲、数据结构与算法、软件工程基础、数据库基础、公共基础知识答案和全国计算机等级考试模拟试题；第四篇是MATLAB软件入门。

本书内容丰富，实用性强，是学习C语言程序设计十分有用的一本参考书，既适合高等学校师生和计算机等级考试培训使用，也可供自学者参考。

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。举报：010-62782989，beiqinquan@tup.tsinghua.edu.cn。

图书在版编目(CIP)数据

计算机基础与C语言程序设计实验指导/焉德军，刘明才主编. —4版. —北京：清华大学出版社，2021.7

高等院校计算机应用系列教材

ISBN 978-7-302-58566-4

I. ①计… II. ①焉… ②刘… III. ①电子计算机—高等学校—教学参考资料 ②C语言—程序设计—高等学校—教学参考资料 IV. ①TP3

中国版本图书馆CIP数据核字(2021)第132328号

责任编辑：胡辰浩

封面设计：高娟妮

版式设计：妙思品位

责任校对：成凤进

责任印制：刘海龙

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>，<http://www.wqbook.com>

地 址：北京清华大学学研大厦A座 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：天津鑫丰华印务有限公司

经 销：全国新华书店

开 本：185mm×260mm 印 张：18.75 字 数：480千字

版 次：2012年8月第1版 2021年8月第4版 印 次：2021年8月第1次印刷

定 价：79.00元

产品编号：089483-01

在信息社会，信息的获取、存储、传输、处理和应用能力越来越成为个人的一种最基本的生存能力，正逐步被社会作为衡量一个人文化素质高低的重要标志。目前，计算机应用能力已成为影响人们生活方式、学习方式和工作方式的重要因素。大学计算机基础课程，作为非计算机专业学生的必修基础课，其教学目标就是为学生提供计算机方面的知识、能力与素质的教育，培养学生掌握一定的计算机基础知识、技术与方法，以及增强利用计算机解决本专业领域中问题的意识与能力。

多年来，大学计算机基础教学形成了大一上学期讲授大学计算机基础课程、下学期讲授计算机程序设计基础课程的教学模式。目前，绝大多数二本院校依然采取这种教学模式。这种模式在实践中存在如下弊端。

第一，因城乡、地区的差别，新生入学时计算机水平参差不齐，给教学带来很大困难。随着我国中小学信息技术教育的逐步普及，高校新生计算机知识水平的起点也逐年提高。同时，由于我国中学信息科学教育水平的不平衡，来自城市的学生入学时已经具备计算机的基本技能，而来自农村的一些学生，特别是来自西部欠发达地区和少数民族地区的一些学生，入学时才刚刚接触计算机。这种差异使计算机基础教学的组织与安排非常困难。

第二，学时少、内容多、周期短，并且与专业课学习脱节，严重影响了学生的学习积极性和程序设计思想的培养。在大一上学期讲授大学计算机基础课程时，由于内容宽泛，涉及面广，每堂课要讲授或上机练习的内容又多，计算机基础知识好一点的学生上课不愿意听讲、不屑于练习，而计算机基础知识相对差一点的学生又听不懂，极大地挫伤了学生学习计算机知识的兴趣和积极性。大一下学期讲授计算机程序设计基础课程，由于学时少、周期短，在教学中普遍缺乏利用程序设计解决实际问题 and 专业问题能力的训练，学完计算机程序设计基础课程后，多数学生还不能真正领会计算机的强大功能，不能利用所学的计算机知识解决相关的专业问题。

第三，计算机基础教学与大学生对获取全国计算机等级考试证书的需求脱节。由于就业的压力，多数二本院校的学生在毕业时迫切需要获得全国计算机二级等级考试证书。但是，在传统的计算机基础教学模式下，学生最快在大二上学期才能参加全国计算机等级考试，一次性过级率相对较低。为在毕业前获得计算机二级证书，一些学生不得不一次又一次地参加校外培训，花费了很多精力。

针对计算机基础教学中存在的问题和不足，2009年，我校对计算机基础教学进行了改革，建立了以学生为本，以就业需求为导向，以实践能力、创新能力和计算机应用能力培养为目标，以大一下学期学生能够顺利参加全国计算机等级考试、提高我校的全国计算机等级考试过级率

为“抓手”，适合学生特点和需求、符合教育规律和学生认知心理的计算机基础教学的新教学内容体系和教学模式。

为了深化改革，顺应社会需求，在 2009 年改革的基础上，2019 年我们及时调整教学内容。新的教学内容体系和教学模式是：根据不同学科、专业学生的特点和需求，分别开设“C 语言程序设计”和“Python 语言程序设计”课程，并以程序设计课程为主线，以相应的全国计算机二级等级考试大纲为依据，对传统的大学计算机基础教学的两门课程“大学计算机基础”和“C(Python)语言程序设计”的教学内容进行梳理、整合，并将 MATLAB 基础知识和数学建模入门知识纳入计算机基础教学内容体系中来，使之更贴近学生的需求，更符合学生的认知规律，更有利于学生计算机应用能力的培养和信息素养的提高。

(1) 新的教学模式将传统的在下一下学期讲授的“C(Python)语言程序设计”课程提前至大一上学期开始，并延伸到大一下学期结束，更符合大学生的认知规律。大一上学期讲完“C(Python)语言程序设计”课程的基本内容，下学期通过参加 3 月下旬的全国计算机等级考试，使学生的程序设计基础知识得到强化；等级考试后，通过综合性、设计性实验，使学生的计算机应用能力得到进一步提高。

(2) 将与专业课学习密切相关的计算机软件(MATLAB 软件)纳入大一的计算机基础教学中来，并结合数学建模进行讲解，为学生学习后续专业课程和参加数学建模竞赛奠定了扎实的基础，对学生的实践能力、创新能力的培养起到了非常好的作用。

(3) 改革考核方式，实行阶段性滚动考试，加强学习过程的监督与考核，极大地提高了学生自主学习的积极性。基于“百科园通用考试平台”开发了 C 语言、Python 语言、Office 应用的题库，为学生自主练习提供了方便，并且为按单元进行的阶段性考试和期末考试提供了重要保障。

(4) 为提高学生使用软件解决实际问题和专业问题的能力，开设不同的模块课程，使学生可以根据自己的专业需求、兴趣爱好和个人能力等具体情况选修相应的课程，达到考核要求即可获得相应的学分。在这里可供学生选修的课程有“办公自动化应用”“网络工程师培训”“Flash 动画设计与制作”“Excel VBA 数据处理技术”“图形图像处理”“Office 2010 应用”“计算机组装、维护与应用软件实训”“音频视频处理”“Visio 图形化设计”“MATLAB 基础与应用技术”等。

多年的教学实践使我们体会到，新的教学内容体系和教学模式至少有以下 4 点好处。

第一，拉长了大学计算机程序设计基础课程的学习周期，由原来的一个学期变为现在的两个学期，分 3 个阶段实施，符合学生的认知规律，并且对培养学生的编程思想和利用计算机解决实际问题的能力非常有益。

第二，将获得全国计算机二级等级考试证书作为新生入学的第一个阶段性目标，可以使學生尽快摆脱刚入大学时的“迷茫”状态，有利于优良学风的建设。

第三，满足了学生对全国计算机二级等级证书的需求，增加了学生将来就业的筹码。

第四，提高了学生的素质，增强了学生的自主学习能力和利用软件解决实际问题的能力。

自 2010 年 5 月，我们成立了教材编写委员会，着手进行系列教材的编写工作以来，目前《计

计算机基础与 C 语言程序设计》这套教材已经出版到第四版。

《计算机基础与 C 语言程序设计》教材包括：计算机入门基础知识，全国计算机二级等级考试大纲所要求的程序设计相关内容。

《计算机基础与 C 语言程序设计实验指导》辅助教材包括：《计算机基础与 C 语言程序设计》习题解答、实验指导，全国计算机二级等级考试介绍(包括考试大纲、公共基础知识的相关内容以及模拟试题)以及应用软件选讲(MATLAB 软件简介)。

教材是体现教学内容和教学方法的知识载体，是进行教学的基本工具，是深化教育教学改革、全面推进素质教育、培养创新人才的重要保证。教材建设是提高教学质量的基础性工作，要为学生知识、能力、素质的协调发展创造条件。该套教材的使用，对我校全国计算机等级考试过级率的提高起到了至关重要的作用。与改革前相比，我校的全国计算机等级考试过级率提高了 20%。该套教材是 2013 年辽宁省教学成果二等奖“以就业需求为导向、以计算机应用能力培养为目标的计算机基础教学新模式”的主要成果之一，其中，《计算机基础与 C 语言程序设计》和《计算机基础与 C 语言程序设计实验指导》在 2014 年被评为辽宁省第二批“十二五”普通高等教育本科省级规划教材，《计算机基础与 C 语言程序设计》在 2020 年又被评为辽宁省省级优秀教材。

有关我校的计算机基础教学改革：2009 年 10 月，获得辽宁省教育教学改革项目立项；2010 年 1 月，在首届全国民族院校计算机基础课程教学研讨会上，我校做了“基于应用型人才培养的计算机基础教学课程体系及教学内容的探讨”的主题发言，初步介绍了计算机基础教学改革思路与设想，得到与会代表的热烈反响；2011 年 7 月，在辽宁省计算机基础学会年会上，我校做了“基于能力培养与等级考试需求的计算机基础教学改革”的主题发言，得到与会同行的充分肯定和兄弟院校的广泛关注；2011 年 11 月，获得国家民族事务委员会本科教学改革与质量建设研究项目立项；2012 年 10 月，在全国高等院校计算机基础教学研究学会学术年会上，发表了题为《新形势下的计算机基础教学改革实践——以大连民族学院为例》的学术论文，全面介绍了我校的计算机基础教学改革，获得优秀论文二等奖；2013 年 6 月，在国家民族事务委员会 2013 年民族院校教学观摩会上，我校介绍了计算机基础课教学改革的经验，并于同年获得辽宁省教学成果二等奖；2020 年 10 月，在辽宁省计算机基础教育学会 2020 年学术年会上，我校介绍了计算机基础教学改革的新思路。

该套教材在第三版的基础上进行了修订，使教材结构更加合理。该套教材适合作为高等院校的计算机基础教学用书，也可作为学生自学计算机基础知识和相关程序设计基础知识、准备全国计算机二级等级考试的参考用书。

为了继续做好计算机基础教学的改革工作，我们热忱欢迎专家、同行以及广大读者多提宝贵意见！

焉德军
2021 年 4 月

前 言

本书是《计算机基础与 C 语言程序设计(第四版)》的配套实验指导书,全书共分 4 篇。

第一篇为 C 语言程序设计实验指导,包括 3 章内容。第 1 章介绍了使用 Visual C++ 2010 编辑、编译、运行 C 程序的方法。Visual C++ 2010 也是全国计算机等级考试(C 语言)指定的编译系统。第 2 章是上机实验部分,安排了 11 个实验,并根据实验内容分别安排了不同的学时——2 学时或 4 学时。实验进度与主教材同步。教师可根据具体的上机时长安排实验,并根据具体情况选取实验内容,还可根据每个学生的不同情况,适当安排必做题和选做题作为课后练习。学生在开展上机实验之前应做好实验准备,如阅读实验内容、复习主教材中的有关章节等。对于程序设计题,需要在进行上机实验之前编好程序。第 3 章给出了实验参考答案。

第二篇为《计算机基础与 C 语言程序设计(第四版)》习题解答。选择题和填空题主要测试读者对基本概念、基本理论和基本方法的掌握程度。在学完每章后,读者应独立将选择题和填空题做完,然后与给出的答案做比较,由此检查自己的学习情况。编程题的类型和数目较多,读者可根据自己的情况选做题目。每一道编程题的解法都有多种,书中只提供了一种答案,供读者参考和比较,以启发思路。本书所有程序都在 Visual C++ 2010 环境下调试通过,它们也可在 Dev C++环境下运行。

第三篇是全国计算机等级考试二级 C 介绍,包括全国计算机等级考试大纲、基础知识和模拟试题,另外还给出了模拟试题的解答。这部分可供参加全国计算机等级考试的读者参考使用。

第四篇是 MATLAB 软件入门,我们以 MATLAB 7.10.0 版本为例,介绍了 MATLAB 软件的使用方法及其在数学建模中的应用。

本书第一篇由刘明才编写,第二篇由辛慧杰编写,第三篇由王鹏编写,第四篇由焉德军编写。

在本书的编写过程中,得到了郑智强、张丽丽、邹冰冰和通拉嘎若曼等几位老师的热情帮助,另外我们还参考了一些网上资源,在此一并致谢。

本书作者长期从事高等学校计算机基础课程的教学工作,在总结多年的“大学计算机基础”“C 语言程序设计”课程的教学经验和教改实践的基础上,编写了本套教材。由于作者水平有限,书中难免存在不足,恳请读者批评指正。我们的邮箱是 992116@qq.com,电话是 010-62796045。

编 者
2021 年 4 月

目 录

第一篇 C 语言程序设计实验指导1

第 1 章 Visual C++ 2010 使用指南1

1.1 运行 C 程序1

1.2 打开 C 源文件8

1.3 调试 C 程序9

第 2 章 实验内容11

实验一 熟悉 VC 环境12

实验二 数据类型13

实验三 运算符和表达式14

实验四 选择结构15

实验五 循环结构17

实验六 数组19

实验七 函数21

实验八 指针24

实验九 结构体27

实验十 文件30

实验十一 综合设计33

第 3 章 实验参考答案34

实验一 熟悉 VC 环境34

实验二 数据类型35

实验三 运算符和表达式36

实验四 选择结构37

实验五 循环结构38

实验六 数组40

实验七 函数42

实验八 指针43

实验九 结构体45

实验十 文件47

实验十一 综合设计49

第二篇 《计算机基础与 C 语言程序设计(第四版)》习题解答55

第 1 章 习题解答55

一、选择题55

二、填空题57

第 2 章 习题解答57

一、选择题57

二、填空题58

三、编程题59

第 3 章 习题解答60

一、选择题60

二、填空题63

三、编程题64

第 4 章 习题解答66

一、选择题66

二、填空题68

三、编程题69

第 5 章 习题解答71

一、选择题71

二、填空题74

三、编程题75

第 6 章 习题解答81

一、选择题81

二、填空题84

三、编程题84

第 7 章 习题解答90

一、选择题90

二、填空题93

三、编程题95

第8章 习题解答	100	第3章 软件工程基础	158
一、选择题	100	3.1 软件工程的基本概念	158
二、填空题	102	3.2 软件需求分析	159
三、编程题	103	3.3 软件设计	160
第9章 习题解答	104	3.4 程序设计基础	164
一、选择题	104	3.5 软件测试及调试	167
二、填空题	107	3.6 习题	169
三、编程题(要求使用指针完成)	110	第4章 数据库基础	171
第10章 习题解答	116	4.1 数据库系统的基本概念	171
一、选择题	116	4.2 数据模型	175
二、填空题	120	4.3 关系代数	179
三、编程题	121	4.4 数据库设计	184
第11章 习题解答	125	4.5 习题	186
一、选择题	125	第5章 公共基础知识答案	189
二、填空题	127	5.1 数据结构与算法习题解答	189
三、编程题	127	5.2 软件工程基础习题解答	193
第三篇 全国计算机等级考试		5.3 数据库基础习题解答	196
二级C介绍	133	第6章 全国计算机等级考试	
第1章 全国计算机等级		模拟试题	200
考试大纲	133	6.1 模拟试题一	200
1.1 全国计算机等级考试(二级C)		6.2 模拟试题二	215
考试大纲	133	6.3 模拟试题三	229
1.2 全国计算机等级考试(二级公共		6.4 模拟试题四	244
基础)考试大纲	135	第四篇 MATLAB 软件入门	257
第2章 数据结构与算法	137	P.1 MATLAB 简介	257
2.1 算法	137	P.2 向量与矩阵	258
2.2 数据结构的基本概念	138	P.3 矩阵与数组的基本运算	262
2.3 线性表及其顺序存储结构	141	P.4 函数	263
2.4 栈和队列	142	P.5 MATLAB 绘图	265
2.5 线性链表	144	P.6 MATLAB 编程	273
2.6 树与二叉树	147	P.7 MATLAB 的符号运算	277
2.7 查找技术	152	P.8 MATLAB 在数学建模中的应用	280
2.8 排序技术	153	参考文献	291
2.9 习题	155		

C语言程序设计实验指导

第 1 章 Visual C++ 2010 使用指南

Visual C++ 软件是目前使用极为广泛的可视化开发环境,可用于对 C 程序或 C++程序进行各种操作,如建立、打开、浏览、编辑、保存、编译、链接和调试等。

本章主要介绍利用 Visual C++ 2010 集成环境(简称 VC 环境)对 C 程序进行编译和运行的一般方法。

1.1 运行 C 程序

使用 Visual C++ 2010 运行 C 程序一般分以下几个阶段。

1. 启动 Visual C++ 2010 集成环境

在 Windows 操作系统下,启动 Visual C++ 2010 系统的方法是:选择“开始”→“程序”→Microsoft Visual Studio 2010→Microsoft Visual C++ 2010 命令,则出现 Visual C++ 2010 集成环境窗口,如图 1-1 所示。

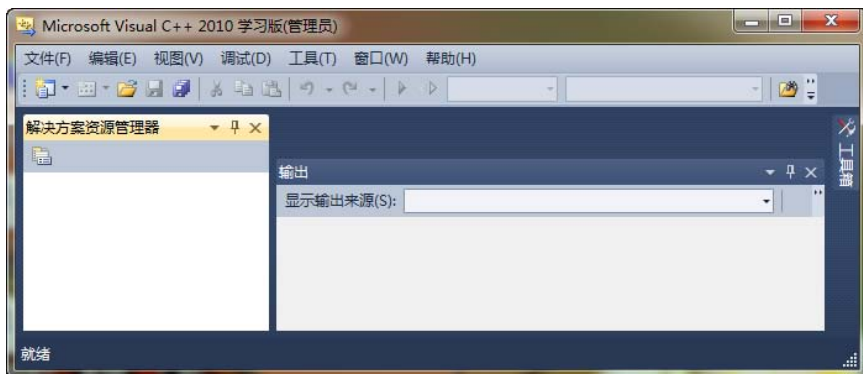


图 1-1 Visual C++ 2010 集成环境窗口

也可以在桌面上为 Visual C++ 2010 系统创建一个快捷方式, 利用该快捷方式启动 Visual C++ 2010 系统。

2. 创建项目

为了使用 Visual C++ 2010 系统运行 C 程序, 首先需要创建项目(Project)。项目中存放了 C 程序的所有信息。创建项目的步骤如下:

(1) 选择 Visual C++ 2010 集成环境窗口中的“文件(F)”→“新建(N)”→“项目(P)”菜单命令, 如图 1-2 所示, 打开“新建项目”对话框。

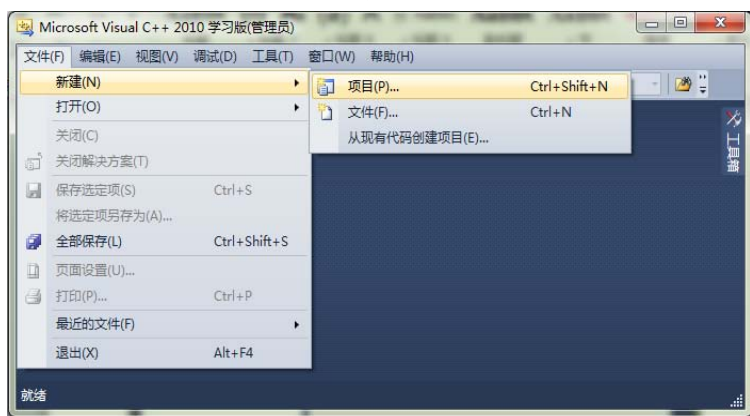


图 1-2 通过菜单命令新建项目

(2) 在如图 1-3 所示的“新建项目”对话框中, 从左侧的“最近的模板”栏目的“已安装的模板”下选择 Visual C++, 在展开的节点中选择 Win32 选项, 然后在对话框中间的栏目中选择“Win32 控制台应用程序”, 并在对话框下方的“名称(N)”文本框中对项目进行命名(如 exampl1), 而在“位置(L)”下拉列表中指定项目的存放位置(也可采用默认路径), 如图 1-3 所示。在输入项目名称后, 解决方案名称默认为项目名称(也可自定义解决方案名称), 如果选中“为解决方案创建目录”复选框, 系统将为整个项目创建一个与解决方案名称同名的目录。单击“确定”按钮, 将弹出如图 1-4 所示的应用程序向导。

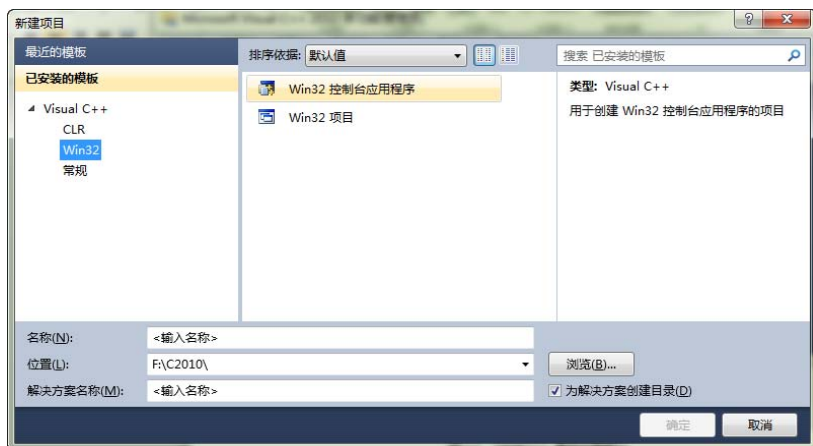


图 1-3 创建并设置项目

(3) 在图 1-4 所示的界面中单击“下一步”按钮，将会弹出如图 1-5 所示的应用程序设置界面。首先在“应用程序类型”区域选中“控制台应用程序”单选按钮，然后在“附加选项”区域选中“空项目”复选框，单击“完成”按钮。



图 1-4 应用程序向导



图 1-5 设置应用程序

(4) 回到 Visual C++ 2010 集成环境窗口，弹出“解决方案资源管理器”窗格，如图 1-6 所示。

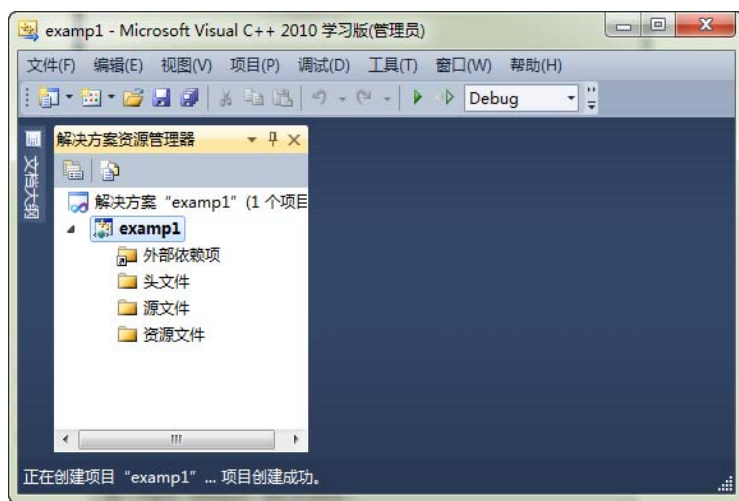


图 1-6 “解决方案资源管理器” 窗格

3. 建立 C 源文件

在“解决方案资源管理器”窗格中右击“源文件”图标，从弹出的菜单中选择“添加(D)”→“新建项(W)”命令，如图 1-7 所示。

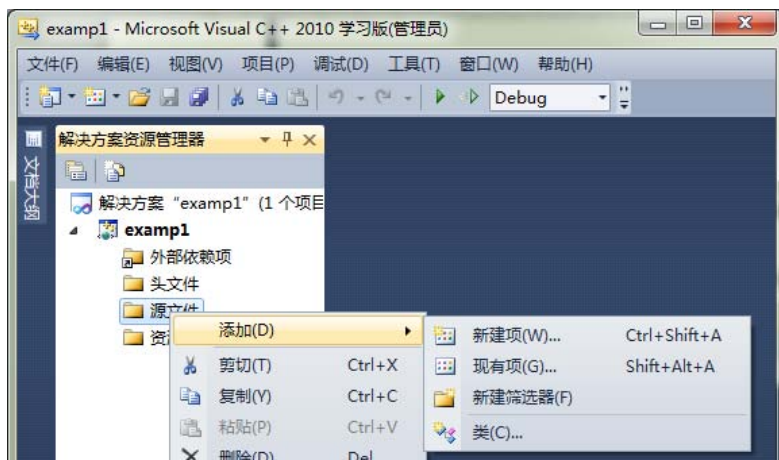


图 1-7 建立 C 源文件

在弹出的“添加新项”对话框中，在“已安装的模板”下展开 Visual C++ 节点，选择“代码”。在对话框的中间栏目中选“C++ 文件(.cpp)”，在下方的“名称(N)”文本框中输入源文件的名称，在“位置(L)”下拉列表中输入源文件的位置(也可使用默认位置，默认位置为前面新建的解决方案文件夹)，如图 1-8 所示。单击“确定”按钮，重新回到 Visual C++ 2010 集成环境窗口，窗口的右侧出现了编辑窗体，如图 1-9 所示。

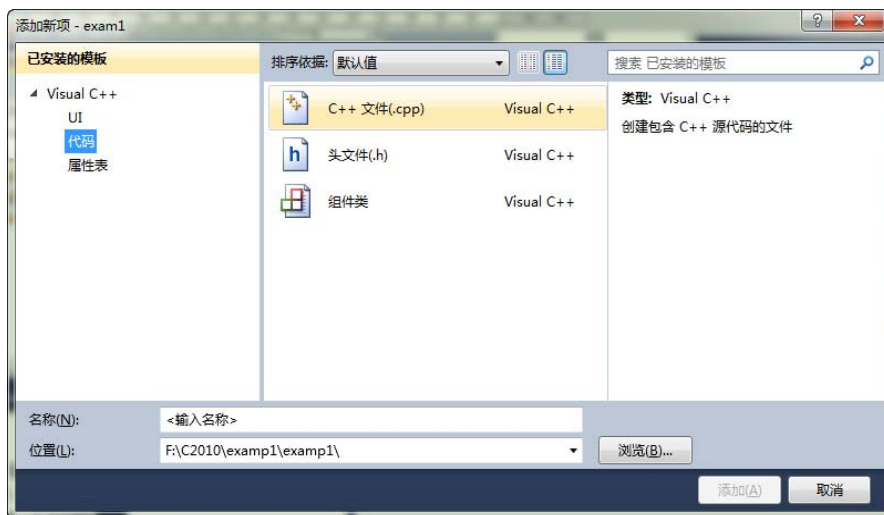


图 1-8 “添加新项”对话框

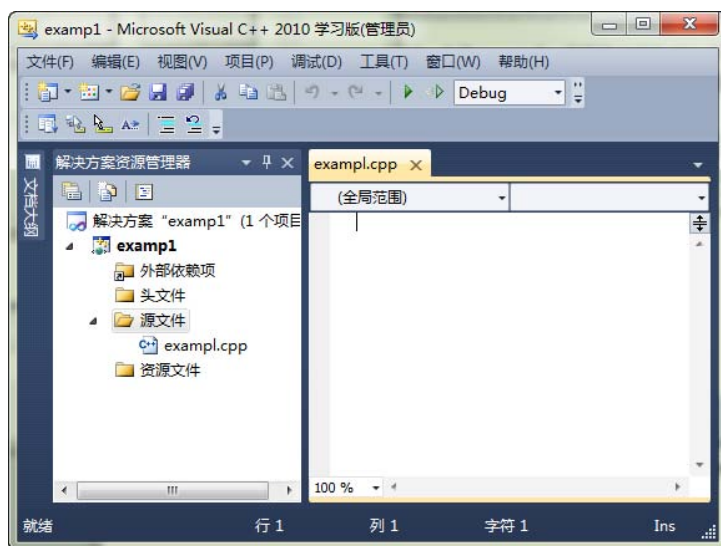


图 1-9 出现的编辑窗体

4. 编辑 C 源文件

在编辑窗体中输入 C 源文件的内容，例如，可输入如下代码：

```
#include<stdio.h>
void main()
{
    int x,y,sum;
    printf("Input a integer:");
    scanf("%d",&x);
```

```

printf("Input another integer:");
scanf("%d",&y);
sum=x+y;
printf("sum=%d\n",sum);
}

```

结果如图 1-10 所示。

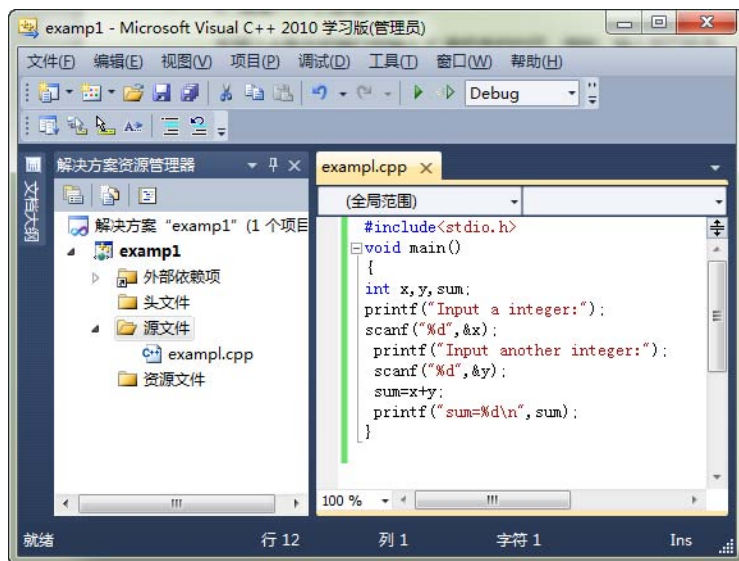


图 1-10 编辑 C 源文件

5. 编译 C 源文件

在 Visual C++ 2010 集成环境窗口中选择“调试(D)”→“启动调试(S)”菜单命令，弹出如图 1-11 所示的提示对话框。单击“是(Y)”按钮，编译 exampl.cpp 源文件。

编译结果如图 1-12 所示。如果没有错误，系统将显示提示信息，例如：

```

===== 生成: 成功1 个, 失败0 个, 最新 0 个, 跳过 0 个 =====

```

这表示没有任何错误。另外，有时虽然会出现一些警告信息，但却不影响程序的执行。如果程序有错误，那么编译结果如图 1-13 所示，系统给出的错误信息的格式如下：

```

<源程序路径>(行)<错误代码>: <错误内容>

```

系统会根据错误的不同显示不同的错误信息。可根据编译结果中给出的错误信息，进入编辑查错状态，到 C 源程序中进行修改。



图 1-11 提示对话框

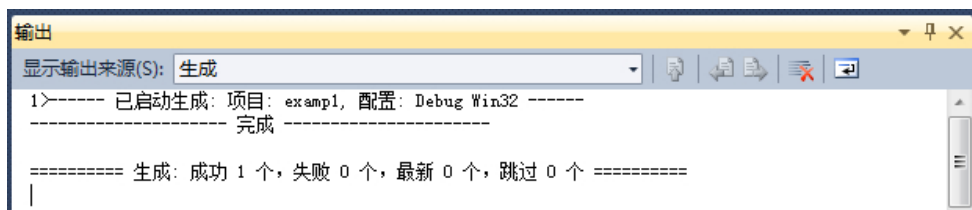


图 1-12 编译成功时的提示信息

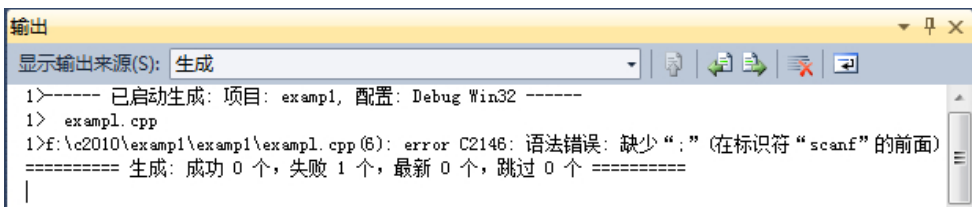


图 1-13 编译失败时给出的出错信息

从图 1-13 可以看出, 程序中有 1 个错误(error), 系统给出的错误信息如下:

```
f:\c2010\examp1\examp1\examp1.cpp(6): error C2146: 语法错误: 缺少“;” (在标识符“scanf”的前面)
```

以上信息说明在 examp1.cpp 的第 6 行中, 标识符 scanf 前缺少分号。

双击编译结果中的出错信息, 编辑窗体中会指示对应的出错位置, 可根据出错信息对代码进行纠正。例如, 在程序的第 6 行的末尾加上分号, 然后再次编译, 编译结果中就没有错误了。

在检查程序时一定要细心, 可首先查看第一个错误出现的地方及其前面的一小段程序。在查出并改正这个错误之后, 可以看一看其后的几个错误说明中的错误位置是否和第一个错误的位置邻近。如果邻近, 那么有可能反映的还是那个错误, 这时可以再编译一次, 根据经验, 错误的数量将大为减少。重复上述过程, 直到纠正了所有的错误为止。

6. 运行程序

按快捷键 Ctrl+F5, 即可运行程序。

运行程序后, Visual C++ 2010 将自动弹出输入输出窗口, 如图 1-14 所示。输入数据后, 程序将输出结果, 按任意键可关闭输入输出窗口。



图 1-14 输入输出窗口

另外, 生成的可执行文件可以单独运行。根据之前创建项目时所做的设置, 本例的所有信息都位于 F:\c2010\examp1\ 目录下。实际上, 可执行文件 examp1.exe 所在的目录为 F:\c2010\examp1\Debug。打开 Debug 文件夹, 双击 examp1.exe 文件即可运行程序。

关闭 Visual C++ 2010 集成环境窗口, 系统将自动保存各种文件(它们都被保存到与项目名称同名的文件夹中)。

对于程序的编译和运行, Visual C++ 2010 还提供了一组工具按钮, 如图 1-15 所示。

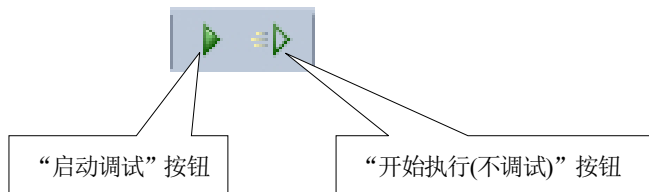


图 1-15 工具按钮

1.2 打开 C 源文件

如果要打开某个已建立的 C 源文件, 可使用以下两种方法。

第一种方法如下: 启动 Visual C++ 2010 后, 选择集成环境窗口中的“文件(F)”→“打开(O)”→“文件(F)”菜单命令, 如图 1-16 所示。



图 1-16 使用菜单命令打开 C 源文件

在“打开项目”对话框中, 可根据自己的需要选择打开项目还是文件, 如图 1-17 所示。如果需要打开的是项目, 那么请找到扩展名为.sln 的文件, 然后单击“解决方案资源管理器”窗

格中的解决方案，在 C 源文件所在的文件夹中双击想要打开的.cpp 源文件，即可对代码进行编辑、调试和运行。

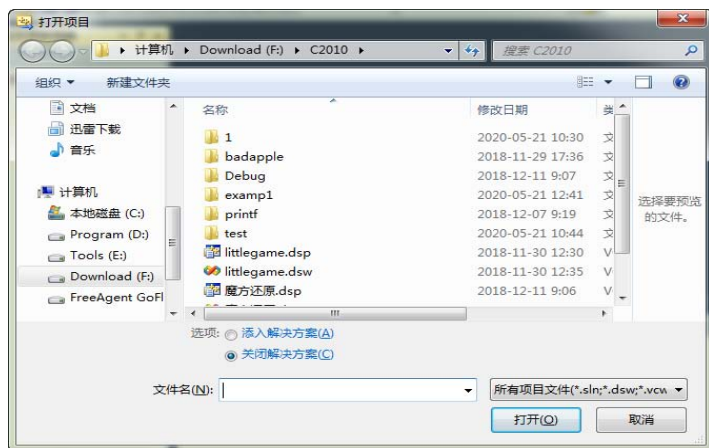


图 1-17 选择想要打开的项目或文件

另一种打开 C 源文件的方法如下：打开项目所在的文件夹(如 examp1 文件夹)，双击项目文件(如 examp1.sln)即可启动 Visual C++ 2010。然后单击“解决方案资源管理器”窗格中的解决方案，在 C 源文件所在的文件夹中双击想要打开的.cpp 源文件，即可对代码进行编辑、调试和运行。

注意：如果单独打开 C 源文件，那么程序不可调试，也不可运行，只能进行编辑。

1.3 调试 C 程序

下面介绍使用 Visual C++ 2010 提供的调试器调试 C 程序的一般方法。假设想要调试的程序如下：

```
/* 求 n!(也就是 1*2*3*...*n), n=6 */
#include<stdio.h>
void main()
{
    int p,i,n;
    p=1;
    n=6;
    for(i=1;i<=n;i++)
        p=p*i;
    printf("n%d!=%d\n",n,p);
}
```

1. 使程序执行到光标所在行时暂停，以便观察中间结果

(1) 在需要暂停的行上单击，定位光标。

(2) 按 Ctrl+F10 快捷键，程序将在执行到光标所在行时暂停，如图 1-18 所示，这么做是为了查看相关的中间结果。在图 1-18 中，“局部变量”窗格中自动显示了相关变量的值：n 和 p

的值分别是 6 和 1，而变量 i 尚未赋值，因而不确定。在编辑窗体中，左侧的箭头指示了当前程序暂停的位置。

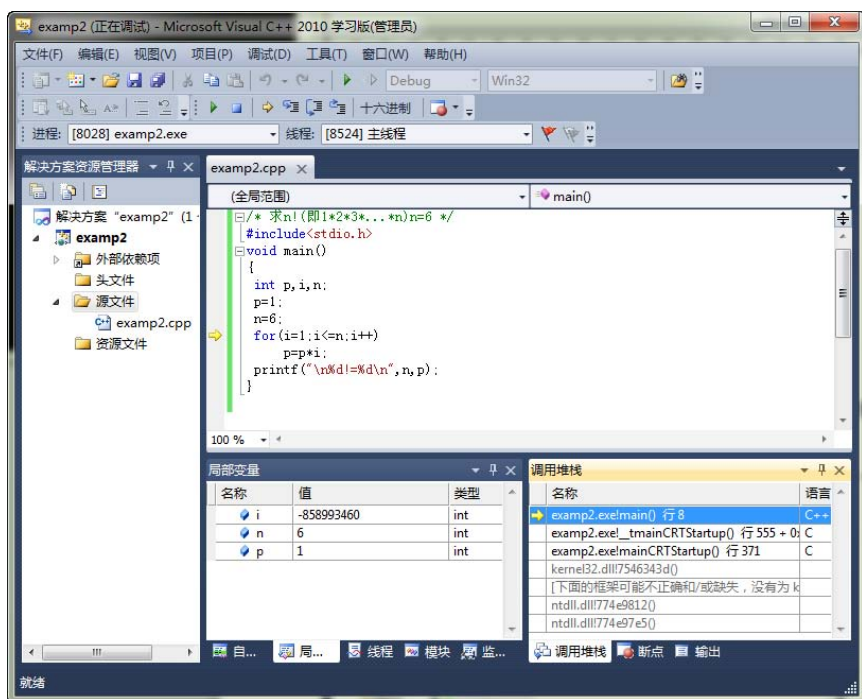


图 1-18 观察变量的值

在编辑窗体中，如果把光标移到后面的某个位置，再按 **Ctrl+F10** 快捷键，程序将从当前的暂停位置继续执行到新的暂停位置，然后再次暂停。

(3) 按 **Shift+F5** 快捷键，程序将停止调试，回到正常的运行状态。

2. 通过设置断点，使程序暂停以便观察中间结果

(1) 在需要设置断点的行的左侧边框上单击或按 **F9** 功能键，断点所在行的左侧将出现一个红色的圆点标志。当光标位于包含断点的行时，按 **F9** 功能键可去掉断点标志。设置了断点之后，可使用前面介绍的方法查看变量的值或结束调试，也可按照下面的步骤进行操作。

(2) 单击“启动调试”按钮(如图 1-15 所示)或按 **F5** 功能键，程序将在执行到第一个断点时暂停，这时在“局部变量”窗格中可以查看有关变量的值。再按 **F5** 功能键，程序继续往下执行，并在执行到第二个断点时暂停，以此类推。

(3) 按 **Shift+F5** 快捷键，程序将停止调试，回到正常的运行状态。

3. 单步执行

在调试过程中，当程序执行到某个位置时，如果发现结果已经不正确了，那么说明在此之前一定有错误存在。如果能够确定某一小段程序有错，我们就可以使用前面介绍的方法将程序暂停在这一小段程序的头一行，输入若干变量，然后单步执行，一次执行一行语句，逐行检查，从而找出错误并进行修改。单步执行时，可单击“调试”工具条中的“逐语句”按钮，如图 1-19

所示,也可按 F10 功能键。当遇到自定义函数调用时,如果想进入函数内部并单步执行,可单击“调试”工具条中的“逐语句”按钮或按 F11 功能键。要想结束函数的单步执行,可单击“调试”工具条中的“跳出”按钮或按 Shift+F11 快捷键。对于不是函数调用的语句来说,F11 功能键和 F10 功能键的作用相同,但一般对于系统函数不要按 F11 功能键。

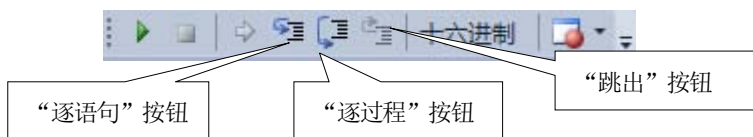


图 1-19 “调试”工具条

上面只对 Visual C++ 2010 中的主要功能进行了介绍,对于其他功能,读者可以自行练习或参考有关 Visual C++ 2010 的使用手册。

第2章 实验内容

本章给出了 11 个实验,每个实验所需学时及其与主教材《计算机基础与 C 语言程序设计(第四版)》(以下简称“主教材”)中各章的对应关系如表 2-1 所示。

表 2-1 每个实验所需学时及其与主教材中各章的对应关系

序 号	题 目	学 时	主 教 材
实验一	熟悉 VC 环境	2	第 2 章
实验二	数据类型	2	第 3 章
实验三	运算符和表达式	2	第 4 章
实验四	选择结构	2	第 5 章
实验五	循环结构	4	第 5 章
实验六	数组	4	第 6 章
实验七	函数	4	第 7 章
实验八	指针	4	第 9 章
实验九	结构体	4	第 10 章
实验十	文件	4	第 11 章
实验十一	综合设计	4	各章

教师可根据实际情况选取实验项目以及其中的内容。

实验一 熟悉 VC 环境

1. 实验目的

- (1) 熟悉 Visual C++ 2010 集成环境。
- (2) 通过编辑和运行给出的程序，掌握如何在计算机上编辑、编译、链接和运行 C 程序。

2. 实验准备

- (1) 阅读主教材第 2 章的主要内容。
- (2) 阅读本书第一篇第 1 章的主要内容。
- (3) 了解实验内容。

3. 实验内容

(1) 程序改错：改正下面程序中的错误，将运行正确的程序存盘，文件名为 ex1_1.c。程序的功能是求两个整数的乘积。程序在运行时会输出提示信息 “Input a, b:”，当输入 3 和 5 时，输出 p=15。

程序如下：

```
#include<stdio.h>
int main()
{
    /*****found*****/
    int a,b;
    printf("Input a,b:");
    /*****found*****/
    scanf("%d%d",&a,&b");
    p=a*b;    /* “*” 为乘法运算符 */
    printf("p=%d\n",p);
    return 0;
}
```

注意：注释部分/*****found*****/表示修改下方代码中的错误。

(2) 程序填空：下面是求两个实数相除的程序，请填空并使程序运行正确，以文件名 ex1_2.c 存盘。当输入 3 和 2 时，输出 c=1.500000。

程序如下：

```
#include<stdio.h>
int main()
{
    float a,b,c;
    scanf( _____ 1 _____ );
```



```

c=a/b;    /* “/” 是除法运算符 */
printf(    2    );
return 0;
}

```

(3) 编写程序：输入立方体的长、宽、高，输出其体积。

(4) 编写程序：输入圆的半径，输出其面积。

实验二 数据类型

1. 实验目的

- (1) 通过实验加深对数据类型的理解，熟悉整型、字符型和实型变量的用法。
- (2) 掌握格式输入输出函数的用法。
- (3) 进一步熟悉 C 程序的编辑、编译、链接和运行过程。

2. 实验准备

- (1) 阅读主教材第 3 章的有关内容。
- (2) 阅读本书第一篇第 1 章的主要内容。
- (3) 了解实验内容。

3. 实验内容

(1) 程序改错：改正下面程序中的错误，使程序运行正确。程序的功能是：输入一个大写字母，输出对应的小写字母。例如，当输入 A 时，输出 a。

程序如下：

```

/*****found*****/
include<stdio.h>
int main()
{
    char c1,c2;
    printf("c1=?\n");
    /*****found*****/
    getchar(c1);
    c2=c1+32;    /* 小写字母的 ASCII 码值比对应的大写字母大 32 */
    printf("%c\n",c2);
    return 0;
}

```

(2) 程序填空：以下程序的功能是，输入一个十进制整数，输出对应的八进制数和十六进制数，请填空并使程序运行正确。例如，当输入 26 时，输出 032 和 0x1a。

程序如下：

```
#include<stdio.h>
int main()
{
    int a;
    printf("Please input a number:");
    scanf("____1____",&a);
    printf("____2____\n",a,a);
    return 0;
}
```

- (3) 编写程序, 输出各基本数据类型的长度(字节数)。
- (4) 编写程序, 已知直角三角形的两条直角边, 求斜边的长度。

实验三 运算符和表达式

1. 实验目的

- (1) 通过实验加深对运算符及表达式的理解。
- (2) 掌握自动类型转换规则和强制类型转换方法。
- (3) 掌握运算符的优先级和结合性。
- (4) 理解表达式的求解过程。

2. 实验准备

- (1) 阅读主教材第4章的有关内容。
- (2) 了解实验内容。

3. 实验内容

(1) 程序改错: 以下程序的功能是, 如果输入大写字母, 则输出对应的小写字母; 如果输入的是其他字符, 则原样输出。改正错误, 使程序运行正确。

程序如下:

```
#include <stdio.h>
int main()
{
    char ch1,ch2;
    printf("Input a capital letter:");
    ch1=getchar();
    /*****found*****/
    ch2= 'A'<=ch1<='Z'?ch1+32:ch1;
    putchar(ch2);
    putchar("\n");
}
```

```
return 0;
}
```

(2) 程序填空：以下程序的功能是，输入一个两位的整数，按倒序输出。例如，如果输入 56，则输出 65。请填空，使程序运行正确。

程序如下：

```
#include <stdio.h>
int main()
{
    int n,k;
    printf("Input a number:");
    scanf("%d",&n);
    k=_____+n/10;
    printf("%d\n",k);
    return 0;
}
```

(3) 编写程序：输入两个整数，输出它们的和、差、积、商、余数。

(4) 编写程序：输入一个 3 位的整数，求各位上的数字之和。例如，如果输入 312，则输出 6。

实验四 选择结构

1. 实验目的

- (1) 通过实验加深对结构化程序设计的理解。
- (2) 掌握 if 和 if-else 语句的使用方法。
- (3) 掌握 switch 语句的使用方法。

2. 实验准备

- (1) 阅读主教材第 5 章的有关内容。
- (2) 了解实验内容。

3. 实验内容

(1) 程序改错：以下程序的功能是，输入学生的成绩，将它们转换成 Good、Pass、Fail 并输出。转换规则是：80~100 为 Good，60~79 为 Pass，0~59 为 Fail。改正错误，直到程序运行正确为止。

程序如下：

```
#include<stdio.h>
int main()
{ float score;
```

```
printf("Please input score:\n");
scanf("%f",&score);
printf("score=%f\n",score);
/*****found*****/
switch(score/10)
{ case 0:
  case 1:
  case 2:
  case 3:
  case 4:
  /***** found *****/
  case 5: printf("Fail\n");
  case 6:
  /***** found *****/
  case 7: printf("Pass\n");
  case 8:
  case 9:
  case 10: printf("Good\n");
}
return 0;
}
```

(2) 程序填空：输入 x 的值，按下式计算 y 的值并输出结果。

$$y = \begin{cases} -1 & x < 0 \\ 0 & x = 0 \\ 1 & x > 0 \end{cases}$$

程序如下：

```
#include<stdio.h>
int main()
{ double x;
  int y;
  scanf("____1____", &x);
  if( ____2____ ) y=-1;
  else if( ____3____ ) y=0;
  else y=1;
  printf("y=%d\n",y);
  return 0;
}
```

(3) 程序设计：使用 if 语句编写程序，输入 x 的值，按下式计算 y 的值并输出。

$$y = \begin{cases} x^2 + 10 & 0 \leq x \leq 8 \\ x^3 - 10 & x < 0 \text{ 或 } x > 8 \end{cases}$$

(4) 程序设计：输入一个 3 位的正整数，找出使用其中各位上的数字所能组成的最大数和最小数。例如，如果输入 528，则最大数为 852、最小数为 258。

实验五 循环结构

1. 实验目的

- (1) 通过实验加深对结构化程序设计的理解。
- (2) 掌握 while、for 和 do-while 语句的使用方法。
- (3) 学会使用顺序结构、选择结构和循环结构编写一般程序。

2. 实验准备

- (1) 阅读主教材第 5 章的有关内容。
- (2) 了解实验内容。

3. 实验内容

(1) 程序改错：以下程序的功能是，求数字 1~100 的和，也就是计算 $1+2+3+\cdots+100$ 。改正错误，使程序运行正确，输出结果为 $\text{sum}=5050$ 。

程序如下：

```
#include<stdio.h>
int main()
{
    /******found******/
    int i,sum;
    i=1;
    while(i<=100)
    /******found******/
    sum=sum+i;
    /******found******/
    i++;
    printf("sum=%d\n",sum);
    return 0;
}
```

(2) 程序改错：输入一个整数，计算它是一个几位数。例如，如果输入 123123，则输出 $n=6$ 。改正错误，直到程序运行正确为止。

程序如下：

```
#include<stdio.h>
int main()
{ long x;
```

```
int n=0;
printf("input x=");
/*****found*****/
scanf("%d",&x);
do
{
    n++;
    x/=10;
/*****found*****/
}while(x!=0);
printf("n=%d\n",n);
}
```

(3) 程序填空：计算数字的阶乘，如果输入 6，则输出 $6!=720$ 。

程序如下：

```
#include<stdio.h>
int main()
{
    int i,n;
    long np;
    printf("Input a number:");
    scanf("%d",&n);

    _____
    for(i=2;i<=n;i++)
    _____
    printf("%d!=%ld\n",n,np);
    return 0;
}
```

(4) 程序填空：求 Fibonacci 数列的前 40 项。Fibonacci 数列如下：

1, 1, 2, 3, 5, 8, 13, 21, ...

Fibonacci 数列的特点是：最前面的两项都为 1，其余每一项都是前两项之和。

程序如下：

```
#include<stdio.h>
int main()
{
    int i;
    long f1, f2;

    _____;
    for(i=1;i<=20;i++)
    {
        printf("%12ld%12ld",f1,f2);
        if(i%2==0)printf("\n");
        f1+=f2;
    }
    _____;
}
```

```

}
return 0;
}

```

- (5) 编写程序, 输入一个正整数 n , 输出 2 与 n 之间的所有素数(要求 n 必须大于 2)。
 (6) 使用循环语句编写程序, 要求输出如下图形:

```

      * * * * *
     * * * * *
    * * * * *
   * * * * *
  * * * * *
 * * * * *
* * * * *

```

实验六 数组

1. 实验目的

- (1) 通过实验掌握数组的使用方法。
 (2) 学会使用数组编写一般程序。

2. 实验准备

- (1) 阅读主教材第 6 章的有关内容。
 (2) 了解实验内容。

3. 实验内容

- (1) 程序改错: 以下程序的功能是, 输入 1, 输出 “1,2,4”。修改错误, 使程序运行正确。
 程序如下:

```

#include<stdio.h>
int main()
/*****found*****/
{ int a(3);
  int i;
  printf("Please input a[0]:");
/*****found*****/
  scanf("%d",&a);
  for(i=1;i<3;i++) a[i]=2*a[i-1];
/*****found*****/
  printf("%d, %d, %d\n",a);
  return 0;
}

```

- (2) 程序改错: 改正下面程序中的错误, 使其完成以下功能——从键盘输入一行字符, 统

计其中有多少个单词，单词之间用空格分开。

程序如下：

```
#include<stdio.h>
int main()
{   char s[80],c1,c2;
    int i=0,num=0;
    printf("Please input a string:\n");
    gets(s);
    /*****found*****/
    while(s[i]!='\n')
    {   c1=s[i];
        if(i==0) c2=' ';
        else c2=s[i-1];
        /*****found*****/
        if(c1!='&& c2!=' ') num++;
        i++;
    }
    printf("There are %d words.\n",num);
    return 0;
}
```

(3) 程序填空：以下程序的功能是，使用选择排序法对 5 个整数按升序进行排序。请填空并使程序运行正确。

程序如下：

```
#include <stdio.h>
#define N 5
int main()
{   int i, j, k, t, a[N];
    printf("Please input array a:\n");
    for(i=0;i<=N-1;i++) scanf("%d",&a[i]);
    for(i=0;i<N-1;i++)
    {   _____ 1 _____
        for(j=i+1; _____ 2 _____ ;j++)
            if(a[j]<a[k]) k=j;
        if( _____ 3 _____ ){t=a[i];a[i]=a[k];a[k]=t;}
    }
    printf("output the sorted array:\n");
    for(i=0;i<=N-1;i++)   printf("%5d",a[i]);
    printf("\n");
    return 0;
}
```



```
}

```

(4) 程序填空：以下程序的功能是输出 26 个大写英文字母。请填空并使程序运行正确。程序如下：

```
#include <stdio.h>
int main (void)
{   char string[27];
    int i;
    for(i=0; i<26; _____ 1 _____)
        string[i] = _____ 2 _____;
    _____ 3 _____;
    printf("the array contains %s\n", _____ 4 _____);
    return 0;
}
```

(5) 编写程序，求一个 3×3 矩阵的主对角线上的元素之和。

(6) 编写程序，输出以下杨辉三角的前 10 行：

```
1
1   1
1   2   1
1   3   3   1
1   4   6   4   1
1   5  10  10   5   1
1   6  15  20  15   6   1
...
```

实验七 函数

1. 实验目的

- (1) 掌握定义函数的方法。
- (2) 掌握调用函数的方法。
- (3) 学会使用函数编写程序。

2. 实验准备

- (1) 阅读主教材第 7 章的有关内容。
- (2) 了解实验内容。

3. 实验内容

- (1) 程序改错：fac 函数的功能是求 n 的阶乘，也就是求 $n!$ 。请改正程序中的错误，使程序

得出正确的运行结果。

程序如下：

```
#include <stdio.h>
int fac(int n)
/****found****/
{   int f;
    while(n>1)
    {   f*=n;
        /****found******/
        n++;
    }
    return f;
}
int main()
{   int n;
    printf("Please input a number:");
    scanf("%d",&n);
    printf("%d!=%d\n",n,fac(n));
    return 0;
}
```

(2) 程序改错：fun 函数的功能是判断 m 是否为素数，若是返回 1，否则返回 0。main 函数的功能是：按每行 5 个输出 1 和 100 之间的全部素数。分析产生错误的原因并改正错误，直到程序的运行结果正确为止。

程序如下：

```
#include <stdio.h>
/****found******/
void fun( int n)
{   int i,k=1;
    if(n<=1) k=0;
    for(i=2;i<n;i++)
    /****found******/
    if(n%i==0) k=0;
    return k;
}
int main()
{   int m,k=0;
    for(m=1;m<100;m++)
    /****found****/
    if(fun(m)==0)
    {   printf("%4d",m);k++;
        if(k%5==0) printf("\n");
    }
```

```

    }
    return 0;
}

```

(3) 程序填空：已知直角三角形的两条直角边 a 和 b ，求斜边 c 。请填空并运行程序。
程序如下：

```

#include <stdio.h>
#include <math.h>
_____1_____ hypot(double a,double b)
{
    double c;
    c=sqrt(a*a+b*b);
    return c;
}
int main()
{
    double a,b;
    printf("Please input two numbers:");
    scanf("_____2_____",&a,&b);
    printf("%lf\n",hypot(a,b));
    return 0;
}

```

(4) 程序填空：通过函数的递归调用计算数字的阶乘。请填空并运行程序。
程序如下：

```

#include <stdio.h>
long power(int n)
{
    long f;
    if(n>1)_____1_____;
    else f=1;
    return(f);
}
int main()
{
    int n;
    long y;
    printf("input a integer number:\n");
    scanf("%d",&n);
    y= _____2_____;
    printf("%d!=%ld\n",n,y);
    return 0;
}

```

(5) 程序设计：编写 fun 函数，计算如下多项式。

$$s = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots + \frac{x^n}{n!}$$

请在 main 函数中调用 fun 函数。

(6) 程序设计: 编写 fun 函数, 求 n 以内(不包括 n)能同时被 3 与 7 整除的所有自然数之和的平方根 s 并作为函数值返回。请在 main 函数中调用 fun 函数。

实验八 指针

1. 实验目的

- (1) 通过实验进一步掌握指针的概念和使用方法。
- (2) 学会正确使用数组的指针、字符串的指针和函数的指针。
- (3) 学会使用指针编写一般程序。

2. 实验准备

- (1) 阅读主教材第 9 章的有关内容。
- (2) 了解实验内容。

3. 实验内容

(1) 程序改错: 在下面的程序中, fun 函数的功能是实现两个整数之间的交换。例如, 给 a 和 b 分别输入 44 和 56, 输出 “a=56 b=44”。

请改正程序中的错误, 使程序能得出正确的运行结果。

程序如下:

```
#include <stdio.h>
/*****found*****/
void fun(int a,int b)
{
    int t;
    /*****found*****/
    t=b;
    b=a;
    a=t;
}
int main()
{
    int a, b;
    printf("Enter a,b: ");
    scanf("%d%d",&a,&b);
    fun(&a,&b);
    printf("a=%d b=%d\n",a,b);
    return 0;
```

```
}
```

(2) 程序改错: 在下面的程序中, fun 函数的功能是从 n 名学生的成绩中统计出低于平均分的学生人数并作为函数值返回, 平均分存放在形参 aver 指向的存储单元中。例如, 如果输入的 5 名学生的成绩分别为 80、75.6、77、96 和 65.7, 则低于平均分的学生人数为 3(平均分为 78.86)。请改正程序中的错误, 使程序能得出正确的运行结果。

程序如下:

```
#include <stdio.h>
int fun(float *s,int n,float *aver)
{
    float ave, t=0.0;
    int count=0,k,i;
    /******found*****/
    for(k=0;k<n;k++) t=s[k];
    ave=t/n;
    for(i=0;i<n;i++)
        if(s[i]<ave)
            count++;
    /******found*****/
    *aver=&ave;
    return count;
}
int main()
{
    float s[30], aver;
    int m, i;
    printf("Please enter m: ");
    scanf("%d", &m);
    printf("Please enter %d mark:\n", m);
    for(i=0;i<m;i++)
        scanf("%f", s+i);
    printf("The number of students:%d\n", fun(s,m,&aver));
    printf("Ave=%.2f\n", aver);
    return 0;
}
```

(3) 程序填空: 请补充 fun 函数, fun 函数的功能是返回字符数组中指定字符的个数, 想要指定的字符可从键盘输入。例如, 输入字符串 abcd aaf 和字符 a, 程序将输出 3。

程序如下:

```
#include <stdio.h>
#define N 80
int fun(char s[],char ch)
{
    int i=0, n=0;
    while(____1____)
```

```

        { if(____ 2 ____ ) n++;
          i++;
        }
        ____ 3 ____;
    }
int main()
{ int n;
  char str[N], ch;
  printf("Input a string:\n");
  gets(str);
  printf("Input a charactor:\n");
  scanf("%c",&ch);
  n=fun(str,ch);
  printf("number of %c: %d",ch,n);
  return 0;
}

```

(4) 程序填空：数组 str 全由大小写字母组成。请补充 fun 函数，fun 函数的功能是：把数组 str 中的字母转换成紧接着的下一个字母。例如，如果原来的字母为'z'或'Z'，就相应地转换成'a'或'A'。转换结果仍保存在原来的数组中。

程序如下：

```

#include <stdio.h>
#define N 80
void fun(char s[])
{ int i;
  for(i=0;____ 1 ____;i++)
  { if(s[i]=='z'||s[i]=='Z')
    s[i]=____ 2 ____;
    else
    s[i]+=____ 3 ____;
  }
}
int main()
{ char str[N];
  printf("Input a string:\n");
  gets(str);
  printf("original string :\n");
  puts(str);
  fun(str);
  printf("new string:\n");
  puts(str);
  return 0;
}

```

```
}
```

(5) 编写一个函数，功能是计算 n 门功课的平均分，将计算结果作为函数值返回。

(6) 编写一个函数，功能是删除一维数组中所有相同的数，仅保留其中一个。假设一维数组中的数已按从小到大的顺序排好，请将执行删除操作后的数组元素的个数作为函数值返回。

实验九 结构体

1. 实验目的

- (1) 掌握结构体变量的定义和使用方法。
- (2) 掌握结构体数组的使用方法。
- (3) 学会使用结构体类型编写一般程序。

2. 实验准备

- (1) 阅读主教材第 10 章的有关内容。
- (2) 了解实验内容。

3. 实验内容

(1) 程序改错：输入 5 门功课的成绩，计算平均成绩并输出。请修改程序中的错误并进行调试。程序如下：

```
#include <stdio.h>
int main()
{
    struct student
    {
        double score[5], ave;
        /******found******/
    } stu;
    int i;
    stu.ave=0;
    for(i=0;i<5;i++)
        /******found******/
    {
        scanf("%f",&stu.score[i]);
        stu.ave+=stu.score[i]/5.0;
    }
    for(i=0;i<5;i++)
        printf("%6.1f", stu.score[i]);
    printf(" average=%6.1f\n",stu.ave);
    return 0;
}
```

(2) 程序改错：输入 3 个复数的实部和虚部，将它们存放到一个结构体数组中，根据复数

的模,按照由大到小的顺序对这个结构体数组中的元素进行排序并输出。请改正程序中的错误,使程序能得出正确的运行结果。

程序如下:

```
#include <stdio.h>
#include <math.h>
int main()
/*****found*****/
{
    complex
    { double x, y, m; } a[3], temp;
    int i,j,k;
    for(i=0;i<3;i++)
    {
        scanf("%lf%lf",&a[i].x,&a[i].y);
        /*****found*****/
        a[i].m=sqrt(a[i].x*a[i].x+a[i].y*a[i].y);
    }
    for(i=0;i<2;i++)
    {
        k=i;
        for(j=i+1;j<3;j++)
            if(a[k].m<a[j].m) k=j;
        temp=a[i]; a[i]=a[k]; a[k]=temp;
    }
    for(i=0;i<3;i++) printf("%.2f%+.2fi\n",a[i].x,a[i].y);
    return 0;
}
```

(3) 程序填空: 以下程序的功能是读入一行字符,并按输入时的逆序建立一条链表——先输入的字符位于链表的末尾,然后按照与输入时相反的顺序输出这些字符,并释放全部节点。请填空并运行程序。

程序如下:

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    struct node
    {
        char info;
        struct node *link;
    } *top, *p;
    char c;
    top=NULL;
    while((c=____1____) != '\n' )
    {
        p=(struct node *)malloc(sizeof(struct node));
        p->info=c;
        p->link=top;
    }
}
```



```

    _____ 2 _____;
}
while( top )
{ p=top;
  _____ 3 _____;
  putchar(p->info);
  free(p);
}
return 0;
}

```

(4) 程序填空: 在下面的程序中, fun 函数的功能是对 N 名学生的学习成绩, 按从高到低的顺序找出前 $m(m \leq 10)$ 名学生, 并将这些学生的信息存放到一块动态分配的连续存储区域, 将这块存储区域的首地址作为函数值返回。请填空并运行程序。

程序如下:

```

#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#define N 10
typedef struct ss
{ char num[10];
  int s;
}STU;
STU *fun(STU a[], int m)
{ STU b[N], *t;
  int i,j,k;
  t= _____ 1 _____
  for(i=0; i<N;i++) b[i]=a[i];
  for(k=0; k<m;k++)
  { for(i=j=0; i<N; i++)
    { if(b[i].s>b[j].s)j=i;
      _____ 2 _____
    }
    t[k].s=b[j].s;
    b[j].s=0;
  }
  return t;
}
int main()
{ STU a[N]={ {"A01",81}, {"A02",89}, {"A03",66}, {"A04",87}, {"A05",77},
             {"A06",90}, {"A07",79}, {"A08",61}, {"A09",80}, {"A10",71}};
  STU *porder;
  int i, m;

```

```

printf("\nGive the number of the students who have better score: ");
scanf("%d",&m);
while(m>10)
{ printf("\nGinve the number of the students who have better score: ");
  scanf("%d",&m);
}
porder=fun(a,m);
printf("The top :n");
for(i=0; i<m; i++)
    printf(" %s  %d\n",porder[i].num,porder[i].s);
free(porder);
return 0;
}

```

(5) 编写程序：统计候选人的得票信息。假设有 3 名候选人，每次输入一个得票的候选人的姓名，要求最后输出所有候选人的得票信息。

(6) 编写程序：已知学生的记录由学号和学习成绩构成， N 名学生的记录已在 `main` 函数中被保存到一个结构体数组中。请编写一个函数，该函数的功能是找出成绩最低的那名学生的记录，然后通过形参返回给 `main` 函数(规定只有一个最低分)。

实验十 文件

1. 实验目的

- (1) 通过实验进一步理解文本文件和二进制文件的概念及特点。
- (2) 掌握文件的建立、读写和关闭方法。
- (3) 熟悉顺序存取文件的方法。
- (4) 了解随机存取文件的方法。

2. 实验准备

- (1) 阅读主教材第 11 章的有关内容。
- (2) 了解实验内容。

3. 实验内容

(1) 程序改错：从键盘输入一个字符串，将其中的小写字母全部转换成大写字母，然后输出到磁盘文件 `test` 中并进行保存。输入的字符串以!结束。请修改程序中的错误并进行调试。

程序如下：

```

#include "stdio.h"
#include "string.h"
int main()

```

```

{ FILE *fp;
  char str[100];
  int i=0;
  /******found*****/
  fp=fopen("test.txt","r");
  printf("please input a string:\n");
  /******found*****/
  gets(&str);
  while(str[i]!='\n')
  {   if(str[i]>='a'&&str[i]<='z')
      str[i]=str[i]-32;
      fputc(str[i],fp);
      i++;
  }
  /******found*****/
  fclose(*fp);
  fp=fopen("test.txt","r");
  fgets(str,strlen(str)+1,fp);
  printf("%s\n",str);
  fclose(fp);
  return 0;
}

```

(2) 程序改错：将若干学生的档案存放到一个文件中，并显示其中的内容。请修改程序中的错误并进行调试。

程序如下：

```

#include <stdio.h>
struct student
{   int num;
    char name[10];
    int age;
};
struct student stu[3]={ {101,"Li Mei",18},{102,"Ji Hua",19},{103,"Sun Hao",18}};
int main()
{   struct student *p;
    /******found*****/
    file fp;
    int i;
    fp=fopen("stu_list","wb");
    /******found*****/
    for(*p=stu;p<stu+3;p++)

```

```

fwrite(p,sizeof(struct student),1,fp);
fclose(fp);
fp=fopen("stu_list","rb");
printf(" No.   Name       age\n");
for(i=1;i<=3;i++)
    { fread(p,sizeof(struct student),1,fp);
      printf("%4d %-10s %4d\n",p->num,p->name,(*p).age);
    }
fclose(fp);
return 0;
}

```

(3) 程序填空：以下程序与文件操作有关，请填空并运行程序。
程序如下：

```

#include <stdio.h>
#include <stdlib.h>
int main()
{ /* 定义文件指针 fp */
   _____ 1 _____ *fp;
   char   filename[10];
   printf("Please input the name of file: ");
   scanf("%s", filename); /* 输入字符串并赋给变量 filename */
   /* 以读的方式打开文件 */
   if((fp=fopen(filename, "r")) == NULL)
   {   printf("Cannot open the file.\n");
       _____ 2 _____; /* 正常跳出程序 */
   }
   /* 关闭文件 */
   _____ 3 _____;
   return 0;
}

```

(4) 程序填空：以下程序的功能是从键盘输入一个字符串，然后对这个字符串进行升序排列并输出到文件 test.txt 中，最后从 test.txt 文件中读出这个字符串并显示出来。请填空并运行程序。
程序如下：

```

#include<stdio.h>
#include<string.h>
#include <stdlib.h>
int main()
{ FILE *fp;
  char t,str[100];
  int n,i,j;

```

```

if((fp=fopen("test.txt","w"))==NULL)
    { printf("can't open this file.\n");exit(0);}
printf("input a string:\n");
gets(str);
    _____ 1 _____;
for(i=0; i<n ;i++)
    for(j=0; _____ 2 _____;j++)
        if(str[j]>str[j+1]) { t=str[j];str[j]=str[j+1];str[j+1]=t; }
    _____ 3 _____;
fclose(fp);
fp=fopen("test.txt","r");
fgets(str,100,fp);
printf("%s\n",str);
fclose(fp);
return 0;
}

```

(5) 编写程序：求 Fibonacci 数列的前 40 项并将它们保存到一个文件中，然后从这个文件中读出数据并显示在屏幕上。Fibonacci 数列如下：

1, 1, 2, 3, 5, 8, 13, 21, ...

Fibonacci 数列的前两项均为 1，之后的每一项都是前两项之和。

(6) 编写程序：有 5 名学生，每名学生的数据包括学号、姓名、成绩，从键盘输入这 5 名学生的数据，将它们存放到文件 stud.dat 中，然后将 stud.dat 文件中的数据读出并计算平均成绩，最后将数据显示在屏幕上。

实验十一 综合设计

1. 实验目的

- (1) 通过实验了解较大程序的设计方法。
- (2) 掌握结构化程序设计方法。

2. 实验准备

- (1) 复习主教材各章的主要内容。
- (2) 了解实验内容。

3. 实验内容

编写学生成绩管理程序。学生信息包括：学号、姓名、 M 门功课的成绩、总成绩、平均成绩。一名学生的信息称为一条记录，要求实现以下功能。

- (1) 输入记录。
- (2) 显示记录。
- (3) 排序记录。
- (4) 查询记录。

第3章 实验参考答案

实验一 熟悉 VC 环境

(1) 程序改错

第1处: 将 `int a,b;` 改为 `int a,b,p;`

第2处: 将 `scanf("%d%d",&a,&b);` 改为 `scanf("%d%d",&a,&b);`

(2) 程序填空

第1处填空: `"%f%f",&a,&b`

第2处填空: `"c=%f\n",c`

(3) 编写程序

计算立方体体积的程序如下:

```
#include<stdio.h>
int main()
{
    float a,b,c,v;
    printf("a,b,c=?\n");
    scanf("%f%f%f",&a,&b,&c);
    v=a*b*c;;
    printf("v=%f\n",v);
    return 0;
}
```

程序运行结果如下:

```
a,b,c=?
2 3 4✓
v=24.000000
```

(4) 编写程序

计算圆面积的程序如下:

```
#include<stdio.h>
int main()
```

```
{
    float r,s;
    printf("r=?");
    scanf("%f",&r);
    s=3.141592*r*r;
    printf("s=%f\n",s);
    return 0;
}
```

程序运行结果如下:

```
r=?1✓
s=3.141592
```

实验二 数据类型

(1) 程序改错

第 1 处: 开头加#

第 2 处: 将 `getchar(c1);`改为 `c1=getchar();`

(2) 程序填空

第 1 处填空: `%d`

第 2 处填空: `%#o,%#x`

(3) 编写程序

程序如下:

```
#include<stdio.h>
int main()
{
    printf("char:%d\n",sizeof(char));
    printf("short:%d\n",sizeof(short));
    printf("int:%d\n",sizeof(int));
    printf("long:%d\n",sizeof(long));
    printf("float:%d\n",sizeof(float));
    printf("double:%d\n",sizeof(double));
    return 0;
}
```

程序运行结果如下:

```
char:1
short:2
int:4
long:4
float:4
```

double:8

(4) 编写程序

程序如下:

```
#include<stdio.h>
#include<math.h>
int main()
{
    float a,b,c;
    scanf("%f%f",&a,&b);
    c=sqrt(a*a+b*b);
    printf("\n%f\n",c);
    return 0;
}
```

程序运行结果如下:

```
3 4✓
5.00000
```

实验三 运算符和表达式

(1) 程序改错

将 `ch2='A'<=ch1<='Z'` 改为 `ch2=ch1>='A' && ch1<='Z'`

(2) 程序填空

`n%10*10`

(3) 编写程序

程序如下:

```
#include<stdio.h>
int main()
{
    int a,b;
    printf("Input two numbers:");
    scanf("%d%d",&a,&b);
    printf("%d+%d=%d\n",a,b,a+b);
    printf("%d-%d=%d\n",a,b,a-b);
    printf("%d*d=%d\n",a,b,a*b);
    printf("%d/%d=%d\n",a,b,a/b);
    printf("%d%%d=%d\n",a,b,a%b);
    return 0;
}
```


程序运行结果如下:

```
Input two numbers:12 10✓
12+10=22
12-10=2
12*10=120
12/10=1
12%10=2
```

(4) 编写程序

程序如下:

```
#include<stdio.h>
int main()
{   int a,b;
    printf("a=?");
    scanf("%d",&a);
    b=a/100+a%100/10+a%10;
    printf("b=%u\n",b);
    return 0;
}
```

程序运行结果如下:

```
a=?123✓
b=6
```

实验四 选择结构

(1) 程序改错

第 1 处: 将 `score/10` 改为 `(int)score/10`

第 2 处: 加 `break`;

第 3 处: 加 `break`;

(2) 程序填空

第 1 处填空: `%lf`

第 2 处填空: `x<0`

第 3 处填空: `x==0`

(3) 编写程序

程序如下:

```
#include<stdio.h>
int main()
{   float x,y;
    printf("Input x:");
```

```
scanf( "%f", &x);
if(x>=0&&x<=8)y=x*x+10;
else y=x*x*x-10;
printf("y=%f\n",y);
return 0;
}
```

程序运行结果如下:

```
Input x:4✓
y=26.000000
Input x:-3✓
y=-37.000000
```

(4) 编写程序

程序如下:

```
#include<stdio.h>
int main()
{   int x,max,mid,min,maxnum,minnum,t;
    printf("Input a number:");
    scanf( "%d", &x);
    max=x/100;
    mid=x%100/10;
    min=x%10;
    if(max<mid){t=max;max=mid;mid=t;}
    if(max<min){t=max;max=min;min=t;}
    if(mid<min){t=mid;mid=min;min=t;}
    maxnum=max*100+mid*10+min;
    minnum=min*100+mid*10+max;
    printf("maxnum=%d\nminnum=%d\n",maxnum,minnum);
    return 0;
}
```

程序运行结果如下:

```
Input a number:528✓
maxnum=852
minnum=258
```

实验五 循环结构

(1) 程序改错

第1处: 将 sum 改为 sum=0

第 2 处: 将 `sum=sum+i;` 改为 `{ sum=sum+i;`

第 3 处: 将 `i++;` 改为 `i++;}`

(2) 程序改错

第 1 处: 将 `%d` 改为 `%ld`

第 2 处: 将 `x=0` 改为 `x!=0`

(3) 程序填空

第 1 处填空: `np=1;`

第 2 处填空: `np=np*i;`

(4) 程序填空

第 1 处填空: `f1=f2=1` 或 `f1=1;f2=1`

第 2 处填空: `f2+=f1` 或 `f2=f2+f1`

(5) 编写程序

程序如下:

```
#include <stdio.h>
#include <math.h>
int main()
{   int N,n,i,k,flag,count=0;
    printf("input N:");
    scanf("%d",&N);
    for(n=2;n<=N;n++)
    {   flag=1;
        k=sqrt(n);
        for(i=2;i<=k;i++)
            if(n%i==0) {flag=0; break;}
        if(flag==1)
        {   printf("%5d",n); count++;
            if(count%10==0) printf("\n");
        }
    }
    printf("\n");
    return 0;
}
```

程序运行结果如下:

```
input N:20✓
      2   3   5   7  11  13  17  19
```

(6) 编写程序

程序如下:

```
#include <stdio.h>
int main()
{   int n,i,j;
    for(i=1;i<=6;i++)
    {   for(j=1;j<=6-i;j++) putchar(' ');
        for(j=1;j<=6;j++) putchar('*');
        putchar('\n');
    }
    return 0;
}
```

程序运行结果如下:

```

          * * * * *
         * * * * *
        * * * * *
       * * * * *
      * * * * *
     * * * * *
    * * * * *
```

实验六 数组

(1) 程序改错

第1处: 将 a(3)改为 a[3]

第2处: 将 &a 改为 &a[0]或 a

第3处: 将 a 改为 a[0],a[1],a[2]

(2) 程序改错

第1处: 将 '\n'改为 '\0'

第2处: 将 c2!=''改为 c2==''

(3) 程序填空

第1处填空: k=i;

第2处填空: j<=N-1

第3处填空: i!=k

(4) 程序填空

第1处填空: i++

第2处填空: 'A'+i

第3处填空: string[i]='\0'

第4处填空: string

(5) 编写程序

程序如下:

```
#include <stdio.h>
int main()
{
    int i,j,a[3][3],sum=0;
    printf("Please input 3×3 array:\n");
    for(i=0;i<3;i++)
        for(j=0;j<3;j++)
            scanf("%d",&a[i][j]);
    for(i=0;i<3;i++) sum=sum+a[i][i];
    printf("sum=%d\n",sum);
    return 0;
}
```

程序运行结果如下:

```
Please input 3×3 array:
1  2  3✓
4  5  6✓
7  8  9✓
sum=15
```

(6) 编写程序

程序如下:

```
#include <stdio.h>
int main()
{
    int i,j,a[10][10]={0};
    for(i=0;i<10;i++)
    {
        a[i][0]=1;
        a[i][i]=1;
    }
    for(i=2;i<10;i++)
        for(j=1;j<i;j++)
            a[i][j]=a[i-1][j-1]+a[i-1][j];
    for(i=0;i<10;i++)
    {
        for(j=0;j<=i;j++)
            printf("%4d",a[i][j]);
        printf("\n");
    }
    return 0;
}
```

程序运行结果如下:

```
1
1  1
```

```

1  2  1
1  3  3  1
1  4  6  4  1
1  5 10 10 5  1
1  6 15 20 15 6  1
1  7 21 35 35 21 7  1
1  8 28 56 70 56 28 8  1
1  9 36 84 126 126 84 36 9  1
    
```

实验七 函数

(1) 程序改错

第1处: 将 `int f` 改为 `int f=1`

第2处: 将 `n++` 改为 `n--`

(2) 程序改错

第1处: 将 `void` 改为 `int`

第2处: 将 `n%i=0` 改为 `n%i==0`

第3处: 将 `fun(m)==0` 改为 `fun(m)==1` 或 `fun(m)`

(3) 程序填空

第1处填空: `double`

第2处填空: `%lf%lf`

(4) 程序填空

第1处填空: `f=n*power(n-1)`

第2处填空: `power(n)`

(5) 编写程序

程序如下:

```

#include <stdio.h>
double fun(double x, int n)
{
    double s=1.0,p=1.0;
    int i,j,t=1;
    for(i=1; i<=n; i++)
    {
        t=t*i;
        p=p*x;
        s=s+p/t;
    }
    return s;
}
int main()
{
    printf("%f\n", fun(0.3,10));
}
    
```

```
return 0;
}
```

程序运行结果如下:

```
1.349859
```

(6) 编写程序

程序如下:

```
#include <math.h>
#include <stdio.h>
double fun( int n)
{ double s=0.0;
  int i;
  for(i=1; i<n; i++)
    if(i%3==0&& i%7==0) s=s+i;
    s=sqrt(s);
  return s;
}
main()
{ printf("s=%f\n",fun(1000));
  return 0;
}
```

程序运行结果如下:

```
s=153.909064
```

实验八 指针

(1) 程序改错

第 1 处: void fun(int a,int b)应改为 void fun(int*a,int*b)

第 2 处: t=b;b=a;a=t;应改为 t=*b;*b=*a;*a=t;

(2) 程序改错

第 1 处: t=s[k];应改为 t+=s[k];

第 2 处: *aver=&ave;应改为*aver=ave;

(3) 程序填空

第 1 处填空: s[i]

第 2 处填空: s[i]==ch

第 3 处填空: return n

(4) 程序填空

第 1 处填空: s[i]!='\0'

第2处填空: 25

第3处填空: 1

(5) 编写程序

程序如下:

```
#include <stdio.h>
float fun (float *a, int n)
{
    float ave=0.0;
    int i;
    for(i=0; i<n; i++) ave+=a[i];
    ave/=n;
    return ave;
}
int main()
{
    float score[30]={90.5,72,80,61.5,55}, aver;
    aver=fun(score,5);
    printf("Average score is:%5.2f\n",aver);
    return 0;
}
```

程序运行结果如下:

```
Average score is:71.80
```

(6) 编写程序

程序如下:

```
#include <stdio.h>
#define N 80
int fun(int a[],int n)
{
    int i,t,j=0,*p=a;
    t=p[0];
    for(i=1; i<n; i++)
        if(t!=p[i]) {a[j]=t; t=p[i]; j++; }
    if(i>=n) a[j]=t;
    return j+1;
}
int main()
{
    int a[N]={2,2,2,3,4,4,5,6,6,6,6,7,7,8,9,9,10,10,10,10}, i,n=20;
    printf("The original data :\n");
    for(i=0; i<n; i++) printf("%3d",a[i]);
    n=fun(a,n);
    printf("\nThe data after deleted :\n");
    for(i=0;i<n;i++) printf("%3d",a[i]);
}
```



```
printf("\n ");
return 0;
}
```

程序运行结果如下:

```
The original data :
2 2 2 3 4 4 5 6 6 6 6 7 7 8 9 9 10 10 10 10
The data after deleted :
2 3 4 5 6 7 8 9 10
```

实验九 结构体

(1) 程序改错

第 1 处: 句末加分号;

第 2 处: %f 应改为 %lf

(2) 程序改错

第 1 处: complex 应改为 struct complex

第 2 处: a[j]->m 应改为 a[i].m

(3) 程序填空

第 1 处填空: getchar()

第 2 处填空: top=p

第 3 处填空: top=p->link

(4) 程序填空

第 1 处填空: (STU*)malloc(sizeof(STU)*m);

第 2 处填空: strcpy(t[k].num,b[j].num);

(5) 编写程序

程序如下:

```
#include<stdio.h>
#include<string.h>
#define N 10
struct person
{ char name[20];
  int count;
} leader[3]={"wang",0,"liu",0,"li",0};
int main()
{ int i,j;
  char lname[20];
  printf("Please enter selected person's name:\n");
  for(i=1;i<=N;i++)
```

```

    {   gets(lname);
        for(j=0;j<3;j++)
            if(strcmp(lname,leader[j].name)==0)
                leader[j].count++;
    }
    printf("The result:\n");
    for(i=0;i<3;i++)
        printf("%5s:%d\n",leader[i].name,leader[i].count);
    return 0;
}

```

程序运行结果如下:

```

wang✓
wang✓
li✓
liu✓
li✓
wang✓
liu✓
liu✓
li✓
wang✓
The result:
wang:4
    liu:3
    li:3

```

(6) 编写程序

程序如下:

```

#include <stdio.h>
#include <string.h>
#define N 10
typedef struct ss
{   char num[10];
    int s;
}STU;
void fun(STU a[], STU *s)
{   int i,min;
    min=a[0].s;
    for(i=0; i<N; i++)
        if(a[i].s<min)
            {   min=a[i].s;

```

```

        *s=a[i];
    }
}
int main()
{
    STU a[N]={ {"A01",81}, {"A02",89}, {"A03",66}, {"A04",87}, {"A05",77},
               {"A06",90}, {"A07",79}, {"A08",61}, {"A09",80}, {"A10",71}}, m;

    int i;
    printf("**** The original data ****\n");
    for(i=0;i<N; i++)
        printf("N0=%s Mark=%d\n", a[i].num,a[i].s);
    fun(a,&m);
    printf("***** The result *****\n");
    printf("The lowest : %s,%d\n", m.num, m.s);
    return 0;
}

```

程序运行结果如下:

```

**** The original data ****
N0=A01 Mark=81
N0=A02 Mark=89
N0=A03 Mark=66
N0=A04 Mark=87
N0=A05 Mark=77
N0=A06 Mark=90
N0=A07 Mark=79
N0=A08 Mark=61
N0=A09 Mark=80
N0=A10 Mark=71
***** The result *****
The lowest :A08 ,61

```

实验十 文件

(1) 程序改错

第 1 处: "r"应改为"w"

第 2 处: &str 应改为 str

第 3 处: *fp 应改为 fp

(2) 程序改错

第 1 处: file fp;应改为 FILE *fp;

第 2 处: *p=stu 应改为 p=stu

(3) 程序填空

第 1 处填空: FILE

第 2 处填空: exit(0)

第 3 处填空: fclose(fp)

(4) 程序填空

第 1 处填空: n=strlen(str)

第 2 处填空: j<n-i-1

第 3 处填空: fputs(str,fp)

(5) 编写程序

程序如下:

```
#include<stdio.h>
int main()
{   long f1=1,f2=1;
    int i;
    FILE *fp;
    if((fp=fopen("fib.txt","w+"))==NULL)
        {   printf("can't open this file.\n");exit(0);}
    for(i=1;i<=20;i++)
    {
        fprintf(fp,"%15ld%15ld",f1,f2);
        if(i%2==0) fprintf(fp,"n");
        f1=f1+f2;
        f2=f2+f1;
    }
    rewind(fp);
    printf("n");
    for(i=1;i<=40;i++)
    {
        fscanf(fp,"%ld",&f1);
        printf("%15ld",f1);
        if(i%4==0)printf("n");
    }
    fclose(fp);
    return 0;
}
```

(6) 编写程序

程序如下:

```
#include<stdio.h>
#define N 5
struct stu
{   int num;
```

```

    char name[10];
    float score;
} st, *sp;
int main()
{
    FILE *fp;
    int i;
    float x, ave=0;
    sp=&st;
    fp=fopen("stud.dat", "wb+");
    printf("input number name score of %d students:\n", N);
    for(i=0; i<N; i++)
    {
        scanf("%d%s%f", &sp->num, sp->name, &x);
        sp->score=x;
        fwrite(sp, sizeof(struct stu), 1, fp);
    }
    rewind(fp);
    printf("\nnumber    name        score\n");
    for(i=0; i<N; i++)
    {
        fread(sp, sizeof(struct stu), 1, fp);
        printf("%-10d%-10s%-10.0f\n", sp->num, sp->name, sp->score);
        ave=ave+sp->score;
    }
    ave=ave/N;
    printf("average=%0.0f\n", ave);
    fclose(fp);
    return 0;
}

```

实验十一 综合设计

分析：从题目要求可知，每一名学生的记录包括 5 个属性——学号、姓名、 M 门功课的成绩、总成绩和平均成绩。每个属性需要使用不同的数据类型来表示，学号可以用整数表示，姓名必须用字符串表示，每科成绩可用整数表示， M 门功课的成绩可用整型数组表示，总成绩可用整数表示，平均成绩用实数表示。因此，每一名学生的记录可用结构体表示。

程序如下：

```

struct student
{
    int number;
    char name[15];
    int score[M];
    int sum;
    float average;
}

```

}

根据题目要求,可编写4个函数来分别完成相应的功能。同时为了方便用户使用,还可编写一个供用户选择功能的 Menu 函数。本题选择使用简单的字符界面作为操作菜单。菜单的形式确定后,从程序设计角度画出系统结构图,如图 3-1 所示。

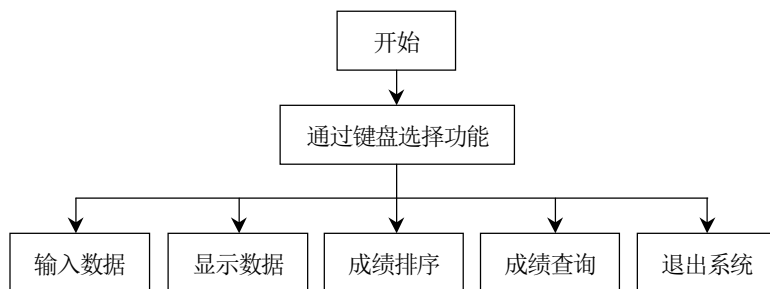


图 3-1 系统结构图

根据以上分析,可编写如下程序:

```

#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
#define N 3          /* 学生人数 */
#define M 2          /* 功课数 */
struct student
{
    int number;       /* 学号 */
    char name[15];    /* 姓名 */
    int score[M];     /* M 门功课的成绩 */
    int sum;          /* 总成绩 */
    float average;    /* 平均成绩 */
};
typedef struct student STU;
/* Output 函数
   功能: 输出 n 名学生的信息
   参数: 指向结构体数组的指针 pt 表示想要输出的学生信息的起始地址
   返回值: 无
*/
void Output(STU *pt, int n)
{
    STU *p;
    int i;
    printf("%8s%10s", "number", "name"); /* 以下三行输出表头 */
    for(i=0; i<M; i++)
        printf("%7s%d", "score", i+1);
    printf("%8s%8s\n", "sum", "average");
    for(p=pt; p<pt+n; p++) /* 输出 n 名学生的信息 */

```

```

{
    printf("%8d%10s",p->number,p->name);
    for(i=0;i<M;i++)
        printf("%8d",p->score[i]);
    printf("%8d%8.1f\n",p->sum,p->average);
}
}
/* Input 函数
   功能：从键盘输入学生信息
   参数：指向结构体数组的指针 pt 表示想要输入的学生信息的起始地址
   返回值：无
*/
void Input(STU *pt)
{
    int i;
    STU *p;
    for(p=pt;p<pt+N;p++)
    {
        printf("\n Input number of student:");
        scanf("%d",&p->number);
        printf("\n Input name:");
        scanf("%s",p->name);
        p->sum=0;
        for(i=0;i<M;i++)
        {
            printf("\n Input score%d:",i+1);
            scanf("%d",p->score+i);
            p->sum+=(p->score+i);
        }
        p->average=p->sum/(float)M;
    }
}
/* Sort 函数
   功能：使用选择排序法按总成绩由高到低进行排序
   参数：pt 表示指向结构体数组中第一个元素的指针
   返回值：无
*/
void Sort(STU *pt)
{
    int i,j,k;
    STU temp;
    for(i=0;i<N;i++)
    {
        k=i;
        for(j=i;j<N;j++)

```

```

        if((pt+j)->sum>(pt+k)->sum) k=j;
    if(k!=i)
    {
        temp=*(pt+k);
        *(pt+k)=*(pt+i);
        *(pt+i)=temp;
    }
}
}
/* Find 函数
   功能: 按学号查找学生
   参数: pt 表示指向结构体数组中第一个元素的指针
   返回值: 无
*/
void Find(STU *pt)
{
    int i,k=-1,n;
    printf("Please input the number you want to search:");
    scanf("%d",&n);
    for(i=0;i<N;i++)
        if((pt+i)->number==n) k=i;
    if(k==-1)
        printf("\n Not found!\n");
    else
        Output(pt+k,1);
}
/* Menu 函数
   功能: 显示菜单并获知用户通过键盘选择的功能
   参数: 无
   返回值: 用户通过键盘输入的字符
*/
char Menu(void)
{
    system("cls");    /* 清屏函数 */
    printf("\n*****\n");
    printf("      1. Input record\n");
    printf("      2. List record\n");
    printf("      3. Sort record\n");
    printf("      4. Find record\n");
    printf("      5. Exit \n");
    printf("*****\n");
    printf("Input your choice:\n");
    return getch();
}
int main()            /* 主程序 */

```



```

{
    STU stu[N];
    while(1)
    {
        switch(Menu())    /* 显示菜单并获得用户输入的字符 */
        {
            case '1':
                Input(stu);    /* 输入学生信息 */
                break;
            case '2':
                Output(stu,N);    /* 显示学生信息 */
                break;
            case '3':
                Sort(stu);    /* 排序 */
                printf("Sorted result\n");
                Output(stu,N);    /* 显示排序结果 */
                break;
            case '4':
                Find(stu);    /* 查找学生信息 */
                break;
            case '5':
                exit(0);    /* 退出程序 */
                break;
            default:
                break;
        }
        printf("\nPress any key to menu");
        getch();
    }
    return 0;
}

```

程序运行后将会出现以下菜单：

```

*****
1. Input record
2. List record
3. Sort record
4. Find record
5. Exit
*****
Input your choice:

```

用户通过键盘输入 1，屏幕上将出现如下信息：

```

Input number of student:1✓

```

```

Input name:LiuMing✓
Input score1:78✓
Input score2:80✓
Input number of student:2✓
Input name:WingLi✓
Input score1:60✓
Input score2:90✓
Input number of student:3✓
Input name:ZhangHong✓
Input score1:77✓
Input score2:88✓
Press any key to menu

```

按任意键，返回菜单，通过键盘选择 2，屏幕上将显示 3 名学生的信息：

```

number  name  score1  score2  sum  average
    1  LiMing    78    80   158    79.0
    2  WangLi    60    90   150    75.0
    3  ZhangHong 77    88   165    82.5
Press any key to menu

```

按任意键，返回菜单，通过键盘选择 3，屏幕上将显示如下信息：

```

Sorted result
number  name  score1  score2  sum  average
    3  ZhangHong 77    88   165    82.5
    1  LiMing    78    80   158    79.0
    2  WangLi    60    90   150    75.0
Press any key to menu

```

按任意键，返回菜单，通过键盘选择 4，屏幕上将显示如下信息：

```

Please input the number you want to search:

```

按任意键，通过键盘选择 2，屏幕上将显示以下信息：

```

number  name  score1  score2  sum  average
    2  WangLi    60    90   150    75.0
Press any key to menu

```

按任意键，返回菜单，通过键盘选择 5，便可退出程序。

程序说明：程序所能管理的学生人数 N 和功课数 M 是在程序中确定的，想要调整的话，就必须修改程序，这正是静态结构体数组的致命缺陷。为了解决这个问题，我们需要使用动态链表结构。