

课程设计指导

第一篇

- 第 1 章 课程设计目的及要求
- 第 2 章 课程设计选题
- 第 3 章 课程设计评价

1.1 课程设计的目的和任务

C 语言是一种编程灵活、特色鲜明的程序设计语言。学习 C 语言除了学习必需的基本知识,如概念、方法和语法规则外,更重要的是要上机实践。C 语言课程设计是很多高校计算机相关专业的必修课程,是 C 语言程序设计课程的后续实践环节。它是根据教学计划的要求,在教师的指导下,对学生实施程序设计训练的必要过程,是对前期课堂教学效果的检验。相比 C 语言程序设计课程的上机实验环节,C 语言课程设计阶段要求更高,选题也更接近实际应用。通过 C 语言课程设计,可以达到以下目的。

(1) 深入理解 C 语言理论知识。学生通过完整项目的开发,可以更好地巩固 C 语言程序设计课程学习的内容,对先前学习的 C 语言理论知识有一个更深层次的认识,深入理解结构体、指针、链表、动态分配内存和文件操作等 C 程序设计中的中高级技术。

(2) 提高分析和解决问题的能力。课程设计为学生提供了一个既动手又动脑,独立实践的机会,有助于学生独立思考,并将课本上的理论知识和实际应用有机地结合起来,使学生了解高级程序设计语言的结构,掌握基本的程序设计过程和技巧,掌握基本的分析问题和利用计算机求解问题的能力,具备初步的高级语言程序设计能力。

(3) 提高 C 语言编程能力。使学生掌握面向过程语言的结构化程序设计方法,强化上机动手编程能力,熟练掌握 C 语言的调试方法,初步培养良好的编程习惯和编程风格,切实提高学生程序设计、调试、测试等工程实践能力。

(4) 提高学生的综合素质。包括创新能力、团队合作能力、项目文档撰写能力等。合作开发模式有助于增强同学之间的团队合作精神,也体会到今后工作中团队合作的重要性和必要性。

(5) 为后续课程奠定基础:为后续课程(如面向对象程序设计、数据结

构、Java 程序设计等)奠定必要的实践基础,使学生通过课程设计掌握高级编程语言的知识和编程技术,掌握程序设计的思想和方法,具备利用计算机求解实际问题的能力。

课程设计的任务是通过布置具有一定难度和编程工作量的课程设计题目,让学生系统地、综合地根据所学习的 C 语言相关知识,编写一个功能完善、实用性强,知识点覆盖面广的应用程序。要求学生能够遵循软件开发过程的基本规范,运用结构化程序设计的方法,按照课程设计的题目要求,独自地完成设计、编写、调试和测试应用程序及编写文档的任务。

采取的教学模式通常是指导教师指定若干课程设计的题目,学生 3 或 4 人一组,自由分组,并在指定题目中自主选题,进行合作项目开发,完成老师指定的题目,课程设计结束后提交设计报告,并进行设计答辩,根据小组项目完成质量和个人的表现对课程设计实践成绩进行评定。

1.2 课程设计的过程

课程设计主要过程分为如下 5 个阶段。

1. 选题阶段

学生在指导老师的指导下按照课程设计的要求和自己的实际情况进行选题。学生在选择过程中按照候选任务书的难易程度、小组成员的兴趣等因素,寻找合适的课程设计题目,并经指导教师确认后定题,一旦题目选定,不允许随意更改。

2. 分析设计阶段

指导教师应积极引导自主学习 and 钻研问题,明确设计要求,找出实现方法,按照需求分析、总体设计、详细设计这几个步骤进行。并在规定时间内上交设计报告书。

3. 编码调试阶段

根据设计分析方案编写 C 代码,然后调试该代码,实现题目要求的功能。并在规定时间内反馈工程实现情况。

4. 总结报告阶段

将以上各阶段的成果和文档汇总成最终的课程设计报告,总结设计工作,要求学生按照需求分析、总体设计、详细设计、编码、测试的步骤撰写报告内容。

5. 答辩考核阶段

学生课程设计完成后,须提交相关课程设计报告纸质版,准备答辩 PPT 演示文档,并完成答辩,成绩由多部分组成,具体可参考第 3 章课程设计评价的内容。

1.3 课程设计的要求

学生开展课程设计通常要满足以下基本要求。

(1) 要求学生熟练掌握 C 语言的基本概念、基本数据类型、基本语句、数组、函数、指针、结构体类型、链表的处理及其灵活应用,掌握 C 语言中文件的操作和使用方法;

(2) 理清系统的总体框架,合理地划分系统的功能模块,画出功能模块图;

(3) 要求使用 C 语言利用面向过程的结构化程序设计方法和模块化思想编程,突出 C 语言的函数特征,以多个函数实现每一个子功能;

(4) 各模块单独编写程序代码,分别测试,最后整合各个模块的功能进行联合调试。各组中的同学之间开展讨论和协作、合理分工,认真完成课题;

(5) 必须实现需求分析中确定的基本功能,达到课程设计基本要求;

(6) 在源程序中合理使用注释,使程序容易阅读和理解;

(7) 程序界面要求友好、直观、易操作,能够进行菜单式功能选择,进行简单界面设计;

(8) 具有清晰的程序流程图和数据结构的详细定义;

(9) 程序应具备一定的容错能力。

参加课程设计的学生,应当认真完成课程设计的全部过程,满足课程设计的基本要求。在达到基本要求之后,鼓励学生进行创新设计,并以最终课程设计成果来证明其独立完成各种实际任务的能力,从而反映出理解和运用本课程知识的水平和能力。课程设计结束后,要求提交的材料包括:

(1) 课程设计报告电子版和纸质版各一(按格式书写);

(2) 源程序一份,能编译成可执行文件并能正常运行(如有特殊运行要求,需要附说明书一份);

(3) 答辩 PPT 演示文稿(含程序演示录制的视频文件)。

2.1 课程设计选题要素

恰当的选题是开展课程设计的前提,通过对常规 C 语言课程设计的项目选题进行调研和梳理,可以总结出一个好的选题需要满足以下几个关键要素。

(1) 可实施性。课程设计的选题首先要符合教学目标,使学生能够运用理论课程中所学的基本知识,进行基本技能方面的训练,具有可实施性。要求难度适中,让不同基础的学生经过努力都可以完成任务、有所收获,不能制定规模过大、要求过高、不切实际的题目。

(2) 可扩展性。完成课程设计选题所需要的绝大多数理论知识应该在相应的理论课程中讲授过,但考虑到课程设计的题目比理论课程中的练习题复杂度高,必然会应用一些没有学过的知识,然而,对这些需要扩展的知识,教师应在设计过程中补充讲解。

(3) 典型性。题目要具有较好的典型性和代表性,便于学生通过一个项目的实践,掌握一类项目开发所需要的相关知识和技术,最终实现举一反三的能力。

(4) 趣味性。题目要尽可能具有趣味元素,从而激发学生的课程设计的积极性,能够主动投入到课程设计工作中去,实现以学生为中心的自主学习。

(5) 新颖性。除了典型性和趣味性,选题还应该具有一定的新颖性,如果要求学生完成一些和课程设计教科书上要求完全一致的项目,学生的学习积极性和实践能力的提高将大打折扣。因此,项目选题应尽可能结合生产、科研、管理、教学等方面的实际需求进行命题,并具有启发性,鼓励学生大胆创新、在题目大框架下自行挖掘和开发特色功能。

2.2 课程设计题目类型

C 语言课程设计的常规题目可以有以下几个类别。

(1) 信息管理类。信息管理类项目是指开发一些小型的 MIS,如图书管

理系统、学生成绩管理系统、员工工资管理系统、通讯录、飞机订票系统等,这些项目可以综合应用到 C 语言中的一些难点知识点,如指针、结构体、链表、文件等,非常有利于加深学生对相关知识的理解。

(2) 游戏开发类。游戏开发类项目是指开发一些适合用 C 语言编程的小型游戏项目,如贪吃蛇、推箱子、俄罗斯方块、五子棋、迷宫等,这些游戏项目可以激发学生的学习兴趣,提高课程设计的积极性,能收到较好的效果。

(3) 应用工具类。该类项目主要是一些非常实用的小工具,比如电子时钟、万年历、画图板、计算器、文本编辑器等。这些项目与日常生活非常接近,有助于学生准确获取项目需求,并需要综合运用 C 语言中的很多知识点,如数组、结构体等,以及一些图形编程技术。

(4) 网络编程类。网络编程类项目可以让学生了解网络编程函数和编程技巧,理解网络协议和套接字编程的基本概念,常见的开发项目包括 Ping 程序设计、TCP 程序设计、UDP 程序设计、HTTP 程序设计。通过开发服务器端和客户端程序,实现基本的通信。

(5) 嵌入式应用类。C 语言在嵌入式系统中的应用具有绝对优势,嵌入式操作系统下的应用开发也是 C 语言最主要的应用场合之一,包括驱动程序开发、Linux 下的应用程序等。需要了解嵌入式操作系统的相关知识。

2.3 课程设计选题建议

C 语言课程设计实践环节的项目选题要能够符合学生的知识结构,缩小与先导课程的跨度,让每个学生都能在有限的时间内有效开展课程设计工作,并有所收获。项目开发使用的知识和技术绝大部分应该是学生已经学过的,仅可以补充少量的学生没有学过的知识和技术。

通常“C 语言程序设计”是开展“C 语言课程设计”的先导课程,通过对 C 语言课程设计常规选题的调研,2.2 节已经总结了主要的 5 大类课程设计选题,考虑到先导课程中已经学习过的关键知识点,以及参与 C 课程设计环节的学生类别,对课程设计选题给出如下建议。

(1) 对于大学低年级同学,由于很多计算机基础课尚未学习,对程序的系统概念,特别是整体设计的概念没有真正理解,如果项目要求采用的技术是先导课程中没有讲授的(比如数据库技术、网络通信技术等),势必违背学生的认知规律,难以保证课程设计的实践效果。因此,适合低年级同学的项目选题包括游戏类、基于文件存储的信息管理类和应用工具类。

这些选题可以通过 C 语言 Win32 控制台应用程序来实现,和学生在先导课程中的要求一致。其中,基于文件存储的信息管理类项目采用文件形式进行数据的存储和读取,从而复习和巩固先导课程中介绍的文件关键知识点,同时避免了使用未学过的数据库。游戏类项目具有极大的趣味性,可以提高学生的学习兴趣和实践的积极性,应用工具类项目也是非常具有代表性的,对提升学生的实践能力有很大帮助。即使希望获得更好的界面效果,也只需要补充一些图形编程技术相关的图形描画库的知识介绍即可,从而很好地缩小了与先导课程的跨度,实现平稳过渡和有效衔接,符合学生的认知规律,最大程度提升学生的学习兴趣,保证课程设计的效果。

(2) 对于大学高年级同学,由于已经学习了计算机网络、数据库原理、嵌入式系统原理等基础课程,因此,可以开展更加复杂的课程设计项目,例如,网络编程类别的项目(基于网

络协议的通信聊天软件等)、嵌入式应用类的项目(Linux 操作系统下的应用开发等)和基于数据库存储的信息管理类的项目(面向 MySQL、SQL Server 数据库的信息管理系统等)。

2.4 课程设计任务书

任务书是课程设计题目的具体表现形式,在课程设计开始时下发给学生。课程设计任务书可以有以下两种形式。

(1)“命题式”任务书。“命题式”任务书中的要求往往过于细化,通常把一个软件需要实现的功能一一清楚地列举出来,甚至给出了最终要求的界面截图。并且最终验收考核时,对照项目功能要求,把项目的完成程度作为一个重要的考核指标,功能实现完整的小组可以获得较高的评价,而功能开发不够全面,即使部分功能开发的较为完善,也不能获得较高的评价,甚至开发了没有要求的功能,也不会获得额外的加分。在这种评价机制的指导思想下,学生们只会对照老师要求的功能逐一开发,不会过多地思考软件的需求定位,也不会软件设计上花费过多的时间,最终导致多个小组开发的项目功能和界面基本完全一致。这种做法,势必对学生的项目开发积极性造成影响,也会限制学生的创新力,不利于学生创新能力的培养。

(2)“启发式”任务书。“启发式”任务书不必细化项目需求,仅提供项目应用领域以及一些提示性的功能建议,注重让学生去挖掘项目需求,撰写需求文档,并以此驱动项目开发的全过程,学生自主开展项目的设计工作,包括界面设计、功能模块设计、数据结构及文件存储结构设计等。从而引导学生独立思考、主动探究、自主学习,并且有目的、有计划、有效率地开展项目开发,提高学生创新能力、实践能力和自主学习能力。

在最终考核时,会考虑项目开发各个阶段的完成情况,综合各阶段表现对学生进行评估,并且除了考虑项目的复杂度和工作量指标外,还会重点考察项目组的创新能力、项目的功能特色或技术特色,对创新性也作为考核的指标,因此,创新性强的小组也可以获得更高的评价。“启发式”任务书可以引导学生自主开展项目需求分析、设计、编码和测试工作,提升学生独立分析问题、解决问题的能力,以及有利于学生创新思维的培养。

3.1 课程设计报告

3.1.1 课程设计报告的内容

课程设计报告是进行课程设计评价的重要文档,如果课程设计采取的是分组开发的形式,则每个小组提交一份课程设计报告即可。课程设计报告通常应该包含如下几个方面的内容。

(1) 封面。写明课程设计题目、组长姓名和学号、项目组成员姓名和学号、指导教师、完成日期等。

(2) 项目参数。包括任务书内容、开发工具、开发平台、代码行数、开发周期、成员详细分工等。

(3) 报告正文。

① 目录:生成报告正文1级~3级标题的目录。

② 设计目的:程序功能简介、涉及技术介绍等。

③ 需求分析:程序详细功能需求,以及必要的性能需求分析。

④ 总体设计:程序模块划分,绘制系统功能结构图。

⑤ 详细设计:程序模块内部的详细设计,包括函数划分,以及函数内部逻辑设计,绘制程序流程图。

⑥ 实现:给出核心功能的源代码及代码分析过程,要求代码符合编码规范,并撰写适量的代码注释。

⑦ 测试:对核心功能模块编写测试用例,并整理测试用例表,重点关注错误输入和边界值输入的测试用例。

⑧ 设计总结:撰写设计体会,总结本次设计所取得的经验和收获,重点对设计过程中遇到的困难,以及解决的方法进行阐述。如果程序未能全部调试通过,则应分析其原因。

⑨ 参考文献:给出设计和书写报告中所参考的文献列表。

(4) 评分表。包括项目评分和成员评分两个部分。具体评价指标参见

3.3 节,成员评分依据项目评分和成员参与程度给出最终每个同学的成绩。

3.1.2 课程设计报告里程碑

课程设计报告不是课程设计全都结束之后再提交,而是在课程设计各个阶段分别提交阶段性课程设计报告,是指导教师审查课程设计进度的重要依据,作为推进课程设计各个阶段工作的里程碑。通常可以设置以下三个时间节点。

(1) 总体设计结束后。总体设计结束后,学生通常已经明确了自己小组的开发任务和需要的技术,详细划分了系统的功能模块,设计了功能结构图,并在小组内部进行了开发分工。此时提交的课程设计报告,应完成 3.1.1 节介绍的“①~④”部分。

(2) 编码结束后。编码结束后,学生通常已经基本实现了项目的主要功能,每个小组成员针对自己负责的模块进行详细设计,绘制了各自模块内部核心函数的程序流程图。此时提交的课程设计报告,应完成 3.1.1 节介绍的“①~⑥”部分。

(3) 测试结束后。测试结束后,经过对项目的各个功能模块进行的详细测试,项目中的 Bug 已经基本排除,程序趋于完善,课程设计已经接近尾声。此时提交的课程设计报告即为最终完整的课程设计报告。

3.2 课程设计答辩

课程设计答辩不是必须要开展的环节,教师可以根据实际情况决定是否答辩。课程设计答辩是指以小组为单位,对课程设计成果物(主要为文档、源代码、可执行程序)进行集中展示和讲解,通过教师提问和学生回答,作为课程设计评价的重要依据。

开展课程设计答辩具有以下优点。

(1) 有助于提高学生的重视程度。学生在课程设计开始就被告知最后要进行课程设计答辩,比没有答辩要求的学生重视程度更高,能付出更多的努力,以期在答辩环节能经得起全班同学的比较和顺利回答老师的提问。

(2) 有助于锻炼学生的综合能力。课程设计答辩,有助于提高学生的语言表达能力,归纳总结能力,演示文档撰写能力等综合能力,在当今激烈的社会竞争中,这些能力对学生的未来发展也是至关重要的,只会技术,不会把自己的设计表达出来,不能很好地与他人沟通,将不利于学生的全面发展。

(3) 有助于教师对课程设计进行评分。答辩时的表现,通常可以直接反映出一个小项目开发的技术水平,通过项目的现场演示,可以直观了解一个项目的设计难度、实现程度、界面友好性等情况。同时,答辩还可以间接反映出小组每个成员的学习态度和参与程度。通过面对面的答辩环节,可以深入了解每个学生的实际参与情况、技术掌握情况等。

课程设计答辩时,要提交课程设计报告纸质版文档给答辩评委,准备答辩 PPT 演示文档,以及录制的程序运行演示视频,答辩时间通常不少于 10 分钟,提问不少于 5 分钟。答辩结束后,教师负责在课程设计报告最后一页评分表上给出评分。

3.3 课程设计评价机制

3.3.1 课程设计评价分级标准

评价是检测学生理解问题和解决问题能力的一个重要手段,教师需要严格跟踪课程设计进度,审查学生各个阶段提交的文档。最终考核验收时由指导教师根据本课程设计的要求严格把关,公平公正,仔细评审学生提交的课程设计报告,认真组织课程设计答辩,对学生的学习态度、出勤情况、动手能力、独立分析和解决问题的能力、创新精神、设计报告质量和答辩水平等指标进行综合考评。建议教师在成绩记录时将学生的成绩分为 A、B、C、D、E 5 个档次。以下是评判标准。

优(A): 按要求完成题目,有完整的符合标准的文档,文档有条理、文笔通顺,格式正确,其中有总体设计思想的论述,有正确的流程图,程序完全实现设计方案,设计方案先进,软件可靠性好。答辩回答问题正确,对系统的演示流畅,源代码解释清晰。

良(B): 完成设计题目,有完整的符合标准的文档,文档有条理、文笔通顺,格式正确;有完全实现设计方案的软件,设计方案较先进。答辩回答问题较好,对系统的演示较为流畅,源代码解释较为清晰。

中(C): 基本完成题目,有完整的符合标准的文档,有基本实现设计方案的软件,设计方案正确。答辩回答问题基本正确,对系统的演示基本完成,源代码解释较为清楚。

及格(D): 基本完成题目,有完整的符合标准的文档,有基本实现设计方案的软件,设计方案基本正确。答辩回答问题基本正确,系统演示能够完成。源代码解释基本清楚。

不及格(E): 没有完成题目的要求,没有完整的符合标准的文档,软件没有基本实现设计方案,设计方案不正确。答辩回答问题不正确,系统演示不能够完成,源代码解释不清楚。

3.3.2 课程设计评价指标

每个小组成员的最终成绩可以由阶段检查成绩、小组项目设计水平和组长打分三部分组成,其中,阶段检查占 30%,项目设计水平占 60%,组长打分占 10%。具体的评分细则描述如下。

(1) 阶段检查。阶段检查占 30 分,主要由平时出勤情况和三个阶段里程碑提交的报告质量,以及每个团队成员的参与情况综合决定。

(2) 小组项目设计水平。小组项目评分占 60 分,主要分为 5 个方面进行评价: 问题规模(15 分)、技术难度(15 分)、实现程度(15 分)、报告质量(10 分)、答辩情况(5 分)。表 3.1 是项目设计水平的评分细则。

表 3.1 项目设计水平的评分细则

考察项目	总分	评分细则	分数
问题规模	15 分	创新完成小组指定任务,工作量饱满	12~15 分
		全部完成小组指定任务,工作量略显不足	9~12 分
		少量小组指定任务未完成,工作量明显不足	5~9 分
		大量的小组指定任务未完成,工作量明显不足	0~5 分

续表

考察项目	总分	评分细则	分数
技术难度	15分	采用技术先进、模型正确	12~15分
		采用技术得当、模型正确	9~12分
		采用技术不够先进、模型存在一些问题	5~9分
		采用技术落后、模型存在很多问题	0~5分
实现程度	15分	系统实现完整,界面友好,没有错误	12~15分
		系统实现完整,界面友好,存在少许错误	9~12分
		系统实现不够完整,界面不够友好,存在一些错误	5~9分
		系统实现不够完整,界面不够友好,存在明显错误	0~5分
报告质量	10分	报告完整、格式统一、结构清晰、图表正确	9~10分
		报告完整、格式较为统一、结构较为清晰、图表较为正确	7~9分
		报告基本完整、有格式问题、结构基本清晰、图表基本正确	5~7分
		报告不够完整、格式问题较多、结构较混乱、图表有错误	0~5分
答辩情况	5分	答辩表达清晰,主要问题回答正确深入	5分
		答辩表达清晰,主要问题回答正确,但不够深入	4分
		答辩表达基本清晰,主要问题回答基本正确,有一些错误	3分
		答辩表达不清晰,主要问题回答有误或回答不出	0~2分

(3) 组长打分。

课程设计是团队开发方式,项目组长是开发团队的核心,起到了重要的领导作用,因此,在课程设计评分环节应该给项目组长一定的权限。此外,由于项目组长和项目组成员的接触更为密切,更能了解成员的参与程度和技术水平,因此项目组长打分具有一定的说服力。组长评分占10分,主要考察以下两个方面。

一是承担工作量的大小:总分5分。

- ① 工作量饱满(5分);
- ② 工作量一般(3~4分);
- ③ 工作量较少(1~2分)
- ④ 工作量很少(0分)。

二是团队精神:总分5分。

- ① 积极参与团队讨论,主动沟通,对小组工作提出建设性的建议(5分);
- ② 能参与团队讨论,但较为被动(3~4分);
- ③ 较少参与团队讨论(1~2分);
- ④ 基本不参与团队讨论(0分)。

3.3.3 课程设计评分表

课程设计报告最后一页的评分表可以设计成如表3.2所示的格式。其中,小组项目设计水平得分 $A=A_1+A_2+A_3+A_4+A_5$,总分60分。每一个成员的最终成绩由小组项目设计水平得分 A +阶段评分 B +组长评分 C 组成,再转换为分级制 $A\sim E$ 的得分即可。

表 3.2 课程设计评分表

项目	说 明		满 分	小组得分
问题规模	考察工作量是否饱满,是否具有创新性		15	A1
技术难度	考察技术是否先进,模型是否正确		15	A2
实现程度	考察实现是否完整,界面是否友好,是否有 Bug		15	A3
报告质量	考察报告是否完整,格式是否统一,结构是否清晰,图表是否规范		10	A4
答辩情况	考察表达是否清晰,问题回答是否正确		5	A5
项目设计水平总成绩			60	A
成员名	阶段评分(总分 30 分)	组长评分(总分 10 分)	最终得分(总分 100 分)	最终分级(A~E)
成员 1	B1	C1	A+B1+C1	
成员 2	B2	C2	A+B2+C2	
成员 3	B3	C3	A+B3+C3	
成员 4	B4	C4	A+B4+C4	

课程设计预备知识

第二篇

- 第 4 章 C 语言基础知识
- 第 5 章 软件工程基础知识
- 第 6 章 C 语言开发平台
- 第 7 章 数据管理技术
- 第 8 章 图形编程技术

C 语言基础知识

第 4 章

4.1 C 语言概述

在学习 C 语言之前,先了解一下 C 语言的历史。

C 语言于 1972 年 11 月问世。1978 年,美国电话电报公司(AT&T)贝尔实验室正式发布 C 语言。1983 年,由美国国家标准局(American National Standards Institute,ANSI)开始制定 C 语言标准,于 1989 年 12 月完成,并在 1990 年春天发布,称为 ANSI C,有时也被称为 C89 或 C90。

C 语言是一门通用的、模块化、程序化的编程语言,被广泛应用于操作系统和应用软件的开发。由于其高效和可移植性,适应于不同硬件和软件平台,深受开发人员的青睐。

1. C 语言早期发展

1969—1973 年在美国电话电报公司(AT&T)贝尔实验室开始了 C 语言的最初研发。根据 C 语言的发明者丹尼斯·里奇(Dennis Ritchie)说,C 语言最重要的研发时期是在 1972 年。

C 语言之所以命名为 C,是因为 C 语言源自 Ken Thompson 发明的 B 语言,而 B 语言则源自 BCPL 语言。C 语言的诞生是和 UNIX 操作系统的开发密不可分的,原先的 UNIX 操作系统都是用汇编语言写的,1973 年,UNIX 操作系统的核心用 C 语言改写,从此以后,C 语言成为编写操作系统的主要语言。

2. ANSI C 和 ISO C

20 世纪 70 年代到 20 世纪 80 年代,C 语言被广泛应用,从大型主机到小型微机,也衍生了 C 语言的很多不同版本。

为统一 C 语言版本,1983 年,美国国家标准局(American National Standards Institute,ANSI)成立了一个委员会,来制定 C 语言标准。1989 年,C 语言标准被批准,被称为 ANSI X3.159—1989 Programming Language C。这个版本的 C 语言标准通常被称为 ANSI C。又由于这个版本是 1989 年完成

制定的,因此也被称为 C89。

后来 ANSI 把这个标准提交到 ISO(国际化标准组织),1990 年被 ISO 采纳为国际标准,称为 ISO C。又因为这个版本是 1990 年发布的,因此也被称为 C90。

ANSI C(C89)与 ISO C(C90)内容基本相同,主要是格式组织不一样。因为 ANSI 与 ISO 的 C 标准内容基本相同,所以对于 C 标准,可以称为 ANSI C,也可以说是 ISO C,或者 ANSI / ISO C。注意:以后看到 ANSI C、ISO C、C89、C90,要知道这些标准的内容都是一样的。目前,几乎所有的开发工具都支持 ANSI / ISO C 标准,它是 C 语言用得最广泛的一个标准版本。

3. C99

在 ANSI C 标准确立之后,C 语言的规范在很长一段时期内都没有大的变动。1995 年,C 程序设计语言工作组对 C 语言进行了一些修改,成为后来的 1999 年发布的 ISO/IEC 9899:1999 标准,通常被称为 C99。

但是各个公司对 C99 的支持所表现出来的兴趣不同。当 GCC 和其他一些商业编译器支持 C99 的大部分特性的时候,微软和 Borland 却似乎对此不感兴趣。GCC(GNU Compiler Collection,GNU 编译器集合)是一套由 GNU 工程开发的支持多种编程语言的编译器。

4.2 C 语言知识点

4.2.1 数据类型

数据类型是按被定义变量的性质、表示形式、占据存储空间的多少、构造特点来划分的。在 C 语言中,数据类型可分为:基本数据类型,构造数据类型,指针类型,空类型 4 大类,如图 4.1 所示。

数据类型说明如表 4.1 所示。

表 4.1 C 语言数据基本类型

数据类型	说明
基本数据类型	基本数据类型最主要的特点是,其值不可以再分解为其他类型。也就是说,基本数据类型是自我说明的
构造数据类型	构造数据类型是根据已定义的一个或多个数据类型用构造的方法来定义的。也就是说,一个构造类型的值可以分解成若干个“成员”或“元素”。每个“成员”都是一个基本数据类型或又是一个构造类型。在 C 语言中,构造类型有以下几种:数组类型、结构体类型、共用体(联合)类型、枚举类型
指针类型	指针是一种特殊的,同时又是具有重要作用的数据类型。其值用来表示某个变量在内存存储器中的地址。虽然指针变量的取值类似于整型量,但这是两个类型完全不同的量,因此不能混为一谈
空类型	其类型说明符为 void,在后面函数中还要详细介绍

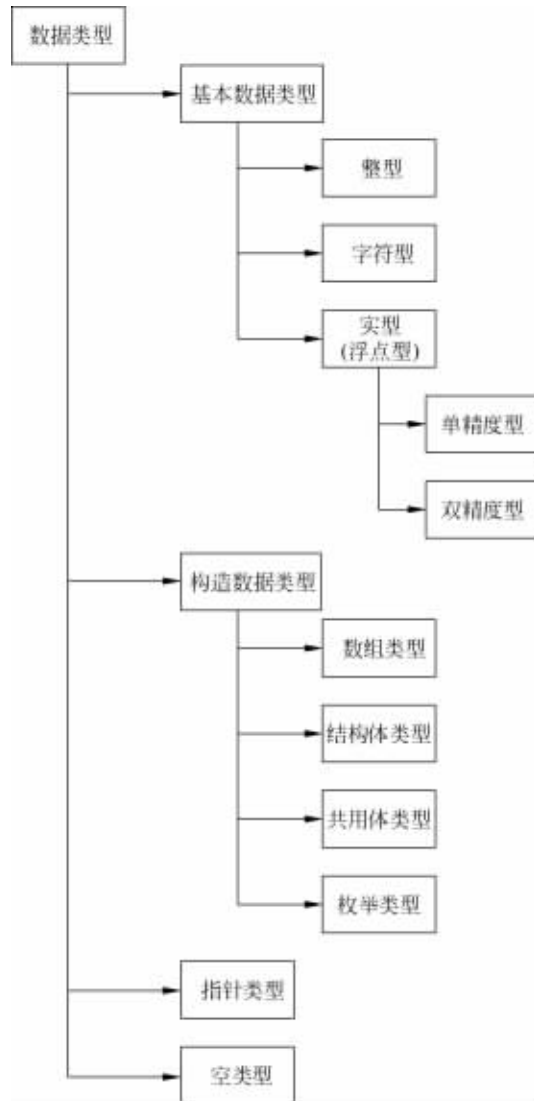


图 4.1 C 语言基本类型

4.2.2 运算符和表达式

C 语言基本算术运算符如表 4.2 所示。

表 4.2 C 语言基本算术运算符

数据类型	说明
算术运算符	用于各类数值运算。包括加(+)、减(-)、乘(*)、除(/)、求余(或称模运算,%)、自增(++)、自减(--)共 7 种
关系运算符	用于比较运算。包括大于(>)、小于(<)、等于(==)、大于等于(>=)、小于等于(<=)和不等不等于(!=)6 种
逻辑运算符	用于逻辑运算。包括与(&&)、或()、非(!)三种

续表

数据类型	说明
位操作运算符	参与运算的量,按二进制位进行运算。包括位与(&)、位或()、位非(~)、位异或(^)、左移(<<)、右移(>>) 6种
赋值运算符	用于赋值运算,分为简单赋值(=)、复合算术赋值(+ =, - =, * =, / =, % =)和复合位运算赋值(& =, =, ^ =, >> =, << =)三类共 11种
条件运算符	这是一个三目运算符,用于条件求值(?:)
逗号运算符	用于把若干表达式组合成一个表达式(,)
指针运算符	用于取内容(*)和取地址(&)两种运算
求字节数运算符	用于计算数据类型所占的字节数(sizeof)
特殊运算符	有括号(),下标[],成员(->..)等几种

表达式是由常量、变量、函数和运算符组合起来的式子。一个表达式有一个值及其类型,它们等于计算表达式所得结果的值和类型。表达式求值按运算符的优先级和结合性规定的顺序进行。单个的常量、变量、函数可以看作是表达式的特例。

4.2.3 输入输出操作

C语言中基本的输入输出函数有以下几个。

putchar(): 把变量中的一个字符常量输出到显示器屏幕上。

getchar(): 从键盘上输入一个字符常量,此常量就是该函数的值。

printf(): 把键盘中的各类数据,加以格式控制输出到显示器屏幕上。

scanf(): 从键盘上输入各类数据,并存放到程序变量中。

puts(): 把数组变量中的一个字符串常量输出到显示器屏幕上。

gets(): 从键盘上输入一个字符串常量并放到程序的数组中。

scanf(): 从一个字符串中提取各类数据。

putchar()和getchar(): 顾名思义就是从输入流中获取一个字符和输出一个字符,比较简单,不再多讲。例如:

```
char c = getchar();
putchar(c);
```

格式化输入输出函数scanf()和printf()是最有用的,所以下面重点讲一下。

1. printf()

一般形式:

```
printf("格式控制",输出列表);
```

例如: printf("a=%d,b=%f,c=%c\n",a,b,c);

1) 格式控制

格式控制是用双引号括起来的字符串,也称“转换控制字符串”,它包含以下两部分信息。

(1) 格式说明: 由“%”和格式字符组成,如%d,%f,%c。它的作用是把输出数据转换为指定格式输出,格式的说明总是由“%”字符开始的。