

【C 语言程序设计】

【The C language Programming】

一、基本信息

课程代码:【2050005】

课程学分:【3】

面向专业:【机械制造】

课程性质:【通识教育基础】

开课院系:【信息技术学院计算机科学与技术系】

使用教材: 主教材【《C 程序设计（第五版）》 谭浩强 清华大学出版社 2010 年 5 月】

辅助教材【《C 语言程序设计与运用》 张晓东等 人民邮电出版社 2014 年再版】

参考教材【The C Programming Language (美国) Brian W. Kernighan 和 Dennis M. Ritchie 机械工业出版社 2006.8 第 1 版】

网络资源【[维基百科 C](#)】

先修课程:【高等数学（下）2100015】、【微型计算机系统 2050151】

二、课程简介

程序设计是理工类各专业的必修课程之一,重在培养学生逻辑思维能力和面向过程的程序设计方法,为今后进一步学习面向对象程序设计打下基础。

通过本课程的学习,使学生运用 C 语言的基本语法和语义,并能进行代码级程序设计;使学生学会从计算机角度思考问题,从而具备解决简单实际问题的 C 程序设计能力;并能把 C 语言作为程序设计的基本工具使用,为今后学习各种以 C 语言为描述工具的后继课程打下基础,也为毕业后可能从事的软件设计与开发工作打好扎实的基础。

本课程教学特点之一:不是简单的按教材上课,而是循序渐进,分模块教学,渐进式引导学生学习 C 语言程序设计。其中第一模块 简单 C 程序设计,第二模块 模块化程序设计,第三模块 复杂数据类型程序设计。实践环节也根据教学模块分为三大块,每个模块均包括三个实验,共九个实验。本课程教学特点之二:注重运用能力的培养,强化实践教学,通过课内外上机实践,提高程序设计能力。

三、选课建议

本课程的前导课程是理解计算机系统概况的课程,例如《计算机导论》。本课程是各种课程设计中需要使用的程序设计工具,更是用来精确地说明有关概念和方法的描述工具。本课程也是面向过程的程序方法的代表课程,更为今后进一步学习面向对象程序设计打下基础。

四、课程与专业毕业要求的关联性

| 计算机科学与技术专业毕业要求 | 关联 |
|--|----|
| LO11: 工程知识: 能够将数学、自然科学、工程基础和专业知识用于解决复杂工程问题 | |
| LO21: 问题分析: 能够应用数学、自然科学和工程科学的基本原理, 识别、表达、并通过文献研究分析复杂工程问题, 以获得有效结论 | ● |
| LO31: 设计/开发解决方案: 能够设计针对复杂工程问题的解决方案, 设计满足特定需求的系统、单元(部件)或工艺流程, 并能够在设计环节中体现创新意识 | |
| LO41: 研究: 能够基于科学原理并采用科学方法对复杂工程问题进行研究, 包括设计实验、分析与解释数据、并通过信息综合得到合理有效的结论 | |
| LO51: 使用现代工具: 能够针对复杂工程问题, 开发、选择与使用恰当的技术、资源、现代工程工具和信息技术工具, 包括对复杂工程问题的预测与模拟, 并能够理解其局限性 | ● |
| LO61: 工程与社会: 能够基于工程相关背景知识进行合理分析, 评价专业工程实践和复杂工程问题解决方案对社会、健康、安全、法律以及文化的影响, 并理解应承担的责任 | |
| LO71: 环境和可持续发展: 能够理解和评价针对复杂工程问题的专业工程实践对环境、社会可持续发展的影响 | |
| LO81: 职业规范: 具有人文社会科学素养、社会责任感, 能够在工程实践中理解并遵守工程职业道德和规范, 履行责任 | |
| LO91: 个人和团队: 能够在多学科背景下的团队中承担个体、团队成员以及负责人的角色 | |
| LO101: 沟通: 能够就复杂工程问题与业界同行及社会公众进行有效沟通和交流, 包括撰写报告和 design 文稿、陈述发言、清晰表达或回应指令。并具备一定的国际视野, 能够在跨文化背景下进行沟通和交流 | |
| LO111: 项目管理: 理解并掌握工程管理原理与经济决策方法, 并能在多学科环境中应用 | |
| LO121: 终身学习: 具有自主学习和终身学习的意识, 有不断学习和适应发展的能力 | ● |

五、课程目标/课程预期学习成果

| 序号 | 课程预期学习成果 | 课程目标 (细化的预期学习成果) | 教与学方式 | 评价方式 |
|----|--|-------------------------|----------|------------|
| 1 | LO211 具备对系统设计、软件开发等涉及到的复杂工程问题进行识别与判断, 并结合专业知识进行有效分解的能力 | 能根据信息描述面向结构化方式进行简单的系统设计 | 讲授、练习、实践 | 实验、报告、实践评价 |
| | LO212 具备 | | 讲授、练习、实 | 实验、测试、 |

| | | | | |
|---|-------------------------------------|---------------------|----------|-------------|
| | 对分解后的复杂工程问题进行表达与建模的能力 | 能根据信息描述进行算法设计 | 践 | 作业 |
| | LO213 具备对复杂工程问题进行分析 and 求解的能力 | 能运用基本流程控制编制程序 | 讲授、练习、实践 | 实验、测试、作业 |
| 2 | LO511 能熟练运用绘图工具，表达和解决计算机系统工程的设计问题 | 能够利用结构化流程图等工具完成算法设计 | 讲授、讨论 | 实验、作业、设计报告 |
| | LO512 能根据具体项目的特点和需求，选择合适的技术工具进行设计开发 | 能结合数据库与 I/O 流完成系统开发 | 讲授、讨论 | 实验、上机练习 |
| 3 | LO121 能够根据课程要求进行自主学习 | 能够主动学习相关知识 | 讨论 | 体现协同学习的作业报告 |

六、课程内容

第 1 章 概论

理解程序设计语言在计算机系统中的地位 and 作用；C 语言基本特点 and 发展简况；源程序 and 可执行程序；编译系统的作用。

理解源程序的结构；源程序编辑、编译、连接 and 执行等上机操作过程；源程序的语法错误 and 逻辑错误。

本章重点是以一个简单计算问题的简单源程序为例，介绍源程序的结构、及从源程序到可执行程序的处理全过程。

第 2 章 算术类型数据

理解常量、数据类型及变量的概念；各种数据类型的值范围 and 内部存储格式；算术类型数据的输入 and 输出，常用库函数。

熟练运用常量的原形式；常用基本数据类型的变量定义。

本章重点是要注意各种不同类型的变量和常量的作用以及它们的区别;设计程序的过程中选择数据类型的一般方法。

第3章 基本运算和表达式

理解基本运算的属性包括功能、目数、优先级、结合性、副作用、可截断性;运算执行时的自动类型转换;数值计算表达式;判断情况的表达式;条件表达式。

熟练运用算术运算 (+ - * / % ++ --); 赋值运算 (= 0=); 关系运算 (< <= > >= == !=); 逻辑运算 (! && ||); 位运算 (<< >> & ~ ^ |)。

本章重点注意各种不同基本运算的功能,以及对参加运算的数据的类型要求。特别注意:

- (1) 算术常规转换和赋值转换这二个自动类型转换规则的效果,以及对计算结果的影响。
- (2) 增减1运算的副作用。
- (3) 逻辑与、逻辑或运算的可截断性。

(4) 表示的真假规则,包括关系运算及逻辑运算的结果类型和结果值,能以值的计算和情况的判断为例,结合表达式展示基本运算的作用和各种特性。

第4章 算法与语句

理解语句和流程控制结构概述。理解C语言语句的类型。运用常用基本算法,基本语句包括表达式语句、复合语句、空语句;选择结构语句;循环结构语句等解决具体问题。

熟练运用 if 和 switch 语句、while/do...while/for 语句、continue 和 break 语句设计程序,及上述语句的执行流程和功能;循环的嵌套的运用。

本章重点是表达式与语句的区别。if...else 条件嵌套的匹配;do...while、while 语句与 for 语句的比较;尤其是 for 语句的执行流程、计数型循环和条件型循环语句的不同设计、语句的综合运用。

教学建议以 if...else 选择为重点,分析 if 语句的执行流程;重视对情况进行判断的表达式构造,进一步熟练地使用各种关系运算和逻辑运算等解决具体问题。

第5章 数组类型

熟练运用一维数组的定义及引用;数值型数组的输入/输出基本处理;字符数组、字符串存储格式及字符串的输入、输出;

运用数组的典型处理解决具体问题。如:求数组的最大(小)值、均值,顺序查找,对分查找,选择交换排序,冒泡排序等;字符串的典型处理,如求串长,字符串复制、连接,字符串的字典次序比较等。

重点强调数组在程序设计中的广泛用途、数组变量定义的整体性和处理的个别性;使用循环

控制语句，按不同的模式处理数组中的数据，重视下标表达式的构造；以数组处理为背景，综合使用循环控制和选择控制语句，尤其是 for-for、for-if 形式的控制；重视字符串在字符数组中的存储格式，结合循环和选择控制语句进行字符串的各种常用处理。

在一维数组的基础上理解二维数组的定义；数组成员的引用及下标表达式；数据的输入/输出。理解二维数组的典型处理，如：计算行（列）和、寻找行（列）的最大（小）值、计算上（下）三角阵数据之和等。

注意数组变量定义的整体性和处理的个别性。

本章重点是使用二重循环控制语句，按不同的模式处理数组中的数据，重视下标表达式的构造及运用。

第 6 章 函数

熟练运用函数的定义；函数的调用；函数调用的实现过程。

理解函数调用时的数据传送机制；函数的原型说明；直接递归函数调用的实现过程；变量的存储类与作用域；预处理程序。运用函数调用规则。

理解直接递归函数定义实例，如计算 n 的阶乘，对一个正数倒序输出等。

注意程序设计的模块化方法；在运用函数定义和函数调用的形式的基础上，理解函数调用的实现过程；特别强调函数头的设计要领，通过大量的函数定义训练，详细理解直接递归函数调用的执行过程；

本章重点是函数定义，函数说明和函数调用的三种格式和用途。注意不同存储类型的变量的生存期限和影响范围的区别。

第 7 章 文件和文件管理

理解文件的基本概念。

运用处理文件的常用库函数：fopen、fclose 等。

运用文本文件处理和常用库函数：fputs、fgets、fprintf、fscanf 等。

运用字节文件处理和常用库函数：fread、fwrite、fseek、ftell 等。

本章重点是要注意文件与变量在数据存储方面的区别和用途，文件处理的基本模式：打开、各种处理、关闭等。

七、课内实验名称及基本要求

| 序号 | 实验名称 | 主要内容 | 实验时数 | 实验类型 | 备注 |
|----|------|------|------|------|----|
|----|------|------|------|------|----|

| | | | | | |
|---|------------|---|---|-----|------|
| 1 | 简单C程序设计 | 运用关系运算表示选择条件，掌握简单分支 if 语句的功能和使用。利用 for 语句掌握计数型简单循环结构的功能和使用。理解函数在 C 程序设计中的作用，掌握函数的定义方式及调用方式。 | 4 | 设计型 | 第一模块 |
| 2 | 模块化程序设计 | 理解嵌套结构概念，掌握选择嵌套、循环嵌套和复合嵌套结构的使用。体验模块化程序设计过程。综合利用已经掌握的相关知识进行 C 程序设计。 | 4 | 设计型 | 第二模块 |
| 3 | 复杂数据类型程序设计 | 掌握一维数组的典型处理。掌握字符串的典型处理。掌握文件操作的常用方法及相应语句的应用技巧，用函数的编程方法来处理结构数组。 | 2 | 设计型 | 第三模块 |

八、评价方式与成绩

| 总评构成 (1+X) | 评价方式 | 占比 |
|------------|--------------|-----|
| 1 | 期终闭卷考 | 40% |
| X1 | 实验、课堂展示、课外学习 | 20% |
| X2 | 上机测试 | 20% |
| X3 | 单元测试 | 20% |

“1”一般为总结性评价，“X”为过程性评价，“X”的次数一般不少于3次，无论是“1”、还是“X”，都可以是纸笔测试，也可以是表现性评价。与能力本位相适应的课程评价方式，较少采用纸笔测试，较多采用表现性评价。

常用的评价方式有：课堂展示、口头报告、论文、日志、反思、调查报告、个人项目报告、小组项目报告、实验报告、读书报告、作品（选集）、口试、课堂小测验、期终闭卷考、期终开卷考、工作现场评估、自我评估、同辈评估等等。**一般课外扩展阅读的检查评价应该成为“X”中的一部分。**

同一门课程由多个教师共同授课的，由课程组共同讨论决定X的内容、次数及比例。

撰写人：张书台

系主任审核签名：

审核时间：

