

## 【程序设计基础 (C 语言)】

### 【Fundamental of Programming (C)】

#### 一、基本信息

课程代码:【2050170】

课程学分:【4】

面向专业:【网络工程】

课程性质:【院级必修课】

开课院系: 信息技术学院软件工程系

使用教材:

教材【C 语言程序设计 (第 4 版),何钦铭等,高等教育出版社,2020.9】

参考书目【C 程序设计语言 (英文版,第 2 版) Brian W. Kernighan Dennis M. Ritchie 机械工业出版社, 2006.8】

先修课程:【无】

#### 二、课程简介

《程序设计基础(C 语言)》是各计算机学科类专业的核心基础课程之一。与《数据结构》、《面向过程程序设计实践》共同构成了“程序设计模块”课程。

“程序设计模块”课程实质是把《程序设计基础(C 语言)》、《数据结构》、《程序设计实践》三门课程的内容整合梳理,打破原有单门课程的界限,以培养学生程序设计能力为本、以“数据对象”为主线、以循序渐进的方式重构模块化课程内容,包括:

- (1) 简单数据模块
- (2) 复合数据模块
- (3) 线性结构模块
- (4) 非线性结构模块
- (5) 检索与排序模块
- (6) 程序设计综合实践

《程序设计基础(C 语言)》在内容上对应(1)(2)模块,通过本课程的学习,重在使学生学会从计算机角度思考问题,培养学生逻辑思维能力和面向过程的程序设计方法;训练学生能运用 C 语言作为程序设计的工具编制结构清晰、可读性好的程序,并会设计测试用例,完成程序的测试。为后继模块的学习并培养面向过程化应用程序的独立设计及实现能力打下一定的理论基础及实践基础。

本课程教学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。

课内总学时为 64,其中 48(边讲边练)+16(课内实验),另外需要学生课外不少于 32 学时的上机实践。

#### 三、选课建议

本课程适用于各计算机学科类专业,需具备一定的理科基础及计算机基本操作能力。建议在

第一或第二学期开设。

#### 四、课程与专业毕业要求的关联性

专业毕业要求	关联
LO1: 工程知识: 能够将数学、自然科学、工程基础和专业知用于解决复杂网络工程问题	●
LO2: 问题分析: 能够应用数学、自然科学和工程科学的基本原理, 并通过文献研究与利用, 识别、表达、分析复杂网络工程问题, 以获得有效结论	●
LO3: 设计解决方案: 能够设计针对复杂网络工程问题的解决方案, 包括满足特定需求的网络系统设计方案、网络工程实施方案和网络测试方案, 并能够在设计环节中体现创新意识, 考虑社会、健康、安全、法律、文化以及环境等因素	
LO4: 研究: 能够基于科学原理并采用科学方法对复杂网络工程问题进行研究, 包括设计实验、分析与解释数据、并通过信息综合得到有效的结论	
LO5: 使用现代工具: 能够针对复杂网络工程问题, 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	
LO6: 工程与社会: 能够基于网络工程相关背景知识进行合理分析, 评价网络工程实践和复杂网络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	
LO7: 环境和可持续发展: 能够理解和评价针对复杂网络工程问题的工程实践对环境、社会可持续发展的影响	
LO8: 职业规范: 具有人文社会科学素养、社会责任感, 能够在网络工程实践中理解并遵守工程职业道德和规范, 履行责任	
LO9: 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	
LO10: 沟通: 能够就复杂网络工程问题与业界同行及社会公众进行有效沟通和交流, 包括撰写报告和设计文稿、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	
LO11: 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	
LO12: 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	

备注: LO=learning outcomes (学习成果)

#### 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	教与学方式	评价方式
1	LO12: 能够将离散数学、算法、数据结构与程序设计等知识与方法, 用于进行计算思维, 用	能够运用程序设计分析方法, 对实际问题进行需求分析	讲课、实验	课堂表现
			讲课、实验	小测验

	于基本算法问题的分析、设计与实现，用于网络系统的工作原理的分析与理解。	能够在需求分析的基础上，找出解决问题的算法		
2	LO21: 能够借助数学、物理和工程科学的基本原理与方法，进行复杂网络系统规划、设计、部署、开发、测试、运维过程中关键工程或技术问题的识别、分析与表达。	根据具体问题的需要，能够运用 C 语言完成程序设计	讲课、实验	期终闭卷考
		根据具体问题，能够进行算法设计	讲课、实验	实验报告
		根据算法设计完成程序后，能够设定合适的测试用例对程序进行测试	讲课、实验	实验报告
		根据课堂讲授的内容，能够在后继课程中灵活运用	讲课、实验	课后练习
		能够将 C 语言中学习到的算法思想，灵活运用到后续的学习中	讲课、实验	课后练习

## 六、课程内容

### 模块 1 简单数据

#### 第 1 单元 关于这门课及 C 语言概述

知道本课程地位、课程目标、辅助资料、教学进度、教学特点、有效的学习方法及课程评价方法。

知道 C 语言在计算机系统中的地位和作用；知道 C 语言基本特点和发展简况。

理解 C 语言源程序的结构；能运用编译器编辑源程序、并对源程序进行编译、连接和执行等操作；理解源程序的语法错误和逻辑错误等问题。

本讲重点是以一个简单源程序为例，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。

本单元的理论课时数 2 学时，实践课时数 0 学时。

#### 第 2 单元 用 C 语言编写程序

理解 C 语言数据表达的基本元素：数据类型、常量及变量的基本概念；知道结构化程序设计的流程控制：顺序结构、选择结构、循环结构三种基本结构；理解程序设计语言的语法要素、函数的作用。

本讲重点以示例引导学生认识 C 语言的各个要素、输入输出、三种流程控制结构、函数等，达到能基本理解 C 语言程序的组成结构，并能简单模仿。

本单元的理论课时数 2 学时，实践课时数 0 学时。

### 第3单元 数据的存储、基本数据类型和表达式

分析程序开发的过程；知道各种数据类型的数值范围和内部存储及输入输出格式控制；能熟练运用常量的原形式和基本数据类型进行变量定义，灵活运用 `int`、`float`、`double`、`char` 等基本类型数据；描述表达式中不同运算符的运算规则。

知道程序设计的过程，并运用主函数、输入/输出函数和数学函数解决简单问题。

本讲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；在设计程序过程能针对问题灵活运用数据类型。

本单元的理论课时数 2 学时，实践课时数 0 学时。

### 第4单元 分支结构

知道 C 语句的组成；理解单分支、双分支及多分支选择控制结构；讨论 `if..else` 多种条件嵌套的匹配规则、`switch` 语句结构的使用特点。

学会运用关系运算、逻辑运算符构造条件表达式，灵活运用 `if-else`、`switch` 语句解决简单选择结构问题。

本讲重点是选择结构程序的实现。

本单元的理论课时数 4 学时，实践课时数 2 学时。

### 第5单元 循环结构

掌握循环控制结构；掌握 `for`、`do...while`、`while` 循环语句的使用方法；比较 `do...while`、`while` 语句与 `for` 语句差异及适用场合分析。

能够灵活应用计数型循环和标识性循环设计的构建方法；灵活运用复合语句、空语句，循环结构语句等基本语句解决循环问题。

本讲重点是循环结构程序的实现。

本单元的理论课时数 4 学时，实践课时数 2 学时。

### 第6单元 数据文件

描述文件的基本概念。

学会运用文件指针和处理文件的常用库函数：`fopen`、`fclose`、`feof`、`fputs`、`fgets`、`fprintf`、`fscanf` 等的使用方式和规则，达到运用程序控制输入/输出数据文件的能力。

本讲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，各种读、写等处理，关闭等。

本单元的理论课时数 4 学时，实践课时数 2 学时

### 第7单元 模块化的 C 程序结构——函数

讨论自顶向下，逐步细化的模块化设计思想划分子模块，知道模块化程序设计方法。

熟练运用函数的定义、函数的调用。

掌握函数调用时的数据传送机制，实参与形参的区别，函数的原型说明，预处理命令等。

在运用函数定义和函数调用形式的基础上，讨论函数调用的实现过程，特别强调函数头的设

计要领。

本讲重点是函数定义、函数说明和函数调用的三种格式和用途；并能运用 return 语句将被调函数的处理结果返回主调函数。学会运用模块化设计方法设计程序解决具体问题。

本单元的理论课时数 4 学时，实践课时数 2 学时。

### 第 8 单元 指针初步

辨别变量的存储类型、作用域、存储区分配。

解释变量地址概念；解释指针的基本概念、定义、赋值及使用方式。

掌握运用指针变量的定义、赋值及引用，解决被调函数向主调函数传递多值问题。

阐释函数的形参是指针的意义，函数的返回值可以是指针类型。

本讲重点是注意变量和变量地址之间的区别，注意指针变量的用途，以及指针基本类型的意义；通过实例讲解函数通过指针类型的参数向外传递计算结果。

本单元的理论课时数 4 学时，实践课时数 2 学时。

## 模块 2 复合数据

### 第 1 单元 数值数组

知道数组的逻辑结构及存储结构、数值型一维数组的定义及使用。

讨论数值型一维数组的一般操作（输入，输出，访问数组元素）、使用指针访问数组元素、数组在函数之间传递的方式。

能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，二分查找，选择交换排序，冒泡排序等。

本讲重点是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；运用循环控制语句，按不同的模式处理数组中的数据，重视下标表达式的构造与循环控制变量的结合。

本单元的理论课时数 6 学时，实践课时数 2 学时

### 第 2 单元 10 章 二维数组

掌握二维数组的声明、初始化及引用。知道二维数组的基本操作方法。

讨论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。

本单元的理论课时数 2 学时，实践课时数 0 学时

### 第 3 单元 字符数组

掌握字符数组的定义、引用和基本操作。掌握字符串的存储及输入/输出操作、字符串在函数之间传递；

本讲重点是字符数组与字符串的区别；字符串的典型处理，如求串长，字符串复制、连接、

比较、截取，字符串的模式匹配（BF 算法）等。

本单元的理论课时数 6 学时，实践课时数 2 学时

#### 第 4 单元 结构类型

知道结构类型的定义、结构变量定义和成员的引用、结构指针的定义及成员的引用。

讨论结构数组及基本操作、结构指针的使用、结构在函数间传递的方式。

本讲重点是结构数组的基本处理，强调结构在程序设计中的广泛用途、结构变量定义的整体性和处理的个别性。

本单元的理论课时数 6 学时，实践课时数 2 学时

### 七、课内实验名称及基本要求（适用于课内实验）

实验序号	实验名称	主要内容	实验时数	实验类型	备注
1	过程控制的程序设计	C 语言的基本要素，if、switch 选择结构语句，for、while、do...while、循环结构语句对简单数据的处理。	4	设计型	模块 1
2	模块化程序设计	选择、循环控制结构语句的综合运用，自定义函数实现模块化程序设计。从文件逐个读取数据到变量；	4	设计型	模块 1
3	一维数组的基本操作	数值型数组及字符串基本运算及其实现，函数之间传递数组，指针访问数组。从数据文件批量读取数值数据到数组；	4	设计型	模块 2
4	批量数据处理	数值数组、字符型数组基本运算及其实现，C 语言字符串处理库函数的使用；从数据文件读取批量数据。	2	设计型	模块 2
5	结构数组的操作	typedef 类型重命名；从数据文件读取数据到结构数组，结构数组的基本操作	2	设计型	模块 2

### 八、评价方式与成绩

总评构成 (1+X)	评价方式	占比
1	期末考试	40%
X1	课堂表现（课堂互动、随堂练习等）	20%
X2	单元测验	20%
X3	作业（包括课内实验及课后作业）	20%

SJQU-QR-JW-033 (A0)

说明：单元测验的次数及大致时间根据每学期的教学进度决定，测验方式：上机。

本课程为考试课程，期末考试由教务处统一安排，考试方式：上机。

撰写人：王敏慧

系主任审核签名：朱丽娟

审核时间：2023年2月20日