

《C 语言程序设计》课程教学大纲

一、基本信息

课程代码	109004	开课学院	电子信息工程
课程名称（中文）	C 语言程序设计		
课程名称（英文）	The C Programming Language		
适用专业	电子科学与技术、微电子科学与工程、集成电路设计与集成系统		
课程类别	<input type="checkbox"/> 通识（通修）类 <input type="checkbox"/> 数学与自然科学类 <input checked="" type="checkbox"/> 工程基础类 <input type="checkbox"/> 专业基础类 <input type="checkbox"/> 专业类 <input type="checkbox"/> 专业拓展类 <input type="checkbox"/> 工程实践与毕业设计（论文）类 <input type="checkbox"/> 职业（方向）类		
课程性质	<input checked="" type="checkbox"/> 必修 <input type="checkbox"/> 限选 <input type="checkbox"/> 任选		
学分	3		
课内总学时		64	
其中：	理论	32	
	上机	32	
	课程实践		
	实验		
	线上（翻转课堂）		
课外学时		86	
智慧教学平台 课程名称和网址			

二、课程简介

本课程是电子信息工程、通信工程、电子科学与技术、微电子科学与工程、集成电路设计与集成系统专业的一门工程基础类课程，高等数学、工程数学、大学物理、电路分析基础等课程是本课程的先修课程，而本课程是 C 语言课程设计、单片机原理及技术等课程的先修课程，在电子信息工程、通信工程、电子科学与技术、微电子科学与工程、集成电路设计与集成系统专业的课程体系中起着承上启下的作用，是非常重要的专业课程。

三、课程思政要求

- 1.职业伦理：厚植职业伦理，提高学生的法律意识和道德意识。
- 2.工匠精神：弘扬工匠精神，培育学生的敬业姿态和创新意识。

四、课程目标及对毕业要求的支撑关系

本课程需支撑电子信息工程等适用专业毕业要求中的 2 个观测点。在毕业要求观测点的指导下，本课程制定了 2 项课程目标和对应的考核办法，详见表 1。

表 1: 课程目标与毕业要求支撑关系

序号	毕业要求	观测点	课程目标	考核方法
1	工程知识	(观测点 1.3): 能将数学、自然科学知识、工程基础和专业知 识,用于描述和解决本专业领域复杂工程问题。	课程目标 1: 能够运用数学、C 语言程序设计方法表达反应工程问题, 获得 C 语言的语 言基础、分支、循环、函数、结构、指针、文件等方面的知识; 能够阅读和应用结构化程序设计方法设计、编写、调试和运行 C 语言程序;	达成度=课程目标 1 得分/课程目标 1 满分
2	设计/开发解决方法	(观测点 3.1): 能够针对特定的复杂工程问题, 设计开发满足需求 的本专业功能部件、测试方案或流程。	课程目标 2: 掌握结构化程序设计的思想和方法, 掌握 C 语言的基本语法, 具有分析问题的能力, 并具有用 C 语言编程解决问题的基本能力; 掌握一种开发工具的使用, 具有一定程度的使用调试器查找问题并解决问题的能力; 具有对简单问题进行分析、并设计算法, 开发软件解决问题的能力; 通过引导学生自学, 初步培养学生适应大学学习, 逐步培养学生的自学能力。	达成度=课程目标 2 得分/课程目标 2 满分

五、教学内容及方法设计

1. 知识单元一: 概述 (支撑课程目标 1)

教学要求:

理解计算机中数据的表示和处理, 理解算法表示的几种方法, 会使用 C 语言集成开发环境, 理解 C 语言的开发过程。

教学内容:

- (1) 知识点一: 计算机如何工作
- (2) ※ 知识点二: 数据在计算机中的表示
- (3) 知识点三: 程序与算法
- (4) ※ 知识点四: C 语言简介
- (5) ※ 知识点五: C 语言集成开发环境

教学方法:

- (1) 讲解加课堂互动的方法, 使学生理解计算机如何工作、数据在计算机中的表示。
- (2) 举例说明程序和算法的区别, 并分别说明各自基本特点。
- (3) 多媒体课件和板书相结合。

2、知识单元二：程序设计起步（支撑课程目标 1）

教学要求：

理解常量和变量的概念与区别；理解 C 语言数据类型的实质及分类，掌握整型、实型、字符型的常量表示，了解各种类型数据占用的内存大小、数据在内存中的存储形式、数据范围等知识；掌握用户自定义标识符的定义规则；熟悉常用运算符的运算方法及其运算优先级、结合方向，正确表示和求解表达式，掌握 C 语言中的类型转换问题。

教学内容：

- (1) 知识点一：C 语言的数据类型
- (2) 知识点二：※常量与变量
- (3) 知识点三：※整型数据，浮点型数据和字符型数据
- (4) 知识点四：变量赋初值，◎赋值运算符和赋值表达式
- (5) 知识点五：※算术运算符和算术表达式
- (6) 知识点六：○逗号运算符和逗号表达式
- (7) 知识点七：各类数值型数据间的混合运算

教学方法：

- (1) 讲解加课堂互动的方法，使学生理解数据类型的基本概念和 C 语言的表达式特点。
- (2) 线上与线下相结合，对重点和难点内容要求学生练习相关习题。
- (3) 多媒体课件和板书相结合。

3、知识单元三：会思考的程序-选择结构（支撑课程目标 1）

教学要求：

重点掌握选择结构的算法构造，if 语句和 switch 语句的语法及其应用。会计算关系以及逻辑表达式的值，理解逻辑表达式的短路现象，能够根据算法需要，使用正确的关系表达式以及逻辑表达式来描述条件；能够构造含有选择结构的算法；熟练掌握阅读、编写、调试含有选择结构算法的程序。

教学内容：

- (1) 知识点一：※关系运算符和关系表达式
- (2) 知识点二：※逻辑运算符和逻辑表达式
- (3) 知识点三：※if 语句
- (4) 知识点四：○switch 语句
- (5) 知识点五：※选择结构程序举例

教学方法：

- (1) 讲解加课堂互动的方法，使学生掌握选择结构程序设计的基本流程和分析方法。

(2) 线上与线下相结合，对重点难点内容要求学生练习相关习题。

(3) 以选择结构为例，理论联系实际。

(4) 利用板书画一些流程图，帮助学生理解程序的执行过程。

4、知识单元四：键盘输入与屏幕输出（支撑课程目标 1）

教学要求：

理解语句的基本概念，掌握语句的功能；掌握字符数据输入输出的实现方法，能够正确的调用 putchar 和 getchar 函数；掌握格式输入输出的实现方法，能够正确的使用 printf 和 scanf 函数，会根据不同的输入格式设置不同的参数；理解顺序结构的概念，能够利用简单的表达式语句和调用输入输出函数语句实现顺序结构程序设计，并初步了解编程环境中调试和修改程序的方法。

教学内容：

(1) 知识点一：C 语句概述和赋值语句

(2) 知识点二：字符数据的输入与输出

(3) 知识点三：※格式输入与输出

教学方法：

(1) 讲解加课堂互动的方法，使学生掌握本章的重点和难点。

(2) 线上与线下相结合，对重点难点内容要求学生练习相关习题。

(3) 以键盘输入和屏幕输出为例，理论联系实际。

(4) 多媒体课件和板书相结合。

5、知识单元五：周而复始-循环结构（支撑课程目标 1、2）

教学要求：

掌握实现循环的三种基本语句：while 语句，do-while 语句，for 语句；掌握三种循环语句之间的相似点和不同点，其中 for 语句最灵活多变，掌握三个变量表达式的意义以及表达式缺省时的 for 语句是如何执行的；掌握嵌套循环是如何执行的，能够使用嵌套循环实现一些具体的功能；掌握 break 语句和 continue 语句，并理解两者在循环语句应用中的区别；能够使用三种循环语句实现一些经典的循环算法，可以分为三大类：递推法，迭代法，枚举法。

教学内容：

(1) 知识点一：用 while 语句实现循环

(2) 知识点二：用 do while 语句实现循环

(3) 知识点三：※用 for 语句实现循环

(4) 知识点四：循环的嵌套和几种循环语句的比较

(5) 知识点五：※break 语句和 continue 语句的使用

(6) 知识点六：◎含循环结构的经典算法求解

教学方法：

- (1) 讲解加课堂互动的方法，使学生掌握本章的重点和难点。
- (2) 线上与线下相结合，对重点难点内容要求学生练习相关习题。
- (3) 以循环结构为例，理论联系实际。
- (4) 多媒体课件和板书相结合。

6、知识单元六：模块化设计-函数（支撑课程目标 1、2）

教学要求：

重点掌握用户自定义函数的定义、调用的基本方法；会正确设定形式参数并能正确传入实参调用函数，理解参数值传递的概念；了解函数声明的必要性及正确方法；对由函数调用所引起的各种变量的生命期和作用域有清晰的了解；对递归函数的执行过程和工作原理有较好的理解，能编写一些简单的递归函数；掌握变量由于其不同存储形式所具有的不同意义及其使用场合。

教学内容：

- (1) 知识点一：◎函数定义的一般形式
- (2) 知识点二：◎函数的参数
- (3) 知识点三：◎函数的一般调用
- (4) 知识点四：○函数的嵌套调用和递归调用
- (5) 知识点五：※局部变量和全局变量
- (6) 知识点六：变量的存储类别

教学方法：

- (1) 讲解加课堂互动的方法，使学生掌握本章的重点和难点。
- (2) 线上与线下相结合，对重点难点内容要求学生练习相关习题。
- (3) 以函数调用为例，理论联系实际。
- (4) 多媒体课件和板书相结合。

7、知识单七：编译预处理和变量的存储类型（支撑课程目标 1、2）

教学要求：

掌握宏定义的基本概念，学会定义和使用宏定义，注意宏定义在程序中运行的步骤，宏展开的原样替换；掌握文件包含的基本概念，能够根据所调用的函数熟练的书写文件包含的代码；可安排学生自学了解条件编译的使用和意义。

教学内容：

- (1) 知识点一：宏定义

(2) 知识点二：文件包含处理

(3) 知识点三：条件编译

教学方法：

(1) 讲解加课堂互动的方法，使学生掌握本章的重点和难点。

(2) 线上与线下相结合，对重点难点内容要求学生练习相关习题。

(3) 以编译预处理为例，理论联系实际。

(4) 多媒体课件和板书相结合。

8、知识单元八：批量数据的处理-数组（支撑课程目标 1、2）

教学要求：

掌握数组的概念及使用数组的意义；掌握一维数组的定义方法和引用方式，理解数组的长度和数组下标的概念；掌握使用一维数组实现简单的排序算法；掌握二维数组的定义方法和引用方式，理解二维数组的内存存放形式；

教学内容：

(1) 知识点一：※一维数组的定义、引用及初始化

(2) 知识点二：○二维数组的定义、引用及初始化

(3) 知识点三：一维数组、二维数组的访问方法

(4) 知识点四：静态数组的定义和使用

(5) 知识点五：数组元素作为函数参数

教学方法：

(1) 讲解加课堂互动的方法，使学生掌握本章的重点和难点。

(2) 线上与线下相结合，对重点难点内容要求学生练习相关习题。

(3) 以批量数据处理为例，理论联系实际。

(4) 多媒体课件和板书相结合。

9、知识单元九：文本的处理-字符串（支撑课程目标 1、2）

教学要求：

掌握字符数组、字符串的定义和使用方法。能使用字符数组、字符串处理文本。

教学内容：

(1) 知识点一：字符数组的定义、初始化、使用

(2) 知识点二：字符数组与字符串的区别

(3) 知识点三：字符数组的输入、输出

(4) 知识点四：各字符串处理函数

教学方法：

- (1) 讲解加课堂互动的方法，使学生掌握本章的重点和难点。
- (2) 线上与线下相结合，对重点难点内容要求学生练习相关习题。
- (3) 以文本处理为例，理论联系实际。
- (4) 多媒体课件和板书相结合。

10、知识单元十：间接访问数据-指针（支撑课程目标 1、2）

教学要求：

熟练掌握指针和指针变量的概念，明确清楚两者的联系和区别；基本掌握用指针表示和引用一维和二维数组的元素，理解数组名做函数参数时是按照指针来处理的，能够定义返回指针值的函数，并会利用函数指针调用函数。

教学内容：

- (1) 知识点一：※地址和指针的概念
- (2) 知识点二：◎指针变量的定义和初始化
- (3) 知识点三：指针的运算
- (4) 知识点四：◎数组与指针
- (5) 知识点五：○指向函数的指针
- (6) 知识点六：返回指针值的函数

教学方法：

- (1) 讲解加课堂互动的方法，使学生掌握本章的重点和难点。
- (2) 线上与线下相结合，对重点难点内容要求学生练习相关习题。
- (3) 以间接访问数据为例，理论联系实际。
- (4) 多媒体课件和板书相结合。

11、知识单元十一：自定义数据类型-结构体、共用体和枚举类型（支撑课程目标 1、2）

教学要求：

掌握结构类型的定义方法，掌握对结构变量的成员访问的方式；

教学内容：

- (1) 知识点一：结构体类型的定义和使用
- (2) 知识点二：结构体变量的定义、初始化及引用
- (3) 知识点三：结构体数组的定义和使用
- (4) 知识点四：公用体类型变量的定义、结构体与共用体的区别
- (5) 知识点五：枚举类型的定义

(6) 知识点六：枚举类型变量的定义、初始化及引用

教学方法：

- (1) 讲解加课堂互动的方法，使学生掌握本章的重点和难点。
- (2) 线上与线下相结合，对重点难点内容要求学生练习相关习题。
- (3) 以自定义数据类型结构体为例，理论联系实际。
- (4) 多媒体课件和板书相结合。

12、知识单元十二：数据持久化-文件（支撑课程目标 1、2）

教学要求：

了解文件的存储结构。掌握文件操作的几大步骤：定义文件指针、打开、读写、关闭，选择合适的打开方式并选择合理的文件读写控制方式。了解文件随机访问的控制方法，掌握文件复制的多种实现方法。

教学内容：

- (1) 知识点一：文件的基本概念
- (2) 知识点二：※文件的打开和关闭
- (3) 知识点三：※文件的读写
- (4) 知识点四：※文件的定位

教学方法：

- (1) 讲解加课堂互动的方法，使学生掌握本章的重点和难点。
- (2) 线上与线下相结合，对重点难点内容要求学生练习相关习题。
- (3) 以文件处理为例，理论联系实际。
- (4) 多媒体课件和板书相结合。

符号备注：※重点、○难点、◎重点且难点

六、课程学时分配

本课程总学时 150，其中授课 32 学时，上机 32 学时，此外，本课程根据教学需要还设置了 86 个课外学时。

表 2：课程学时分配表

知识单元	教学环节 学时	课内总学时					课外学时	
		理论	上机	课程 实践	实验	线上 (翻转课堂)		小计
知识单元一：概述		2	2				4	
知识单元二：程序设计起步		4	4				8	
知识单元三：会思考的程序-选择结构		2	2				4	

教学环节 时数 知识单元	课内总学时						课外学时
	理论	上机	课程 实践	实验	线上 (翻转课堂)	小计	
知识单元四：键盘输入与屏幕输出	2	2				4	
知识单元五：周而复始-循环结构	4	4				8	
知识单元六：模块化设计-函数	2	2				4	
知识单元七：编译预处理	2	2				4	
知识单元八：批量数据处理-数组	4	4				8	
知识单元九：文本的处理-字符串	2	2				4	
知识单元十：间接访问数据-指针	4	4				8	
知识单元十一：自定义数据类型-结构体、共用体和枚举类型	2	2				4	
知识单元十二：文件	2	2				4	
总计	32	32				64	86

七、教学资源

1.教材与讲义

《C 语言程序设计教程》，谢旻，吕俊，张军强，上海交通大学出版社，2018 年

2.主要参考书

- [1] 《C 语言程序设计实训教程》，杨丽萍，宁以风，刘粉香，高等教育出版社，2012 年
- [2] 《C 语言参考手册（原书第 5 版）》，（美国）哈比森，机械工业出版社，2008 年
- [3] 《C 程序设计语言（第 2 版）》，（美国）Brian W.Ketnighan,Dennis M.Ritchie，机械工业出版社，2005 年
- [4] 《C Primer Plus（第五版）》，（美国）Stephen Prata 著，云巅工作室，人民邮电出版社，2013 年
- [5] 《明解 C 语言（第三版）》入门篇，（日本）柴田望洋 著 管杰 罗勇 杜晓静 译，人民邮电出版社，2015 年
- [6] 《C 语言程序设计实验指导》，李斌，张月琴，上海交通大学出版社，2017 年
- [7] 《C 语言程序设计教程（第 4 版）》，朱鸣华，罗晓芳，董明，孟军，汪德刚等，机械工业出版社，2019 年

八、学习要求与课程考核

1.学习要求

本课程是工程基础类课程，要求学生思想上重视本门课程，并按照老师要求结合线上资源积极完成

老师布置的任务和作业，课堂跟随老师的步骤，积极思考，多动手实验，提高自己分析问题和解决问题的能力。

2.课程考核方式

本课程考核方式为期末闭卷考试，无期中考试。

3.分项成绩评分标准与方法

(1) 平时成绩

平时成绩根据课后作业、课堂互动的情况并依据如下评分标准和方法评定，评分标准和方法如表 3 所示。

表 3：平时成绩评定标准和方法

平时成绩构成	优秀 (90~100)	良好 (80~89)	中等 (70~79)	及格 (60~69)	不及格 (<60)	所占比例
课程目标 1	能够很好地运用数学、C 语言程序设计方法表达反应工程问题，获得 C 语言的语言基础、分支、循环、函数、结构、指针、文件等方面的知识；能够很好地阅读和应用结构化程序设计方法设计、编写、调试和运行 C 语言程序；	能够较好地运用数学、C 语言程序设计方法表达反应工程问题，获得 C 语言的语言基础、分支、循环、函数、结构、指针、文件等方面的知识；能够较好地阅读和应用结构化程序设计方法设计、编写、调试和运行 C 语言程序；	运用数学、C 语言程序设计方法表达反应工程问题，获得 C 语言的语言基础、分支、循环、函数、结构、指针、文件等方面的知识；阅读和应用结构化程序设计方法设计、编写、调试和运行 C 语言程序；	基本能够运用数学、C 语言程序设计方法表达反应工程问题，获得 C 语言的语言基础、分支、循环、函数、结构、指针、文件等方面的知识；基本能够阅读和应用结构化程序设计方法设计、编写、调试和运行 C 语言程序；	运用数学、C 语言程序设计方法表达反应工程问题，获得 C 语言的语言基础、分支、循环、函数、结构、指针、文件等方面的知识的能力较差；阅读和应用结构化程序设计方法设计、编写、调试和运行 C 语言程序的能力较差；	40%
课程目标 2	能够很好地掌握结构化程序设计的思想和方法，掌握 C 语言的基本语法，具有分析问题的能力，并具有用 C 语言编程解决问题的能力；能较很好地掌握一种开发工具的使用，具有一定的使用调	能够较好地掌握结构化程序设计的思想和方法，掌握 C 语言的基本语法，具有分析问题的能力，并具有用 C 语言编程解决问题的能力；能较好地掌握一种开发工具的使用，具有一定的使用调	掌握结构化程序设计的思想和方法，掌握 C 语言的基本语法，具有分析问题的能力，并具有用 C 语言编程解决问题的能力；地掌握一种开发工具的使用，具有一定的使用调	基本能够掌握结构化程序设计的思想和方法，掌握 C 语言的基本语法，具有分析问题的能力，并具有用 C 语言编程解决问题的能力；基本能够掌握一种开发工具的使用，具有一定的使用调	掌握结构化程序设计的思想和方法，掌握 C 语言的基本语法，具有分析问题的能力，并具有用 C 语言编程解决问题的能力；基本能力较差；掌握一种开发工具的使用，具有一定的使用调	60%

平时成绩构成	优秀 (90~100)	良好 (80~89)	中等 (70~79)	及格 (60~69)	不及格 (<60)	所占比例
	试器查找问题并解决问题的能力；具有对简单问题进行分析、并设计算法，开发软件解决问题的能力；通过引导学生自学，初步培养学生适应大学学习，逐步培养学生的自学能力。	查找问题并解决问题的能力；具有对简单问题进行分析、并设计算法，开发软件解决问题的能力；通过引导学生自学，初步培养学生适应大学学习，逐步培养学生的自学能力。	题的能力；具有对简单问题进行分析、并设计算法，开发软件解决问题的能力；通过引导学生自学，初步培养学生适应大学学习，逐步培养学生的自学能力。	查找问题并解决问题的能力；具有对简单问题进行分析、并设计算法，开发软件解决问题的能力；通过引导学生自学，初步培养学生适应大学学习，逐步培养学生的自学能力。	有对简单问题进行分析、并设计算法，开发软件解决问题的能力；通过引导学生自学，初步培养学生适应大学学习，逐步培养学生的自学能力较差。	
平时成绩小计						100%

(2) 期末考试成绩

根据学生试卷实际应答情况评定。

4. 总评成绩评分方法

本课程总评成绩由平时成绩和期末考试成绩组成。总评成绩与课程目标的关系以及各项成绩占比详见表 4。

表 4：总评成绩构成

课程目标	总评成绩构成比例		合计分值
	平时成绩	期末考试成绩	
课程目标 1	10%	30%	40
课程目标 2	20%	40%	60
合计	30%	70%	100

九、说明

《C 语言程序设计》是一门实践性很强的工程基础类课程，其实践环节由《C 语言程序设计》课内上机等相关课程解决。

执笔人：

何春柳

审核人：

朱明松

批准人：

赵航

编制时间：2023 年 7 月

《C 语言程序设计》课内上机大纲

一、基本理论与技术知识

本课程上机环节任务是通过 C 语言环境掌握 C 语言的基本语法结构、编程思维方式，该课程强调实践过程。通过《C 语言程序设计》的上机实验，可以帮助学生巩固理论课上学习的语法和算法，培养和提高学生的动手编程能力，培养强调良好的编程风格，训练学生查错、改错与调试的技能，从而逐步建立编程思想。

本环节需要掌握如下的基本理论：

1. 掌握 C 语言程序设计语法知识，包括常量与变量、基本数据类型、运算符与表达式、流程控制、函数、指针与数组、动态内存分配、结构体、文件处理操作及预处理命令等；
2. 掌握结构化程序设计方法，具有分析问题，并使用 C 语言编程解决问题的能力；
3. 能够在编码过程中应用结构化程序设计的良好编码风格；
4. 能够使用 C 语言程序设计的集成开发工具，掌握基本的调试程序的知识，具有一定程度的使用调试器查找问题并解决问题的能力。

二、上机方法、特点与基本要求

教学基本要求：

1. 熟练掌握并且使用书中理论知识点：常量与变量、基本数据类型、运算符与表达式、流程控制、函数、指针与数组、动态内存分配、结构体、文件处理操作及预处理命令等。
2. 熟练使用 C 语言程序设计的集成开发工具，能调试程序，具有一定程度使用调试器查找问题并解决问题的能力。
3. 掌握结构化程序设计方法，具有分析问题的能力，并具有用 C 语言编程解决问题的能力；掌握常用的算法设计，如排序算法和链表的操作。
4. 通过课内实验程序的编写和调试，用实验结果证明算法的正确性，培养严谨的治学态度，同时在互相讨论、解决问题的过程中培养团队协作精神。

三、主要仪器设备

软件：dec++、

硬件：电脑

四、项目/环节设置与内容提要

本课内实验部分具体项目设置如表 1 所示。

表 1：项目/环节设置情况

序号	支撑课程目标	项目名称	内容提要	学时	上机类型	上机要求	每组建议人数
1	课程目标 1	C 程序开发环境	熟悉 DevC 的使用环境, 创建一个简单 C 程序。	2	演示	必做	1
2	课程目标 1	常量及变量	练习 int、float、char 三种基本类型的变量定义和赋初值操作；	2	验证	必做	1
3	课程目标 1	表达式	把代数式转化为 C 语言的合法表达式。	2	验证	必做	1
4	课程目标 1	选择结构程序设计	练习 if、if..else、if..else if...else 语句；应用 switch 书写多分支程序。	2	设计	必做	1
5	课程目标 1	键盘输入与屏幕输出	练习 scanf 语句和 printf 语句实现变量的输入和输出。	2	验证	必做	1
6	课程目标 1	循环结构程序设计一	练习 while、do while、for 语句的基本应用。	2	设计	必做	1
7	课程目标 1	循环结构程序设计二	应用掌握枚举法和迭代法的一般编程思想, 理解 while、do while 语句和 for 语句的适用场合；掌握二维循环的应用。	2	设计	必做	1
8	课程目标 2	函数设计	练习函数的声明、定义和调用。	2	设计	必做	1
9	课程目标 1	编译预处理与变量的存储类型	练习宏定义, 宏展开和条件编译	2	设计	必做	1
10	课程目标 1	一维数组程序设计	熟悉一维数组的基本使用方法。	2	设计	必做	1
11	课程目标 1	二维数组程序设计	熟悉二维数组的应用；掌握字符数组的定义和使用。	2	设计	必做	1
12	课程目标 2	字符串	掌握字符数组、字符串的定义和使用方法。能使用字符数组、字符串处理文本。	2	验证	必做	1
13	课程目标 2	指针一	验证如何通过指针实现间接访问；定义形参为指针的函数, 通过指针变量形参改变主调函数中变量的值, 体会传址的特殊应用。	2	验证/设计	必做	1
14	课程目标 2	指针二	验证指针访问一维数组和二维数组的方式。	2	验证/设计	必做	1
15	课程目标 2	自定义数据类型	验证结构体变量的定义和引用；通过学生结构体数组, 验证并体会结构体数组的应用。	2	验证	必做	1
16	课程目标 2	文件操作	创建一个文件, 实现对文件的读写操作。	2	验证/设计	必做	1

五、上机报告要求

1、画好流程图, 完成程序纸质编码；上机要求程序编译通过, 运行结果正确；

2、测试数据要全面，注意边界测试，并涵盖所有路径；

3、源代码书写要符合良好的编码风格。

六、课程考核与成绩评定

1.考核方式

本课内实践部分考核方式为考查，按百分制计分。

2.评分标准与方法

本课内实践部分评分方法如表 2 所示。

表 2 评分方法

序号	支撑课程目标	项目名称	成绩构成比例	考核/评价细则
1	课程目标 1	项目 1-7、项目 9-11	40%	
2	课程目标 2	项目 8、项目 12-16	60%	
成绩合计			100%	

执笔人：何春柳

审核人：

朱明松

批准人：

赵航

编制时间：2023 年 7 月