

# 《C 语言程序设计(C Language Programming)》

## 教学大纲

制定时间：2024 年 4 月

### 一、课程基本信息

(一)适用专业：理工类各专业

(二)课程代码：3DX1239A

(三)学分/课内学时：3 学分/48 学时

(四)课程类别：基础课程

(五)课程性质：必修/理论课

(六)先修课程：大学计算机基础、高等数学

(七)后续课程：微机原理及应用、C 语言程序设计课程设计、毕业设计等。

### 二、课程教学目标

使学生能理解数据和算法的基本概念，数据是问题的实例，算法是为求解问题所设计的求解步骤，求解步骤的 C 语言语句表达就是 C 语言程序。接受并掌握 C 语言的数据规则、运算规则和语法规则是学习本门课程的基本任务。本课程是为理科、工科各专业开设的计算机技术基础通识必修课程。课程的主要任务是引导学生掌握一门计算机程序设计语言，培养学生逻辑思辨能力、抽象能力和基本的程序设计能力、代码调试能力、程序数值测试能力，了解计算机处理问题的方式方法，为更好地使用计算机处理专业领域的各类问题奠定基础。具体细化为：

#### (一)具体目标

**目标 1：**掌握 C 语言的基础知识、语法规则，能将其用于专业领域中的数据表达和数据处理等问题。

**目标 2：**掌握结构化程序设计方法，能针对专业领域的任务需求设计求解步骤，编写调试出满足功能要求的 C 语言程序。

**目标 3：**能够使用主流的 C 语言程序开发工具。

**目标 4：**能选择一种恰当的 C 语言程序开发工具，完成代码的编辑、编译、调试和测试。

#### (二)课程目标与毕业要求的对应关系

毕业要求	毕业要求指标点	课程目标	教学单元	评价方式
1.工程知识：掌握本专业必需的数学、自然科学、工程基础和专业知识，能够用于	指标点 1.3：能将数学、工程基础和专业知识用于解决计算机控制	目标 1	C 语言的数据类型 表达式与数	平时作业 实验 试卷考核

解决自动化系统中的复杂工程问题。	系统的软硬件分析与设计问题。		值计算 算法设计 结构体、共用体和枚举型	
<b>3.设计/开发解决方案:</b> 能够设计针对自动化系统设计中的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)或控制流程,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。	<b>指标点 3.3:</b> 能够进行自动化系统的硬件选型设计、电气接线图设计、硬件配置和控制程序设计	目标 2	结构化语句 数组 函数 指针 文件操作	平时作业 实验 试卷考核
<b>5.使用现代工具:</b> 能够针对自动化系统的复杂工程问题,分析、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,实现对复杂工程问题的预测与模拟,并能够理解其局限性。	<b>指标点 5.1:</b> 了解常用的电路与电子、自动控制、计算机软硬件工具	目标 3	C 语言程序设计概述	随堂讨论
	<b>指标点 5.2:</b> 能够针对自动化领域复杂工程问题,分析与选择恰当的电路与电子、自动控制、计算机软硬件等技术和工具	目标 4	实验	

### 三、教学内容与方法

#### (一)教学内容及要求

序号	教学单元	教学内容(知识点)	学习产出要求	推荐学时	推荐教学方式	支撑教学目标	备注
1	C 语言的数据类型表达式与数值计算算法设计结构体、共用体和枚举型	C 程序开发环境; C 源程序文件名; C 程序的目标文件名(obj); C 程序的可执行文件(exe); 计算机系统、计算机操作系统、计算机软件系统。	1.C 语言程序的集成开发环境说明。VC6.0、VC2010、Dev-C++、CodeBlocks 等主流 C 程序开发工具。 2.简单且完整的 C 程序编程演示。录入 C 代码的源文件、代码编译的目标文件、连接的可执行文件。 3.针对目标代码说明与 CPU 的关系。 4.针对可执行文件(exe)说明与操作系统的关系。 5.扼要说明计算机硬件系统、操作系统、软件系统三者关系。	2	讲授和演示	目标 3	讲授与演示

序号	教学单元	教学内容 (知识点)	学习产出要求	推荐学时	推荐教学方式	支撑教学目标	备注
2	实验一	认识 C 源程序的录入、代码纠错、编译和链接过程；	1.掌握 C 源代码录入； 2.掌握 C 源程序的编译与错误提示和纠正操作方法； 3.感受 C 源程序的编译产生的目标文件 obj； 4.感受连接产生的可执行文件 exe，在系统控制台下的运行。	2	教师快速演示。学生再进行独立操作。	目标 4	验证性实验
3	C 语言的数据类型表达式与数值计算算法设计结构体、共用体和枚举型	char;int;long;double;unsigned;整型数；实型数；整数补码；实型数的阶码；实型数的精度；整型常量表达形式；实型常量的表达形式。	1.掌握整型数关键：char,int,long,float,double 作用，它们与 unsigned 结合后叫无符号整型数。 2.掌握整型数的补码。原码、反码、求反+1 规则。为什么整型数在系统中要用补码表示。 3.掌握实型数关键字：float,double 的作用，实型数精度。 4.各种类型数据占用内存长度。 5.各种类型数据常量的表达形式。 6.char 型数据的两个属性。	2	讲授与演示并行	目标 1	教师要设置断点演示，让学生有直观认识
4	C 语言的数据类型表达式与数值计算算法设计结构体、共用体和枚举型	变量；变量名与内存及其内存长度；常量；常量的表达形式及其存在位置；C 运算符及其运算优先级；C 表达式与表达式的值。	1.变量，变量的三要素：变量数据类型、变量名和变量的值。变量名用户给一段内存存取的名字；变量名的命名规则。 2.常量与常量的表达形式，常量存在于程序代码段中，所以称之为常量。 3.算术运算符与运算规则。 4.关系运算符与运算规则。 5.逻辑运算符与运算规则。 6.各运算符的运算优先级，一般要掌握算术运算符高于关系运算符，高于逻辑运算符。 7.表达式。运算符和操作数组的算式称为表达式；表达式强调的是数值计算，任何表达式的结果为一个数值。 8.关系表达式和逻辑表达式的结果，要么为 0，要么为 1。	4	讲授与演示结合	目标 1	必做验证性

序号	教学单元	教学内容 (知识点)	学习产出要求	推荐学时	推荐教学方式	支撑教学目标	备注
5	实验二	算术表达式求值; 关系表达式求值; 逻辑表达式求值	1.假定变量, 常量已存在, 并赋有初值。 2.实验内容中指定: 算术表达式、关系表达式、逻辑表达式。表达式中应包含变量、常量。 3.人工分析各表达式的值。 4.机器上调试并执行程序。人工计算与机器计算进行结果对比。	2	在教师的监督指导下, 学生独立操作	目标 4	验证性实验
6	C 语言的数据类型表达式与数值计算算法设计结构体、共用体和枚举型	数据类型与输入输出控制符; %d %c %f %lf %e %g, %s; 转义符及其意义。	1.整型数(int)十进制输入输出控制符%d; 2.单精度(float)输入输出控制符%f; 3.双精度(double)输入输出控制符%lf; 4.字符型(char)的输入输出控制符%c,当字符型数据作为整型数时的输入输出控制符%d; 5.实型数(float,double)输入输出控制符%e,%g; 6.转义符。\\称为转义符, 它转紧跟后面普通字符的义, 将其转义为控制字符; 7.转义符的 8 进制或 16 进制数的表达。 8.字符串的输入输出控制符%s;	2	讲授与演示结合	目标 1	
7	实验三	%d 对应数据类型 int,long,c har; %c 对应 char %f 对应 float 输入; %lf 对应 double; %e,%g 对应 float 和 double	1.int 型数据%d 格式输入与输出。 2.float 型数据%f 格式输入 3.double 型数据%lf 输入 4.float 和 double 数据的%f,% e,%g 输出。 5.char 型数据的%d 与%c 的输入和输出	2		目标 4	验证性实验
8	结构化语句 数组 函数 指针 文件操作	选择语句: if(表达式)单 选; if 与 else 实现 双选一; if 与 else 配对 实现多选一;	1.if(表达式)实现单选判断; 2.if()else 实现双选一情况; 3.if()与 else 配对实现多选一的情况; 4.if 与 else 不成对的配对关系分析。 5.switch(整型表达式)。强调	2	讲授与演示结合	目标 2	

序号	教学单元	教学内容 (知识点)	学习产出要求	推荐学时	推荐教学方式	支撑教学目标	备注
		开关语句：switch 与 case、default; 相关语句 break;	表达式算出的值是一个整型数，它与 case 的关系，以及 default 的意义。 6.switch 语句中使用 break 的作用。				
9	实验四	ifelse; switch case break; default;	1.if 的单选情况; 2.if 的双选一情况; 3.if 的多选一情况; 4.switch 开关语句的使用。	2		目标 4	验证性实验
10	结构化语句 数组 函数 指针 文件操作	for 循环语句; while 循环语句; do{}while 循环语句; continue 语句; break 语句; 当型循环; 直到循环;	1.for(初始表达式; 循环条件表达式; 步长表达式)与循环体才能称为 for 循环语句，它是一个整体; 2.for 语句的执行过程说明: 初始表达式是循环开始时执行，以后不执行。循环条件表达式每次都要执行，当其值为 0 时停止循环语句结束循环，为非 0 时执行循环体，一次循环结束。开始下一次循环首先执行步长表达式，再执行循环条件表达式，如此反复执行。 3.while(表达式)循环体语句。当表达式为 0 时停止循环。当表达式为非 0 时，执行循环体，循环体内一定要实现循环变量的改变，否则将导致死循环。 4.do{循环体}while(条件表达式);循环体首先得到执行，然后再执行循环条件表达式，当表达式为非 0 时转向 do 执行循环体，当表达式值为 0 时循环结束。循环的每一步要求循环变量必须改变，否则会导致死循环。 5.当型循环与直到循环。当条件成立时执行循环，当条件不成立时结束循环。For 与 while 属当型循环，do{}while();为直到循环。 6.continue.当循环体内使用 continue 语句时，它的作用是结束本次循环，进入下一次循环。 7.break 当循环体内存在 break	4	讲授与演示结合	目标 2	

序号	教学单元	教学内容 (知识点)	学习产出要求	推荐学时	推荐教学方式	支撑教学目标	备注
			k 语句时，它的作用强制循环结束。 8.双重循环。				
11	实验五	for; while; dowhile	1.常规形式的 for 循环表达式形式。 2.for 循环表达式缺省情况 3.常规形式的 while 循环表达式形式。 4.while(1)情形下对循环体的要求。 5.常规形式的 dowhile 循环表达式形式。 6.循环体中使用 continue 情况 7.循环体使用 break 的情况。	2		目标 4	验证性实验
12	结构化语句 数组 函数 指针 文件操作	数组概念; 一维数组定义格式; 二维数组定义格式; 数组元素下标(元素序号); 数组元素的引用(元素值的读/写); 数组名为地址常量; 数组占用内存长度计算	1.数组的概念：有限个相同类型数据的有序集合。 2.一维数组、二维的定义方法。 3.数组元素与下标(元素序号), 二维数组元素下标表达式方式。 4.数组名是地址常量, 数组名是用户给一段连续内存的名字, 数组名是这段内存的起始地址。 5.数组占用内存的长度计算与 sizeof 运算的使用。 6.数组定义并赋初值, 初值表。用字符串常量给字符数组赋初值。 7.数组元素的引用。引用即是读/写数组元素值。 8.字符数组与字符串。当字符数组中存在 0 值元素时可称为字符串, 不存在 0 值元素只能称为字符数组。数值 0 即'\0'是字符串的结束标识。	4	讲授与演示结合	目标 2	
13	实验六	一维数组 二维数组	1.一维数组和二维数组定义不赋初值时,输出各元素值。 2.一维数组和二维数组定义并赋初值, 输出各元素值。 3.通过数组下标引用数据元素。 4.计算数组占用内存长度 5.采用选择法对数据元素排序。	2		目标 4	验证性实验

序号	教学单元	教学内容 (知识点)	学习产出要求	推荐学时	推荐教学方式	支撑教学目标	备注
			6.用字符串常量给 char 型数据赋值,并采用%s 输出数组 7.使用二维数组实现多个字符串排序。				
14	结构化语句 数组 函数 指针 文件操作	函数的概念;	1.从数学函数引入 $y=f(x_1,x_2)$ , $f$ 为法则, $x_1,x_2$ 为自变量; 2.C 函数概念。具有特定功能的程序块。 3.函数的定义格式: 函数类型 函数名(形式参列表){函数体} 4.函数类型与返架回值。 5.主调函数与被调函数间,实参与形参是单向值的传递。 6.全局变量、局部变量、变时的作用域和生存期 7.递归函数: 函数体内存在自身调用称为递归函数。递推、折返点、回归过程及其数据变化。	2	讲授与演示结合	目标 2	
15	实验七	函数的定义; 实参与形参; 局部变量; 全局变量; 主调函数; 被调函数; 递归函数;	1.根据功能需求定义函数。 2.掌握主调函数与被调函数,实参与形参,局部变量与全局变量。 3.递归函数的定义与调用。	2		目标 4	验证性实验
16	结构化语句 数组 函数 指针 文件操作	数据指针; 指针数组; 指针作为形参; 数组指针; 指针函数; 函数指针;	1.指针即是地址,地址不是指针。 2.数据指针的定义格式。int *,long*,char*,float*,double*。 3.使用指针引用数据对象。 4.指针数组。double*adrs[10]。 5.数组指针的定义格式。用于指向二维数组的行指针。 6.指针偏移量的计算。 7.指针作为函数形参分析。 8.指针函数。函数类型为指针,即函数返回指针。 9.函数指针,用于指向相同定义格式函数的指针。 10.操作字符串的系统函数。	4	讲授与演示结合	目标 2	
17	实验八	数据指针; 指针数组; 数组指针; 指针作为形	1.体验数据指针(intlongfloat double 和 char); 2.用数据指针遍历一维数组和二维数组级元素。	2		目标 4	验证性实验

序号	教学单元	教学内容 (知识点)	学习产出要求	推荐学时	推荐教学方式	支撑教学目标	备注
		参; 指针函数; 函数指针;	3.指针偏移计算, 单位数据长度。 4.数组指针, 用于指向二维数组行的指针; 5.指针函数, 函数返回指针的函数; 6.函数指针, 指向函数的指针, 函数名也是地址常量。 7.字符串函数 strcpy, strcat 等函数的使用。				
18	C 语言的数据类型表达式与数值计算算法设计 结构体、共用体和枚举型	结构体 struct; 共用体 union 枚举型 enum	1.结构体概念: 把不同类型数集成(封装)在一起构成一种新的数据类型。 2.使用 struct 构造数据类型的格式要求。 3.结构体变量, 结构体数组。及其占用内存长度分析。 4.结构体对象(变量或数组)的数据成员引用形式。 5.结构体数据指针引用数据成员形式。 6.共用体 union, 把不同类型数据集成在一起共享相同内存。 7.共用体举例分析说明。 8.枚举型 enum, 已知整型常量符号化。已知常量与符号关系举例分析	2	实例讲授与演示结合	目标 1	
19	实验九	构造数据类型应用实验	1.指定数据成员, 自己构造数据类型。 2.通过结构体变量引用数据成员(.), 通过结构指针引用数据成员(->). 3.结构体数组。 4.共用体体验。 5.枚举型体验。	2		目标 4	综合性实验
20	结构化语句 数组 函数 指针 文件操作	文件概念 缓冲文件系统;	1.文件。写入存储媒介上的数据集合称为文件。 2.操作系统缓冲文件系统。 3.文件指针的定义形式 FILE *。 4.文件打开方式。fopen 函数参数说明与举例。 5.从文件中读取数据, fread 函数的使用。 6.向文件写入数据。fwrite 函数的使用。 7.文件指针的移动。	2	实例讲授与演示结合	目标 2	



序号	教学单元	教学内容 (知识点)	学习产出要求	推荐学时	推荐教学方式	支撑教学目标	备注
			8.文件关闭。fclose 函数使用。 9.其它专用函数使用选择举例。				
21	实验十	文件读写实验	1.打开一个存在的文本文件，读出数据并显示。 2.改变指定位置的文本数据。 3.追加部分文本。 4.关闭文件。 5.使用记事本打开被修改过的文本文件。 6.自己把一个结构体数组写入文件，再从文件读入数组。	2		目标 4	综合性实验

## (二)教学方法

### 1.课堂讲授

(1)采用启发式教学，激发学生主动学习的兴趣，培养学生独立思考、分析问题和解决问题的能力，引导学生模仿教师演示的源程序获感性认识，教师再以引导，让学生掌握 C 语言比较基础的知识点、熟悉重要的知识点、了解复杂的知识点。

(2)教学内容，系统讲授：①C 语言的数据类型及其关键字 char、int、long、float、double 它们是构成 C 程序的数据基础，不同类型数据占用内存的长度不同，其中 char、int、long 在内存中的计数方式的是补码，而实型数 float 和 double 是阶码。②不同类型数据的输入与输出。尤其要重视 char 型数据的输入与输出。③C 语言的表达式。表达式是 C 程序的计算单元，任何一个表达式的计算结果为数值，表达式的数值只有两种 0 和非 0，非 0 为真，0 为假。表达式的称谓是由表达式中运算符优先级最低的运算符确定的，运算优先级最低的最后运算。④结构化语句。顺序结构、分支结构、循环结构，以及对应语句的执行步骤讲解。⑤一、二维数组。强调数组的概念，数组元素的位置，数组占用内存长度的计算。⑥函数。应当与数学函数  $y=f(x)$  进行对照讲解。被调函数，主调函数，实际参数与形式参数是单向值的传递。⑦指针。指针即是地址，教会学生按内存地址查找数据，指针偏移单位长度与数据类型相关；指针的称谓，int 型指针、char 型指针、long 型指针、float 型指针、double 型指针、指针数组、数组指针、函数指针、指针函数。⑧结构体。不同类型数据集成在一起构成一种新的数据类型称为结构体，以实例演示构造数据类型，以及使用.和->运算符引用结构体数据成员的实例；共用体，把不同类型数据封装在一起共享一段内存。以实例演示；枚举型。已知整型常量符号化。⑨文件。保存在存储介质上的数据集合称为文件。C 操作文件的方法演示与讲解。文件指针 FILE\*、打开或创建文件的函数 fopen、从文件中读取

数据函数 fread、向文件写入函数 fwrite 以及其它文件数据读写的专用函数，停止操作文件数据的函数 fclose。

通过这些内容的讲解与演示，使学生比较全面的了解《C 语言程序设计》的基础，认识 C 语言数据处理的数据规则和语法规则，按规行事。

(3)在教学过程中采用电子教案、多媒体教学与传统板书、教具教学相结合，增强教学的直观性，提高课堂教学过程中学生对知识点的吸收率。

(4)理论教学与实验相结合，引导学生应用数学、物理学、电工电子学的公式进行数值计算，培养学生提出问题，从已知条件着手设计求解问题的步骤，用 C 语言程序去求解问题。

(5)课外答疑，每周至少一次进行答疑，可以在微信群或 QQ 群中进行。

## **2.实验教学**

实验教学是学习掌握《C 语言程序设计》课程的关键环节，包括调试 C 源程序工具的使用，C 源代码的录入与编辑，语句错误的查找与纠正，最终的 exe 可执行文件的数据测试。通过具体的、针对知识的实例编程才能让学生吸收知识点。

实验不能少于 8 个，教师可针对知识设计 10 个实验，也可以指定专门的实验指导教材。

## **3.作业**

作业是反映学生吸收率的重要一环。学生必须按教学进程在“C 语言设计实训与考试系统”中完成规定的作业，促进学生消化吸收知识点。教师在后台浏览学生作业进展情况和答题数据，综合评估出学生掌握知识情况。

## 四、考核及成绩评定

### (一)考核内容及成绩构成

课程考核以考核学生能力培养目标的达成为主要目的，以检查学生对各知识点的掌握程度和应用能力为重要内容，包括平时考核和期末考核两个部分。平时考核采用平时作业、实验、方式评定学生成绩；期末考核采用机试或笔试评定学生成绩。《C语言程序设计》课程目标的考核内容、成绩评定方式、目标分值建议如下：

课程目标	考核内容	成绩评定方式	成绩占总评分比例	目标成绩占当次考核比例	学生当次考核平均得分	目标达成情况 计算公式
<b>目标 1:</b> 掌握 C 语言的基础知识、语法规则，能将其用于专业领域中的数据表达和数据处理等问题。	char,short,int,long,float,double 数据常量与变量、结构体、共用体和枚举型、表达式	作业	5%	100%	$A_1$	$\frac{\left(\frac{A_1}{100\%} \times 5\%\right) + \frac{B_1}{30\%} \times 25\%}{30}$
		考试	25%	30%	$B_1$	
<b>目标 2:</b> 掌握结构化程序设计方法，能针对专业领域的任务需求设计求解步骤，编写调试出满足功能要求的 C 语言程序。	掌握算术表达式、关系表达式和逻辑表达式数值计算的方法和步骤；结构化语句；数组、结构体和共用体集成数据的方法；函数与程序功能模块；指针与数据和函数；文件与数据。	作业	5%	100%	$A_2$	$\frac{\left(\frac{A_2}{100\%} \times 5\%\right) + \frac{B_2}{70\%} \times 45\%}{50}$
		考试	45%	70%	$B_2$	
<b>目标 4:</b> 能选择一种恰当的 C 语言程序开发工具，完成代码的编辑、编译、调试和测试。	基本数据类型、表达式、结构化语句、数组、函数、指针、文件操作	实验	20%	100%	$A_3$	$\frac{\left(\frac{A_3}{100\%} \times 20\%\right)}{20}$
总评成绩(100%)=平时作业(10%)+实验(20%)+机试(70%)			100%	—	—	$\frac{\text{学生总评平均分}}{100}$

## (二)平时考核成绩评定

### 1.作业:

作业 8 次,支撑目标 1 和目标 3,共占总评分 10%。目标 1 占 5%,目标 3 占 5%。  
对应目标的评分标准如下:

<b>对应目标</b>	<b>目标 1:</b> 掌握 C 语言的基础知识、语法规则,能将其用于专业领域中的数据表达和数据处理等问题。	<b>目标 2:</b> 掌握结构化程序设计方法,能针对专业领域的任务需求设计求解步骤,编写调试出满足功能要求的 C 语言程序。	
<b>考查点</b>	char,short,int,long,float,double 基本数据类型; 数据常量与变量(整型常量表达、实型常量表达;整型数补码的意义、实型数的阶码,整型数和实型数占用内存长度) 表达式(算术表达式、关系表达式、逻辑表达式、赋值表达式、逗号表达式) 结构体、共用体和枚举型(使用 struct、union 和 enum 构造数据,以及构造数据类型的使用)	算术表达式、关系表达式和逻辑表达式数值计算的方法和步骤; 结构化语句(if...else、switch、for、while、do...while 三个循环语句,当型和直到两种循环,continue、break); 数组、结构体和共用体集成数据的方法; 函数与程序功能模块; 指针与数据和函数; 文件与数据。	
<b>成绩比例</b>	5%(机器评阅)	5%(机器评阅)	
<b>评分标准</b>	100%至90%	能正确完成针对考查点的习题。	能正确完成针对考查点的习题。
	89.9%至75%	能较好完成针对考查点的习题。	能较好完成针对考查点的习题。
	74.5%至60%	基本能完成针对考查点的习题。	基本能完成针对考查点的习题。
	59.9%至0	基本不能完成针对考查点的习题。	基本不能完成针对考查点的习题。

### 2.实验:

必做实验 8 次,支撑目标 4,共占总评分 20%。学生撰写的实验报告文档提交到实验报告管理系统,教师在办公计算机批阅报告并评定成绩。教师就报告的规范性、报告内容的完整性、代码调试和数据测试结果进行综合评审。对应目标的考查点和评分标准如下:

<b>对应目标</b>	目标 4: 能选择一种恰当的 C 语言程序开发工具,完成代码的编辑、编译、调试和测试。
<b>考查点</b>	了解 C 语言程序的开发工具。熟练使用一种开发工具编写、编译调试 C 语言程序,并能使用数据测试程序的可靠性和稳定性。 应用 C 语言的数据规则表达数据,应用 C 的运算规则表达数值计算式,应用 C 的语法规则编写程序。应用 C 语言构造数据规则,构造出满足自己需要的新的数据类型。 针对实验题目需求,从已知条件着手理逻辑关系,设计出求解算法,编写 C 语言程序程序,在开发环境中编译调试,直至完整的对可执行文件的数据测试。 撰写规范的,内容完整的实验报告,报告文档提交到指定的接收机。
<b>成绩比例</b>	20%(人工评阅)

<b>评分 标准</b>	100%至 90%	能正确完成针对考查点的习题.
	89.9%至 75%	能较好完成针对考查点的习题.
	74.5% 至 60%	基本能完成针对考查点的习题.
	59.9%至 0	基本不能完成针对考查点的习题.

## 五、参考学习资料

### (一)推荐教材:

陈应祖,《C 语言程序设计基础教程》(第 1 版),人民邮电出版社。

谭浩强,《C 程序设计》(第四版),清华大学出版社。

何钦铭、颜晖,《C 语言程序设计》(第 3 版),高等教育出版社,2015 年 8 月。

### (二)实验指导书:

重庆科技大学 CExstudent.exe 线上练习。

自编实践教学。

### (三)课程资源中心:

<http://cqust.fy.chaoxing.com/portal>

制订人: 陈应祖

审核人: 何兰

参与人: 陈应祖、何兰、郭长金、赖军辉、徐显秋、张渝珩、张鹏

制定时间: 2024.4.10