**《程序设计基础》教学大纲**

**课程名称（汉）：**《程序设计基础》

**课程名称（英）：**Programming Fundamentals

**课程代码：**262101031

**课程类型：**专业基础课程模块

**适用对象：**自动化、机器人工程、人工智能专业大一年级本科生

**学时/学分：**80学时（理论48学时、 实验32学时）/4学分

**先修课程：**

**方案版本：**2021版

**一、课程简介**

C语言有丰富灵活的数据类型与运算，方便构建实用的数据结构，其简洁高效的语法表达，清晰的程序结构，能方便描述各类算法与编程思想，是现代很多高级编程语言的基础，本课程是为自动化、机器人工程、人工智能专业等本科生专业设计的一门学科基础教育课程，是学生掌握程序设计的必修课程。

通过本课程的学习，使学生掌握结构化的程序设计方法，模块化的分工机制，软件编制的原理和流程；培养学生严谨的程序设计思想、灵活的计算思维方式及较强的动手能力；并以此为基础，拓宽专业研究思路，让学生逐渐掌握其他软件的开发使用方法。

**二、课程的教学目标**

**课程目标1：**

（1）理解模块化、结构化程序设计思想；

（2）掌握C语言的数据类型、语句格式、函数结构等基本内容；

（3）熟悉C语言的环境和常规调试方法；

**课程目标2：**

（1）能用C语言进行结构化的程序设计；

（2）解决一些简单的应用问题；

（3）以此为工具进一步研究一些控制手段和控制技术；

**课程目标3：**

（1）初步具备辩证思维的能力；

（2）具有热爱科学，实事求是的学风和创新意识、创新精神；

（3）加强职业道德意识、安全规范意识。

**三、课程目标对自动化专业毕业要求的支撑关系**

| **毕业要求** | **毕业要求指标点** | **课程目标** | **支撑权重** |
| --- | --- | --- | --- |
| 工程知识 |  | 1、2 |  |
| 使用现代化工具 |  | 1、2、3 | 0.08 |

**四、教学内容及其基本要求**

| **序号** | **教学内容** | **教学要求**（包括重点难点、教学方法、学生掌握的程度等） | **对应课程目标** |
| --- | --- | --- | --- |
| 1 | 为什么要学C语言 | 【学习目的和要求】  1.知识掌握  理解学习C语言的意义，明确学习目的。  2.能力培养：软件设计的整个过程。  3.素质要求：黑客精神、网络道德、信息安全。  4.教学方法  兴趣驱动和案例驱动教学，演示“九九乘法表”、“猜数游戏”两个程序（游戏），通过演示，调动学生的学习兴趣和主动性；结合实例，讲解VC++6.0 环境的使用方法；介绍编程的基础知识及部分数据类型，以及整个软件开发的过程。  【重点】VC++6.0编程环境的使用。  【难点】如何带动学生的学习兴趣，让他们把该课程当作玩游戏一样来学。 | 课程目标1 |
| 2 | C数据类型 | 【学习目的和要求】  1.知识掌握：常量与变量、基本的数据类型、简单的屏幕输出。  2.能力培养：培养学生的计算思维，计算机与人的表达方式的不同。  3.素质要求：生活中的问题如何用计算机解决。  4.教学方法  赋值号的使用，与数学公式中的区别  在VC环境中编制小程序，完成交换两个变量的值，从而理解C语言的计算过程。  【重点】计算规则与数学中的区别。  【难点】改变以往的思维习惯，按计算机思维解决问题。 | 课程目标1 |
| 3 | 简单的算术运算和表达式 | 【学习目的和要求】  1.知识掌握：算数运算符的种类、算数表达式的计算规则。  2.能力培养：  培养学生的计算思维，计算机与人的表达方式的不同。  3.素质要求：理解数学计算中公式书写习惯与计算机表达习惯的不同。  4.教学方法  提问部分同学，回答预习题目；  【重点】计算规则与数学中的区别。  【难点】改变以往的思维习惯，按计算机思维解决问题。 | 课程目标1 |
| 4 | 键盘输入和屏幕输出 | 【学习目的和要求】  1.知识掌握：键盘输入与结果输出。  2.能力培养：人机交流的互动式操作。  3.素质要求：用固定格式输入输出，理解计算机的严谨格式，何处必须遵守，何处可自己自由发挥。  4.教学方法：  引入三个现实生活中的实例，说明三种结构的特点：  【重点】输入输出格式的语法结构。  【难点】输入格式的设置。 | 课程目标1、2 |
| 5 | 选择控制结构 | 【学习目的和要求】  1.知识掌握：  熟练掌握if语句三种形式（if、if-else、if-else if）的特点及使用方法；掌握switch语句的结构、特点及用法；掌握分支结构程序设计的基本方法，能熟练设计分支结构程序。  2.能力培养：  培养学生的选择思维，习惯计算机的表达方式。  3.素质要求：生活中的选择问题如何用计算机解决。  4.教学方法  讲解“二者选一”的简单if结构的用法。  【重点】单边if；双边if；多边if；switch结构。  【难点】改变以往的思维习惯，按计算机的选择思维和数学的算法解决问题。 | 课程目标1、2、3 |
| 6 | 循环控制结构 | 【学习目的和要求】  1.知识掌握：熟练掌握三种循环控制语句（while、do while、for）的结构特点及使用方法；掌握循环程序设计方法，能利用三种循环控制语句设计循环程序；掌握break和continue语句的用法；掌握多重循环程序的结构特点，能设计简单的二重循环程序。  2.能力培养：通过多次重复计算解决实际问题。  3.素质要求：计算机的优势在于运算速度快，反复的重复计算只是消耗内存和时间，计算机性能许可的范围内，循环均可实现。  4.教学方法  以1+2+……+10为案例，讲解while语句、for语句、do……while循环的语法结构和使用中需注意的问题。  应用举例：穷举法、公式计算、素数问题等。  【重点】三种循环结构； for中三个表达式的计算顺序；循环嵌套；break与continue。  【难点】空语句的误用；复合语句的使用；素数问题，变形的素数问题求解，实质是利用循环的语法知识+break实现。 | 课程目标1、2、3 |
| 7 | 函数 | 【学习目的和要求】  1.知识掌握：  掌握自定义函数的一般结构及函数的定义方法；掌握形参、实参、函数原型等重要概念。掌握函数声明、函数调用的一般方法；了解局部变量、全局变量和变量的存储类型的概念，熟悉auto型和static型局部变量的特点和用法；熟悉函数嵌套、函数递归的概念。  2.能力培养：  学会模块化程序设计的思想。  3.素质要求：大事化小、小事化了的思想（事无巨细、事必躬亲如诸葛亮，是不可取的，借此引出模块化的思想）。  4.教学方法  标准函数（例如：数学函数sqrt（x）），由系统提供，编程时写上#include <math.h> 后就可以直接使用；  局部变量、全局变量及其不同，动态存储变量与静态存储变量以及它们的使用方法。  【重点】参数值传递和地址传递的区别及本质。  【难点】函数的嵌套和递归。 | 课程目标1、2 |
| 8 | 数组 | 【学习目的和要求】  1.知识掌握：  熟练掌握一维数组的定义、初始化及其使用方法；掌握二维数组的定义、初始化及其使用方法；掌握字符数组的定义、初始化及其使用方法；了解一维数组和二维数组在计算机中的存储结构；掌握一维数组的编程应用和二维数组的简单编程应用。  2.能力培养：更加方便快捷的处理实际问题。  3.素质要求：生活中的排队问题，如何用计算机实现在内存中相邻，并且能够顺利找到下一个元素？循环+数组的解决方案。  4.教学方法：  以数字通讯录为例讲解一维数组和二维数组的定义、数组元素的引用、数组的初始化及应用举例；  【重点】函数传递的机制，地址传递和值传递的区别；数组概念的理解和运用。  【难点】数组中的计算问题。 | 课程目标1、2 |
| 9 | 指针 | 【学习目的和要求】  1.知识掌握：  掌握指针和指针变量的概念、掌握指针变量的定义、赋值方法，熟练使用指针访问简单变量；掌握指针与字符串的关系；掌握用简单指针变量作函数参数及传递与调用方法  2.能力培养：  加深学生对内存存储机制的理解和运用。  3.素质要求：内存地址的理解，地址如宾馆房间号，内容如住在里面的客人。  4.教学方法  找学生回答预习题目，由预习题引出内存地址和指针的概念。  【重点】指针的本质；按值传递与按地址传递的区别；指针做参数的使用。  【难点】指针概念的理解和灵活运用。 | 课程目标1 |
| 10 | 字符串 | 【学习目的和要求】  1.知识掌握：  掌握C语言处理字符串的方法，一是用字符数组，二是用字符指针。  2.能力培养：  利用字符串解决名字、图书等存储问题。  3.素质要求：姓名如何存储？  4.教学方法  以通讯录案例为切入点，引入字符串处理问题。  【重点】一个和多个串的存储与提取。  【难点】串的排序问题。 | 课程目标1 |
| 11 | 指针和数组 | 【学习目的和要求】  1.知识掌握：  掌握指向一维数组的指针变量的定义方法，熟练使用指针变量访问一维数组元素；了解指针和二维数组的关系，掌握用指针表示二维数组元素的方法；掌握指针数组，用以处理多个字符串；了解动态数组的概念及使用。  2.能力培养：  培养学生灵活运用指针解决数组问题。  3.素质要求：一个问题可有不同解决方案，优劣对比。  4.教学方法  围绕前面课程中同学已经熟悉的案例，讲解指针法和下标法两种引用数组元素的区别和各自的优缺点；利用前面的字符串排序案例，讲解指针数组的使用及其优势。  【重点】指针法和下标法的区别；指针数组处理多个字符串的用法。  【难点】指针指向二维数组，行指针的概念理解，动态数组。 | 课程目标1、2 |
| 12 | 结构体与共用体 | 【学习目的和要求】  1.知识掌握：  掌握结构体的概念和特点；熟练定义结构体数据类型，掌握结构体变量的定义及初始化方法；了解结构体数组的特点，能使用结构体数组解决简单问题；掌握结构体指针的特点；掌握共用体的概念，熟悉共用体类型和共用体变量的定义和使用方法。  2.能力培养：  培养学生自己构造数据类型的能力，灵活解决实际问题的能力。  3.素质要求：当没有现成的工具可以用时，学会自己制作工具；自己构造适合问题处理的数据类型。  4.教学方法  以通讯录为例，进一步完善其表达数据的方式。围绕表达一个人完整信息的问题，讲解自定义数据类型，根据需要，创造一种新的数据类型。  【重点】结构体变量、结构体数组、结构体指针的用法。  【难点】思维的转换，自己定义一种数据类型，而不是变量！ | 课程目标1、2、3 |
| 13 | 文件操作 | 【学习目的和要求】  1.知识掌握：  了解文件的概念，了解ASCII码文件和二进制文件；了解文件缓冲区的概念，熟悉文件数据类型和文件指针的概念，熟悉文件指针的使用方法；熟悉文件读写方式的概念，熟悉基本的文件操作函数；熟悉文件数据块读写的概念及操作方法。对该门课程内容总结重点知识，加深理解和运用。  2.能力培养：  培养学生灵活编程，能够用文件实现存储和数据提取，更加灵活的使用已有数据并对更新数据进行保存。  3.素质要求：如何长期保留程序的运行结果而不丢失？依靠文件的输入和输出实现。  4.教学方法  找学生回答预习题目，从中引出流、文件的概念，讲解二进制文件和文本文件的区别。使学生了解文件的概念，ASCII码文件和二进制文件；  【重点】打开、关闭、读写函数的参数使用。  【难点】二进制存储的概念较为抽象，不容易理解。 | 课程目标1、2 |

**五、各教学环节及学时分配**

| **知识模块** | **教学内容** | **教学环节及学时** | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **讲授课** | **习题课** | **讨论课** | **实验（实践）** | **合计** |
| 第1章  为什么要学习编程 | 1.1 学习编程的热潮  1.2 为什么要学编程  1.3 什么是“编程”  1.4 怎么学编程 | 2 |  |  | 2 | 4 |
| 第2章  基本数据类型 | 2.1 常量与变量  2.2 简单的屏幕输出  2.3 数据类型  2.4 如何计算变量或数据类型所占内存空间的大小  2.5 变量的赋值和赋值运算符  2.6 本章扩充内容  2.7 本章知识点小结  2.8 本章常见错误小结 | 2 |  |  | 2 | 4 |
| 第3章  基本算术运算 | 3.1 C运算符和表达式  3.2 宏常量与宏替换  3.3 const常量  3.4 自动类型转换与强制类型转换运算符  3.5 常用的标准数学函数 | 2 | 1 |  | 2 | 5 |
| 第4章  键盘输入和屏幕输出 | 4.1 单个字符的输入/输出  4.1.1 字符常量  4.1.2 字符的输入/输出  4.2 数据的格式化屏幕输出  4.3 数据的格式化键盘输入 | 2 |  |  | 2 | 4 |
| 第5章 选择控制结构 | 5.1 生活中与计算机中的问题求解方法  5.2 算法的概念及其描述方法  5.3 关系运算符与关系表达式  5.4 用于单分支控制的条件语句  5.5 用于双分支控制的条件语句  5.6 条件运算符和条件表达式  5.7 用于多分支控制的条件语句  5.8 用于多路选择的switch语句  5.9 逻辑运算符和逻辑表达式 | 4 | 1 |  | 2 | 7 |
| 第6章  循环控制结构 | 6.1 循环控制结构与循环语句  6.2 计数控制的循环  6.3 嵌套循环  6.4 条件控制的循环  6.5 流程的转移控制 | 5 | 1 |  | 2 | 8 |
| 第7章  函数与模块化程序设计 | 7.1 分而治之与信息隐藏  7.2 函数的定义  7.3 向函数传递值和从函数返回值  7.4 函数的递归调用和递归函数  7.5 变量的作用域和生存期  7.6 模块化程序设计 | 5 | 1 |  | 2 | 8 |
| 第8章  数组和算法基础 | 8.1 一维数组的定义和初始化  8.2 二维数组的定义和初始化  8.3 向函数传递一维数组  8.4 排序和查找  8.5 向函数传递二维数组 | 3 | 1 |  | 2 | 6 |
| 第9章  指针 | 9.1 变量的内存地址  9.2 指针变量的定义和初始化  9.3 间接寻址运算符  9.4 按值调用与模拟按引用调用  9.5 用指针变量作函数参数的程序实例  9.6 函数指针及其应用 | 4 | 1 |  | 2 | 7 |
| 第10章  字符串 | 10.1 字符串字面量  10.2 字符串的存储  10.3 字符指针  10.4 字符串的访问和输入/输出  10.5 字符串处理函数  10.6 向函数传递字符串  10.7 从函数返回字符串指针 | 3 | 1 |  | 2 | 6 |
| 第11章  指针和数组 | 11.1 指针和一维数组间的关系  11.2 指针和二维数组间的关系  11.3 指针数组及其应用  11.4 动态数组 | 3 |  |  | 2 | 5 |
| 第12章  结构体和数据结构基础 | 12.1 从基本数据类型到抽象数据类型  12.2 结构体的定义  12.3 结构体数组的定义和初始化  12.4 结构体指针的定义和初始化  12.5 向函数传递结构体  12.6 共用体  12.7 枚举数据类型  12.8 动态数据结构单向链表 | 3 | 1 |  | 2 | 6 |
| 第13章  文件操作 | 13.1 二进制文件和文本文件  13.2 文件的打开和关闭  13.3 按字符读写文件  13.4 按格式读写文件  13.5 按数据块读写文件 | 2 |  |  | 2 | 4 |
| 总复习 |  | 2 |  |  |  | 2 |
| **合计** | | | | | | 80 |

**实验学时分配表**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 实验项目名称 | 学时 | 实验类型 | 实验类别 | 实验内容 | 实验基本要求 | 备注 |
| 01 | C语言程序调试初步 | 2 | 验证 | 必做 | 熟悉vc++6.0或code::block的环境 | 掌握调试程序所用的方法、步骤及基本错误的修改，调试出一个小程序。 |  |
| 02 | 数据定义和相关运算 | 2 | 验证 | 必做 |  | 学会基本的数据定义和相关运算 |  |
| 03 | 顺序结构的程序设计 | 2 | 验证 | 必做 | 编写顺序结构的程序设计 | 实现顺序结构的程序设计题目3~5题 |  |
| 04 | 选择结构的程序设计 | 2 | 验证 | 必做 | 编写选择结构的程序设计 | 实现选择结构的程序设计题目3~5题，学会应基本的分支if,if-else,switch语句及分支语句的嵌套用法 |  |
| 05 | 循环结构的程序设计 | 2 | 设计 | 必做 | 编写循环结构的程序设计 | 实现顺序结构的程序设计题目3~5题，掌握语句的使用，掌握循环嵌套的用法 |  |
| 06 | 认识数据类型 | 2 | 验证 | 必做 | 四种C语言基本数据类型的使用 | 能用基本数据类型及输入、输出语句进行简单编程 |  |
| 07 | 使用运算符和表达式进行计算 | 2 | 验证 | 必做 | 运算符的使用 | 掌握运算符的优先级，正确使用表达式求值 |  |
| 08 | 函数及其应用 | 4 | 设计 | 必做 | 进行函数程序设计 | 掌握函数的定义、调用、尝试用函数解决实际问题 |  |
| 09 | 数组及其应用 | 4 | 设计 | 必做 | 进行数组程序设计 | 掌握数组的定义、初始化及排序算法 |  |
| 10 | 指针及其应用 | 4 | 设计 | 必做 | 进行指针程序设计 | 掌握指针的定义、应用，指针和数组的关系 |  |
| 11 | 字符串及其应用 | 4 | 设计 | 必做 | 进行字符串程序设计 | 掌握字符串的定义、应用，指针和字符串的关系；C中处理字符串的函数 |  |
| 12 | 结构体及其共用体 | 2 | 设计 | 必做 | 进行结构体和共用体的程序设计 | 掌握结构体的定义、应用，指针和结构体的关系 |  |

**六、成绩考核及基本要求**

| **考核**  **环节** | **建议分值** | **考核/评价细则** | **对应课程**  **目标** |
| --- | --- | --- | --- |
| 考勤（含上机） | 10 | 旷课、迟到、早退1次扣1分；旷课、迟到、早退超过3次的学生取消考试资格，直接认定为成绩不及格。 | 3 |
| 课堂表现 | 10 | 根据平常上课过程中的学习态度、回答问题等情况进行评价，表现优秀得9-10分，表现良好得8-9分，表现中等得7-8分，表现合格得6-7分，表现不合格得4-6分。 | 2、3 |
| 作业 | 5 | 根据作业完成情况进行评价，超过3次不交作业，5分全部扣除。优秀得5分，良好得4分，中等得3分，3分以下不及格。 | 2、3 |
| 上机考试 | 15 | 学期的最后一次上机安排上机考试，随机抽取试卷。根据程序的运行结果进行评价。优秀得14-15分，良好得12-13分，中等得10-11分，及格得8-9分，不及格得6-7分。 | 1、2、3 |
| 期末卷面考试 | 60 | 期末考核方式：笔试；试卷总成绩：100；试卷成绩\*60%为考试成绩。 | 1、2、3 |

**七、教材与主要教学参考资源**

## （一）推荐教材与参考书

主讲教材：

《C 语言程序设计》（第4版） 苏小红等， 高等教育出版社， 2019,8

参考教材：

《Ｃ 程序设计》 谭浩强等， 清华大学出版社， 1999.12

《C 语言程序设计》 周世平等， 清华大学出版社， 2016,3

《程序设计基础》（ 第三版）， 吴文虎 徐明星 ， 清华大学出版社， 2012,7  
《程序设计基础》（ C 语言）， 田爱奎等 ， 清华大学出版社， 2011,5

## （二）推荐网站（包括课程网站、专业网站等）

1. https://coursehome.zhihuishu.com/courseHome/1000007419#teachTeam智慧树 C语言程序设计(山东联盟-青岛大学）

2.http://www.cmpedu.com/ 机械工业出版社教育服务网；

3. https://www.icourse163.org/course/HIT-69005?from=searchPage 中国大学MOOC

制定者：徐红梅

审核者：李超

批准者：邓建球

编写时间： 2021 年 6 月