一、初识C语言——C语言是自上而下的高级计算机语言，写起来更加紧凑、简洁、高效，读起来更加易懂，应用起来更加可靠，C语言功能强大且灵活，相比其他编程语言具有可移植的明显优势。C语言的应用范围非常广，在未来依旧会强势流行。C++是C语言的超集，学习好C语言，也就相当于学习了许多C++的知识。编写C程序有七个步骤：定义程序的目标→设计程序→编写代码→编译→运行程序→测试和调试程序→维护和修改程序。C的编程机制是，用程序把源代码转换为可执行文件。典型的，通过编译和链接：编译器把源代码转换为中间代码，链接器把中间代码和其他代码合并，生成可执行文件。

二、C语言概述——新旧的两种注释格式，注意“消失代码”的问题；以前的C语言要求把变量声明在块的顶部，其他语句不能在任何声明的前面，而C99和C11遵循C++惯例，可以把声明放在块中的任何位置等等。声明变量能带来以下好处：让程序的可读性更高，促使程序员在编写代码之前做出计划，有助于发现隐藏在程序中的小错误，保证C语言通过编译。要养成一些编写程序的好习惯，首先main()函数中保留return语句，可以视其为统一的代码风格；选择有意义的函数名，写注释时避免重复啰嗦，使用空行分隔概念上的多个部分，以提高程序的可读性；把自己想象成计算机，跟着程序的步骤一步一步执行；注意使用标准的编译器能够理解的术语。

三、数据和C——C语言有多种数据类型。整数可以表示为八进制、十进制、十六进制。浮点型包括float、double、long double，特别地，注意浮点数舍入错误。基本数据类型由11个关键字组成；有符号整型可表示正整数和负整数；无符号整型只能用于表示零和正整数，在整型类型前加上关键字unsigned表明该类型是无符号整型；必要时可在char前加关键字signed或unsigned注明类型；布尔值表示true和false即1和0；实浮点类型可表示正浮点数和负浮点数；加后缀著名复数的虚部和实部。注意C语言的数据类型关键字。C将字符常量视为int类型而非char类型。注意转义序列字符。

四、字符串和格式化输入/输出——单引号标识单个字符，占用一个存储单元，双引号标识字符串，末尾用\0标记字符串的结束，这意味着数组的容量必须至少比带存储字符串的字符数多1。sizeof运算符以字节为单位给出对象的大小，strlen()函数给出字符串中的自促长度，sizeof运算符给出的数更大，因为它包括了字符串末尾不可见的空字符。使用sizeof运算符，对于类型，应该写成sizeof(char)、sizeof(float)，对于特定量，可写成sizeof name、sizeof 6.28。更好的习惯是所有情况下都使用圆括号。符号常量两种创建方式及：一是声明一个变量再将该变量设置为所需的常量，但这种方法有时程序会无意间改变它的值；二是明示常量，即编译时替换：#define NAME value，用大写表示符号常量是C语言的一贯传统。scanf()函数和printf()函数是输入/输出函数，二者工作原理几乎相同。scanf()把输入的字符串转换成整数、浮点数、字符或字符串，printf()正好相反，把整数、浮点数、字符或字符串转换成显示在屏幕上的文本。

十六、C预处理器和C库——C预处理器在程序执行之前查看程序，所以称预处理器。处理器可以包含程序所需的其他文件，可以选择让编译器查看哪些代码，预处理器并不知道C。预处理器表达式的长度必须是一个逻辑行，一个逻辑行可以是多个物理行。在预处理器之前，编译器要对该程序进行一些翻译：编译器把源代码中出现的字符映射到源字符集；编译器定位并删除每个反斜杠后面跟着换行符的实例；编译器把文本划分成预处理记号序列、空白序列和注释序列。预处理器指令以#作为一行的开始，旧版本要求指令从一行最左边开始，而且#和指令其余部分之间不能有空格，ANSI和后来的标准都允许#号前面有空格或制表符，而且允许在#和指令其余部分之间有空格。指令可以出现在源文件的任何地方，其定义从指令出现的地方到该文件的末尾。C库提供许多有助于完成各种任务的函数。使用库：访问C库、使用库描述。pi值通过计算式4\*atan(1)得到。

十七、高级数据表示——程序开发最重要的部分是找到程序中表示数据的好方法，还必须确定如何存储该数据类型，并设计一系列管理该数据的函数。修改程序遵循原则：强调最初的设计，并简化其他细节。三步完成定义新类型：首先，提供类型属性和相关操作的抽象描述（ADT），其次，开发一个实现ADT的编程接口，即指明如何存储数据和执行所需操作的函数，最后，编写代码实现接口。链表、队列和二叉树是ADT在计算机程序设计中常用的实例，通常用动态内存分配和链式结构来实现，但有时用数组来实现会更好。