**《C 语言程序设计》**

**（第二版）**

**教案**

**北京出版社**

**课时分配表**

|  |  |  |  |
| --- | --- | --- | --- |
| **章序** | **课程内容** | **课时** | **备注** |
| **1** | **认识 C 程序** | **6** |  |
| **2** | **C 程序设计基础** | **7** |  |
| **3** | **顺序结构程序设计** | **5** |  |
| **4** | **选择结构程序设计** | **3** |  |
| **5** | **循环结构程序设计** | **6** |  |
| **6** | **数组** | **5** |  |
| **7** | **用户自定义函数** | **7** |  |
| **8** | **编译预处理** | **5** |  |
| **9** | **指针** | **9** |  |
| **10** | **构造数据类型** | **5** |  |
| **11** | **文件读写技术** | **6** |  |
| **12** | **C 语言综合实训** | **4** |  |
| **13** | **附录** | **4** |  |
| **总计** |  | **72** |  |

**第5课 循环结构程序设计**

|  |  |  |
| --- | --- | --- |
| **课 题** | 循环结构程序设计 | |
| **课 时** | 6课时（270 min）。 | |
| **教学目标** | **知识技能目标：**  1．了解循环结构的编程思想。  2．通过学习与练习掌握实现循环的三种语句。  **思政育人目标：**  让学生通过学习循环结构程序设计，能够由浅入深的思维，具备反复推敲的习惯，养成积极向上的生活态度、一丝不苟的处事作风。 | |
| **教学重难点** | **教学重点：**循环结构的编程思想  **教学难点：**实现循环的三种语句 | |
| **教学方法** | 讲授法、问答法、讨论法 | |
| **教学用具** | 电脑、投影仪、多媒体课件、教材 | |
| **教学设计** | 第1节课：考勤（2min）--知识讲解（40min）--作业布置（3min）  第2节课：知识讲解（40min）--课堂小结（3min）--作业布置（2min）  第3节课：知识讲解（40min）--课堂小结（3min）--作业布置（2min）  第4节课：知识讲解（40min）--课堂小结（3min）--作业布置（2min）  第5节课：知识讲解（40min）--课堂小结（3min）--作业布置（2min）  第6节课：知识讲解（40min）--课堂小结（3min）--作业布置（2min） | |
| **教学过程** | **主 要 教 学 内 容 及 步 骤** | **设计意图** |
| **考勤**  **（2min）** | ■【教师】清点上课人数，记录好考勤  ■【学生】班干部报请假人员及原因 | 培养学生的组织纪律性,掌握学生的出勤情况 |
| **知识讲解**  （40min） | **【教师】**循环结构的编程思想  **任务描述**  观察下面的一组语句：  sum=0;  sum=sum+1;  sum=sum+2;  sum=sum+3;  ……  sum=sum+100;  100 条  可以看出：1，2，3，4，…，100 依次被累加到变量 sum 中，经过 100 次累加后，求出了 1~100 的和。但是这样的编程方法是人们难以接受的。如果使用循环解决这个问题，需要考虑哪些要素？  **预备知识**  循环结构（又称重复结构）是程序中的另一个基本语句。它的特点是：在给定的条件成立时，反复执行某个程序段。给定的条件叫循环条件，反复执行的程序段叫循环体。  C 语言中实现循环的语句包括：当型循环语句 while 和 for，直到型循环语句 dowhile 以及 goto 语句等。它们的结构流程如图 5-1（a）、图 5-1（b）所示。    由图5-1（a）、图5-1（b）可以看出，当型循环把循环控制条件的判断放在循环的入口，即先判断条件，满足条件后才执行循环体，否则就退出循环体；直到型循环则先执行循环体，然后再判断循环控制条件，若满足条件则继续执行循环体，否则就退出循环体。这两种结构可以根据具体情况选择使用。一般来说，如果有循环体执行次数等于0 的可能，则应选择当型循环，否则用直到型循环。但不论是哪一种形式，设计循环程序一般应满足如下3个要素。  1.设置循环的初始状态  例如：设置循环次数的计数值，以及为循环体正常工作而建立的初始状态等。  2. 循环体  这是循环工作的主体，它由循环的工作部分和修改部分组成。循环的工作部分是为完成程序功能而设计的程序段；循环的修改部分则是为保证每一次重复（循环）时，参加执行的信息能够发生有规律的变化而建立的程序段。  3. 循环控制部分  循环控制部分本来应该属于循环体的一部分，由于它是循环程序设计的关键，所以要对它作专门的讨论。每个循环程序必须选择一个循环控制条件来控制循环的运行和结束，而合理地选择该控制条件就成为循环程序设计的关键问题。有时候，循环次数是已知的，此时可以用循环次数作为循环的控制条件；有时循环次数是未知的，那就需要根据具体的情况找出控制循环结束的条件。循环控制条件的选择是很灵活的，有时选择方案可能不止一种，此时，就应通过分析比较选择一种效率最高的方案来实现。  **【学生】**思考、讨论。 | **展示循环结构的编程思想，让学生更加仔细的阅读，从而激发学生的学习欲望。** |
| **作业布置**（3min） | **【教师】**布置课后作业  求 1~100 的和的三大循环要素。 | 通过课后练习，使学生巩固所学新知识 |
| **知识讲解**  （40min） | **【教师】**实现循环的三种语句  **任务描述**  有一分数序列：2/1，3/2，5/3，8/5，13/8，21/13，…，求出这个数列前 20 项的和。  **任务分析**  求和程序共性：  （1）循环初值：sum=0（存放和），i=1（循环控制变量），t1=2，t2=1；（t1/t2 代表被加的数）；  （2）循环体：sum=sum+t1/t2；i=i+1；  （3）循环控制部分：i<=20。  **预备知识**  C 语言提供了三种描述循环的语句：while 语句、do-while 语句和 for 语句。  **一、while 循环语句**  while 循环是“当型”循环，它的一般格式是：  while( 表达式 )  {  循环体  }  【功能】先计算表达式的值，若表达式的值为非 0，则执行一次循环体。每执行一次循环体，就判断一次表达式的值，直到表达式的值为 0 时结束循环。  【例 5-1】编程：用 while 语句求 1+2+ …+100 的和。程序流程图如图 5-4 所示。  源程序：  #include <stdio.h>  void main()  {  int sum=0,i=1;  while(i<=100)  {  sum=sum+i;  i++;  }  printf("1+2+…+100=%d\n",sum);  }  程序运行结果：  1+2+…+100=5050  **二、do-while 循环语**句  do-while 语句用来实现“直到型”循环结构，其一般格式如下：  do  {  循环体  }while( 表达式 );  【功能】先执行一次循环体，然后再计算表达式的值，若表达式的值为非 0，则继续执行一次循环体。重复上述过程，直到表达式的值为 0 时结束循环。  【例 5-2】编程：求 1+2+ …+100 的和（用 do-while 语句实现）。  源程序：  #include <stdio.h>  main()  {  int sum=0,i=1;  do  {  sum=sum+i;  i++;  }while(i<=100);  printf("1+2+…+100=%d\n",sum);  }  程序运行结果：  1+2+…+100=5050  **三、for 循环语句**  for 语句是 C 语言所提供的功能更强、使用更广泛的一种循环语句。for 循环是“当型”循环，它的一般格式是：  for( 表达式 1; 表达式 2; 表达式 3)  {  循环体  }  【功能】首先计算表式 1 的值，再判断表达式 2，如果其值为非 0，则执行循环体，并计算表达式 3；然后再判断表达式 2，直到表达式 2 值为 0 时结束循环。通常，表达式 1 用于给变量赋初值，表达式 2 是循环控制条件，用于判断循环是否继续，表达式 3 则常常是使循环趋于结束的语句。  **任务实施**  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\* 程序 : program5\_1.c \*/  /\* 功能 : 求分数序列 :2/1,3/2,5/3,8/5,13/8,21/13,…前 20 项的和。 \*/  /\* 说明 : 利用 for 语句实现 \*/  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  #include <stdio.h>  void main()  { int i;  fl oat sum=0,t1=2,t2=1,t ;  for(i=1;i<=20;i++)  {  sum=sum+t1/t2;  t=t1;  t1=t1+t2;  t2=t;  }  printf("sum=%f\n",sum);  }  程序运行结果：  sum=32.660259  **【学生】**思考、讨论。 | **通过教师讲解，了解实现循环的三种语句的基本理论知识。** |
| **课堂小结**  （3min） | 【**教师**】**回顾和总结本节课的知识点。**  **这节课我们一起学习了实现循环的三种语句，让学生能够发现循环解题的规律，能体会“变则通，通则久”“易简而天下之理得矣”的人生哲理。** | 通过对所学知识的回顾，培养学生的归纳总结能力 |
| **作业布置**（2min） | **【教师】**布置课后作业  编程，求 1～100之间所有偶数之和。 | 通过课后练习，使学生巩固所学新知识 |
| **知识讲解**（45min） | **【教师】**与循环有关的控制语句  **任务描述**  键盘输入一个整数 n，判断它是否素数。  **任务分析**  所谓素数是指除了 1 和它本身能被它整除以外，其他所有的数都不能被它整除的数。  根据素数的定义，程序可以从以下思路入手。  让循环变量i从2向n-1变化（事实证明，没有必要到 n-1，只要变化到 sqrt（n）就可以了），让 n 被每一个 i 除，整除过程中会遇到两种情况：  （1）只要发现有 1 个数能被 n整除，就可以判断其不是素数，可以立即跳出循环，用 break 语句来实现（此时循环变量 i 的值肯定未超过终值 sqrt（n））。  （2）如果 n 不能被 2 到 sqrt（n）之间的任何一个数整除，则在完成最后一次循环后，i 还要加 1，因此 i肯定超过了终值 sqrt（n）。退出循环体之后，通过判断 i 的值是否大于终值，即可确定其是不是素数。  **预备知识**  一般情况下，循环程序都是由“循环控制条件”控制循环的开始和结束，但有些循环程序需要在某些条件成立的时候改变循环的正常流程，立即终止正在执行的循环操作。与循环有关的控制语句有 break 语句和continue 语句，它们可以控制循环程序的执行流程。  **一、break 语句**  break 语句的一般格式：  break;  【功能】  （1）用在 switch 语句中，结束 case 分支，跳出 switch 结构。  （2）用在循环体中，强迫循环程序立刻结束，转而执行循环体语句后的其他语句。  【例 5-4】分析下列程序的执行结果，并说明它的功能。  #include <stdio.h>  void main()  { int s=0,i;  for(i=1 ; ;i++)  {  if(i>50) break;  s+=i;  }  printf("s=%d",s);  }  程序分析：  该例题中的 for 语句省略“表达式 2”，代表循环条件恒为“真”。因此，该循环程序的结束是当循环条件“i>50”为“真”时，执行 break 语句结束的。  程序运行结果：  s=1275  程序的功能：  　　求 1~50 的和。  **二、continue 语句**  continue 语句的一般格式：  continue;  【功能】中断循环体的本次执行（即跳过循环体中尚未执行的语句），立即开始执行下一次循环。  **任务实施**  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\* 程序 : program5\_2.c \*/  /\* 功能 : 键盘输入一个整数 n, 判断它是否为素数。 \*/  /\* 说明 : 利用 for 语句实现 \*/  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  #include <math.h>  #include <stdio.h>  void main()  {  int n,i,k;  printf(" 请输入一个整数 :");  scanf("%d",&n);  k=sqrt(n);  for(i=2;i<=k;i++)  if(n%i==0) break;  if(i>k) printf("%d 是素数 \n",n);  else printf("%d 不是素数 \n",n);  }  程序运行结果：  请输入一个整数：7 ↙  7 是素数  **【学生】**思考、讨论。 | **教师通过与循环有关的控制语句展示，让学生了解与循环有关的控制语句的基本理论知识。** |
| **课堂小结**  （3min） | 【**教师**】**回顾和总结本节课的知识点。**  **这节课我们一起学习了与循环有关的控制语句，能将抽象的概念转换成程序设计思路，能够举一反三、触类旁通。** | 通过对所学知识的回顾，培养学生的归纳总结能力 |
| **作业布置**（2min） | 【**教师**】**布置课后作业**  **下列程序的功能是，判断整数 n 是不是“完数”，请将程序补充完整。**  **编程提示：有一个数如果恰好等于它的因子之和，这个数就是“完数”。例如，6 的因子为 1、2、3，而 6=1+2+3，因此 6 是“完数”。** | 通过课后练习，使学生巩固所学新知识 |
| **知识讲解**（45min） | **【教师】**循环的嵌套  **任务描述**  编程，打印 100-200 之间的素数。  **预备知识**  在一个循环的循环体内又包含另一个完整的循环结构，称为循环的嵌套。例如：  for(i=1;i<=3;i++) /\* 外循环 \*/  for(j=1;j<=3;j++) /\* 内循环 , 也是外循环的循环体 \*/  printf("\*"); /\* 内循环的循环体 \*/  该程序段的输出结果是：\*\*\*\*\*\*\*\*\* （9 个 \* 号）  内嵌的循环中还可以嵌套循环，这就是多层循环。各种语言中关于循环嵌套的概念都是一样的。C 语言的 3 种循环（while 循环、do-while 循环和 for 循环）可以互相嵌套。  例如，下面几种循环嵌套形式都是合法的。  (1)while()  { …  while()  { …}  }  (2)do  { …  do  { …}  while();  }  while();  (3)for( ; ;)  {  for( ; ;)  {…}  }  (4)while()  { …  do  {…}  while();  …  }  (5)for( ; ;)  { …  循环嵌套  C 语言程序设计（第二版）  82  while()  { … }  …  }  (6)do  {  …  for(;;)  { … }  }  while();  【注意】使用循环嵌套要注意以下几点：  （1）内外循环不能交叉，即：内循环应完整的包含在外循环体内，否则会引起歧义。  （2）循环的内外变量名不应重名，否则会造成混乱。  （3）循环嵌套书写时应采用缩进格式，保持良好的程序设计风格，提高程序的易读性。  **任务实施**  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\* 程序 : program5\_3.c \*/  /\* 功能 : 编程 , 打印 100~200 之间的素数。 \*/  /\* 说明 : 在 program5\_2.c 的基础上 , 对本题用一个嵌套的 for 循环处理即可。 \*/  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  #include <math.h>  #include <stdio.h>  void main()  {  int m,k,i,n=0;  for(m=101;m<=199;m=m+2)  {  if(n%10==0) printf("\n"); /\* 控制一行输出 10 个素数 \*/  k=sqrt(m);  for(i=2;i<=k;i++)  if(m%i==0) break;  if(i>k) { printf("%d ",m);n=n+1;}  }  }  程序运行结果：  101 103 107 109 113 127 131 137 139 149  151 157 163 167 173 179 181 191 193 197  199  **【学生】**思考、讨论。 | **教师通过循环的嵌套展示，让学生了解循环的嵌套的基本理论知识。** |
| **课堂小结**  （3min） | 【**教师**】**回顾和总结本节课的知识点。**  **这节课我们一起学习了循环的嵌套，让学生能与同学之间进行良好的沟通，具备自主学习的意识和能力。** | 通过对所学知识的回顾，培养学生的归纳总结能力 |
| **作业布置**（2min） | 【**教师**】**布置课后作业**  **编程，打印 1000 以内的完数。**  **#include <stdio.h>**  **void main()**  **{**  **int n,i,k,sum;**  **for(n=1;n<=1000;n++)**  **{ sum=0; /\* 注意 :sum 必须在此处赋值 0 \*/**  **for(i=1;i<=n/2;i++)**  **if(n%i==0)\_\_\_\_\_\_\_\_\_\_\_\_\_\_;**  **if(\_\_\_\_\_\_\_\_\_\_\_) printf("%5d",n);**  **}**  **}** | 通过课后练习，使学生巩固所学新知识 |
| **知识讲解**（45min） | **【教师】**典型例题  在全系1000学生中，征集慈善募捐，当总数达到10万元时就结束，统计此时捐款的人数，以及平均每人捐款的数目。  ➢编程思路:  ◆循环次数不确定，但最多循环1000次  ●在循环体中累计捐款总数  ●用if语句检查是否达到10万元  ●如果达到就不再继续执行循环，终止累加  ◆计算人均捐款数  #include之stdio.h>IGHUA UNIVERSITY PRESS  #define SUM 1 00000  void main( )  指定符号常量SUM代表100000  { float amount,aver,total;  int i;  for (i=1 ,total=0;i<=1000;i++)  { printf("please enter amount:") ;  scanf("%f', &amount);  total= total+amount;  if (total> =SUM) break;  }  aver=total /i ;  printf("num=%d\naver= %10.2f\n"  ,i,aver);  **【学生】**思考、讨论。 | **教师通过典型例题展示，让学生了解议论文阅读与写作的基本理论知识。** |
| **课堂小结**  （3min） | 【**教师**】**回顾和总结本节课的知识点。**  **这节课我们一起学习了典型例题，提高学生对循环的理解。** | 通过对所学知识的回顾，培养学生的归纳总结能力 |
| **作业布置**（2min） | 【**教师**】**布置课后作业**  **对于以下代码：**  **int i;**  **for(i=0;i<10;i++)**  **printf("%d",i);**  **for循环结束后，i的值是\_\_**  **A.11**  **B.10**  **C.9**  **D.i不再存在了** | 通过课后练习，使学生巩固所学新知识 |
| **知识讲解**（45min） | **【教师】**实验结论循环结构程序设计  编程是计算机科学中非常重要的一部分，而程序设计中常用到的循环结构也是非常重要的一种控制流程方式。本文将介绍如何进行循环结构程序设计，并给出相应实验结果及分析。  **实验目的：**  通过本次实验，我们将学会如何使用不同类型的循环结构来控制程序流程，并能够编写简单而有效的程序。  **实验原理：**  1. 循环结构的概念：在程序运行过程中，为了重复执行某些操作或者根据特定条件反复执行某些操作，需要使用到循环结构。  2. 循环结构的分类：常见的有while循环、do-while循环和for循环。  3. 循环结构程序设计基本原则：确定好需要重复执行多少次或在什么情况下需要重复执行，选择合适的循环结构语句，编写循环体语句。  **实验步骤：**  1. 确定循环次数和循环条件：在程序设计之前，需要明确需要重复执行多少次或在什么情况下需要重复执行。  2. 选择合适的循环结构语句：根据确定好的循环次数和循环条件，选择while、do-while或for等不同类型的循环结构语句。  3. 编写循环体语句：在循环结构中，需要编写具体的操作语句来实现所需功能。  **实验结果分析：**  通过本次实验，我们可以发现，在程序设计中使用不同类型的循环结构可以有效地控制程序流程，并且能够编写出简单而有效的程序。同时，在实际应用中也需要根据具体需求来选择合适的循环结构类型以及确定好相应的循环次数和条件。  **【学生】**思考、讨论。 | **教师通过文章展示，让学生了解议论文阅读与写作的基本理论知识。** |
| **课堂小结**  （3min） | 【**教师**】**回顾和总结本节课的知识点。**  **这节课我们一起学习了实验结论循环结构程序设计，通过本次实验，我们深入了解循环结构程序设计，并掌握了相应技能。** | 通过对所学知识的回顾，培养学生的归纳总结能力 |
| **作业布置**（2min） | 【**教师**】**布置课后作业**  **简述循环结构程序设计基本原则。** | 通过课后练习，使学生巩固所学新知识 |
| **教学反思** | 在教学过程中，必须创设和谐的教学情境，让课堂能够活起来，让学生也能够在和谐的氛围中体验到学习的乐趣。 | |