第 18 章　上机实验的目的和要求

18.1　上机实验的目的

学习 C 语言程序设计课程不能满足于能看懂书上的程序，而应当熟练地掌握程序设计的全过程，即独立编写出源程序，独立上机调试程序，独立运行程序和分析结果。

程序设计是一门实践性很强的课程，必须保证有足够的上机实验时间。学习本课程应该至少有 20 小时的上机时间，最好能做到与授课时间之比为1∶1。除了教师指定的上机实验以外，应当提倡学生自己课余抽时间多上机实践。

上机实验的目的绝不仅是为了验证教材和讲课的内容，或者验证自己所编的程序正确与否。学习程序设计，上机实验的目的是：

(1) 加深对讲授内容的理解，尤其是一些语法规定，光靠课堂讲授，既枯燥无味又难以记住，但它们都很重要。通过多次上机，就能自然地、熟练地掌握。通过上机来掌握语法规则是行之有效的方法。

(2) 了解和熟悉 C 语言程序开发的环境。一个程序必须在一定的外部环境下才能运行，所谓“环境”，就是指所用的计算机系统的硬件和软件条件。使用者应该了解，为了运行一个 C 程序需要哪些必要的外部条件(例如硬件配置、软件配置)，可以利用哪些系统的功能来帮助自己开发程序。每一种计算机系统的功能和操作方法不完全相同，但只要熟练掌握一两种计算机系统的使用，再遇到其他系统时便会触类旁通，很快就能学会。

(3) 学会上机调试程序。也就是善于发现程序中的错误，并且能很快地排除这些错误，使程序能正确运行。经验丰富的人在编译和连接过程中出现“出错信息”时，一般能很快地判断出错误所在，并改正之。而缺乏经验的人即使在明确的“出错提示”下也往往找不出错误而求助于别人。

要真正掌握计算机应用技术，不仅应当了解和熟悉有关的理论和方法，还要自己动手实现。对程序设计来说，要求会编程序并上机调试，使程序能正常运行，并且会分析运行结果，判断结果是否正确。

调试程序本身是程序设计课程的一个重要的内容和基本要求，应给予充分的重视。调试程序固然可以借鉴他人的现成经验，但更重要的是通过自己的直接实践来积累经验，而且有些经验是只能“意会”，难以“言传”。别人的经验不能代替自己的经验。调试程序的能力是每个程序设计人员应当掌握的一项基本功。

因此，在做实验时千万不要在程序通过后就认为万事大吉、完成任务了。即使运行结果正确，也不等于程序质量高和很完善。在得到正确的结果以后，还应当考虑是否可以对程序作一些改进。

在进行实验时，在调试通过程序以后，可以进一步思考，对程序做一些改动(例如修改一些参数、增加程序的一些功能、改变输入数据的方法等)，再进行编译、连接和运行。甚至应

Scanned with CamScanner

"自设障碍",即把正确的程序改为有错的(例如用 scanf 函数输入变量时,漏写"&"符号;使数组下标出界;使整数溢出等),观察和分析所出现的情况,这样的学习才会有更大的收获。

18.2 上机实验前的准备工作

上机实验前应事先做好准备工作,以提高上机实验的效率。准备工作至少应包括:

(1) 了解所用的计算机系统(包括 C 编译系统)的性能和使用方法。

(2) 复习和掌握与本实验有关的教学内容。

(3) 准备好上机所需的程序。由于计算机实验室给每个学生安排的时间是有限的,要珍惜时间,充分利用。应当在上机前按指定的题目编写好程序。手编程序应书写整齐,并经人工检查无误后再上机,以提高上机效率。初学者切忌不编程序或抄别人的程序去上机,应从一开始就养成严谨的科学作风。

(4) 对运行中可能出现的问题事先作出估计,对程序中自己有疑问的地方应作出记号,以便在上机时给予注意。

(5) 准备好调试和运行时所需的数据。

18.3 上机实验的步骤

上机实验时应该独立上机。上机过程中出现的问题,除了系统的问题以外,一般应自己独立处理,不要动辄问教师。尤其对"出错信息"应善于自己分析判断。这是学习调试程序的良好机会。

上机实验一般应包括以下几个步骤:

(1) 进入 C 工作环境(例如 Visual C++ 6.0 集成环境)。

(2) 输入自己所编好的程序。

(3) 检查一遍已输入的程序是否有错(包括输入时输错的和编程中的错误),如发现有错,及时改正。

(4) 进行编译和连接。如果在编译和连接过程中发现错误,屏幕上会出现"报错信息",根据提示找到出错位置和原因,加以改正,再进行编译……如此反复,直到顺利通过编译和连接为止。

(5) 运行程序并分析运行结果是否合理和正确。在运行时要注意当输入不同数据时所得到的结果是否正确(例如,解 $ax^2+bx+c=0$ 方程时,不同的 a,b,c 组合得到相应的不同结果)。此时应运行几次,分别检查在不同情况下程序是否正确。

(6) 输出程序清单和运行结果。

18.4 实验报告

实验后,应整理出实验报告。实验报告应包括以下内容:

(1) 题目。

(2) 程序清单(计算机打印出的程序清单)。

(3) 运行结果(必须是上面程序清单所对应的打印输出结果)。

Scanned with CamScanner

（4）对运行情况所做的分析以及本次调试程序所取得的经验。如果程序未能通过，应分析其原因。

18.5 实验内容安排的原则

课后习题和上机题统一。教师指定的课后习题就是上机题（可以根据习题量的多少和上机时间的长短，指定习题的全部或一部分作为上机题）。

学生应在实验前将教师指定的题目编好程序，然后上机输入和调试。

Scanned with CamScanner

第19章 实验安排

为了方便各校的教学，根据多数学校的情况，在本章中给出了学习C程序设计课程的上机实验的参考方案。根据教学要求，安排了12个实验。实验的内容与教学紧密结合，从每章的习题中选出一些作为上机题。教材中一章的内容对应1～2次实验。每次实验一般包括4个题目，其中有一题稍难。上机时间一般每周一次，每次2小时。各学校可以根据条件做必要的调整，增加或减少某些部分。在完成以上实验的基础上，最好能根据学生学习的情况，安排1～2次综合的训练，完成一两个有一定难度的程序。

19.1 实验1 C程序的运行环境和运行C程序的方法

1. 实验目的

(1) 了解所用的计算机系统的基本操作方法，学会独立使用该系统。

(2) 了解在该系统上如何编辑、编译、连接和运行一个C程序。

(3) 通过运行简单的C程序，初步了解C源程序的特点。

2. 实验内容和步骤

(1) 检查所用的计算机系统是否已安装了C编译系统并确定它所在的子目录。

(2) 进入所用的C编译集成环境。

(3) 熟悉集成环境的界面和有关菜单的使用方法。

(4) 输入并运行一个简单的、正确的程序。

① 输入下面的程序

```
#include <stdio.h>
int main()
{
  printf ("This is a C program.\n");
  return 0;
}
```

② 仔细观察屏幕上的已输入的程序，检查有无错误。

③ 根据本书第3部分介绍的方法对源程序进行编译，观察屏幕上显示的编译信息。如果出现“出错信息”，则应找出原因并改正之，再进行编译，如果无错，则进行连接。

④ 如果编译、连接无错误，可运行程序，观察分析运行结果。

(5) 输入并编辑一个有错误的C程序。

① 输入以下程序(教材第1章中例1.2，故意漏打或打错几个字符)。

Scanned with CamScanner

```
#include <stdio.h>
int main()
  {int a,b,sum
   a=123; b=456;
   sum=a+b
   print ("sum is %d\n",sum);
   return 0;
  }
```

② 进行编译,仔细分析编译信息窗口,可能显示有多个错误,逐个修改,直到不出现错误。最后与教材上的程序对照。

③ 使程序运行,分析运行结果。

(6) 输入并运行一个需要在运行时输入数据的程序。

① 输入下面的程序:

```
#include <stdio.h>
int main()
  {int max(int x,int y);
   int a, b, c;
   printf("input a&b: ");
   scanf ("%d,%d",&a,&b);
   c=max (a,b);
   printf ("max=%d\\n",c);
   return 0;
}

int max(int x,int y)
  {int z;
   if (x>y) z=x;
   else z=y;
   return (z);
  }
```

② 编译并运行,在运行时从键盘输入整数 2 和 5,然后按回车键,观察运行结果。

③ 将程序中的第 4 行改为

```
int a;b;c;
```

再进行编译,观察其结果。

④ 将 max 函数中的第 3,4 两行合并写为一行,即

```
if(x>y)z=x; else z=y;
```

进行编译和运行,分析结果。

(7) 运行一个自己编写的程序。题目是教材第 1 章的第 6 题。即输入 a,b,c 3 个值,输出其中最大者。

① 输入自己编写的源程序。

Scanned with CamScanner

② 检查程序有无错误(包括语法错误和逻辑错误),有则改之。

③ 编译和连接,仔细分析编译信息,如有错误应找出原因并改正之。

④ 运行程序,输入数据,分析结果。

⑤ 自己修改程序(例如故意改成错的),分析其编译和运行情况。

⑥ 将调试好的程序保存在自己的用户目录中,文件名自定。

⑦ 将编辑窗口清空,再将该文件读入,检查编辑窗口中的内容是否刚才存盘的程序。

⑧ 关闭所用的集成环境,用 Windows 中的"我的电脑"找到刚才使用的用户子目录,浏览其中文件,看有无刚才保存的后缀为 .c 和 .exe 的文件。

3. 预习内容

(1)《C 程序设计(第五版)》第 1 章。

(2) 本书第 3 部分的有关部分。

19.2 实验 2 数据类型、运算符和简单的输入输出

1. 实验目的

(1) 掌握 C 语言数据类型,了解字符型数据和整型数据的内在关系。

(2) 掌握对各种数值型数据的正确输入方法。

(3) 学会使用 C 的有关算术运算符,以及包含这些运算符的表达式,特别是自加(++)和自减(--)运算符的使用。

(4) 学会编写和运行简单的应用程序。

(5) 进一步熟悉 C 程序的编辑、编译、连接和运行的过程。

2. 实验内容和步骤

(1) 输入并运行教材第 3 章第 4 题给出的程序:

```
#include <stdio.h>
int main ()
 {char c1,c2;
  c1=97;
  c2=98;
  printf("%c %c\n"c1,c2);
  printf("%d %d\n",c1,c2);
  return 0;
 }
```

① 运行以上程序,分析为什么会输出这些信息。

② 如果将程序第 4,5 行改为

```
c1=197;
c2=198;
```

Scanned with CamScanner

运行时会输出什么信息？为什么？

③ 如果将程序第 3 行改为

```
int c1,c2;
```

运行时会输出什么信息？为什么？

(2) 输入第 3 章第 5 题的程序。即：用下面的 scanf 函数输入数据，使 a=3,b=7,x=8.5,y=71.82,c1='A',c2='a'。问在键盘上如何输入？

```
#include <stdio.h>
int main()
 {
  int a,b;
  float x,y;
  char c1,c2;
  scanf("a=%db=%d",&a,&b);
  scanf("%f%e",&a,&y);
  scanf("%c%c",&c1,&c2);
  return 0;
 }
```

运行时分别按以下方式输入数据，观察输出结果，分析原因。

① a=3,b=7,x=8.5,y=71.82,A,a↙

② a=3 b=7 x=8.5 y=71.82 A a↙

③ a=3 b=7 8.2 71.82 A a↙

④ a=3 b=7 8.5 71.82Aa↙

⑤ 3 7 8.5 71.82Aa↙

⑥ a=3 b=7↙
8.5 71.82↙
A↙
a↙

⑦ a=3 b=7↙
8.5 71.82↙
Aa↙

⑧ a=3 b=7↙
8.5 71.82Aa↙

通过此题，总结输入数据的规律和容易出错的地方。

(3) 输入以下程序：

```
#include <stdio.h>
int main()
 {int i,j,m,n;
  i=8;
  j=10;
  m=++i;
  n=j++;
```

Scanned with CamScanner

```
printf("%d,%d,%d,%d\n",i,j,m,n);
return 0;
}
```

① 编译和运行程序，注意 i,j,m,n 各变量的值。

② 将第 6,7 行改为

```
m=i++;
n=++j;
```

再编译和运行，分析结果。

③ 程序改为

```
#include <stdio.h>
int main()
{int i,j;
 i=8;
 j=10;
 printf("%d,%d\n",i++,j++);
}
```

再编译和运行，分析结果。

④ 在③的基础上，将 printf 语句改为

```
printf ("%d,%d\n",++i,++j);
```

再编译和运行。

⑤ 再将 printf 语句改为

```
printf ("%d,%d,%d,%d\n",i,j,i++,j++);
```

再编译和运行，分析结果。

⑥ 程序改为：

```
#include <stdio.h>
int main()
{int i,j,m=0,n=0;
 i=8;
 j=10;
 m+=i++;n-=--j;
 printf("i=%d,j=%d,m=%d,n=%d\n",i,j,m,n);
 return 0;
}
```

再编译和运行，分析结果。

(4) 假如我国国民生产总值的年增长率为 9%，计算 10 年后我国国民生产总值与现在相比增长多少百分比。编写程序(本题是教材第 3 章第 1 题)。

计算公式为：$p=(1+r)^n$

r 为年增长率，n 为年数，p 为与现在相比的倍数。

Scanned with CamScanner

① 输入自己编好的程序，编译并运行，分析运行结果。

② 年增长率不在程序中指定，改用 scanf 函数语句输入，分别输入 7%，8%，10%。观察结果。

③ 在程序中增加 printf 函数语句，用来提示输入什么数据，说明输出的是什么数据。

3. 预习内容

预习教材第 3 章。

19.3 实验 3 最简单的 C 程序设计——顺序程序设计

1. 实验目的

(1) 掌握 C 语言中使用最多的一种语句——赋值语句的使用方法。

(2) 掌握各种类型数据的输入输出的方法，能正确使用各种格式转换符。

(3) 进一步掌握编写程序和调试程序的方法。

2. 实验内容和步骤

(1) 通过下面的程序掌握各种格式转换符的正确使用方法。

① 输入以下程序：

```
#include <stdio.h>
int main()
  {int a,b;
   float d,e;
   char c1,c2;
   double f,g;
   long m,n;
   unsiguld int p,q;
   a=61;b=62;
   c1='a';c2='b';
   d=3.56;e=-6.87;
   f=3157.890121;g=0.123456789;
   m=50000;n=-60000;
   p=32768;q=40000;
   printf ("a=%d,b=%d\nc1=%c,c2=%c\nd=%6.2f,e=%6.2f\n",a,b,c1,c2,d,e);
   printf ("f=%15.6f,g=%15.12f\nm=%ld,n=%ld\np=%u,q=%u\n",f,q,m,n,p,q);
  }
```

② 运行此程序并分析结果。

③ 在此基础上，将程序第 10～14 行改为

```
c1=a;c2=b;
f=3157.890121;g=0.123456789;
d=f;e=g;
```

Scanned with CamScanner

p=a=m=50000;q=b=n=-60000;

运行程序,分析结果。

④ 用 sizeof 运算符分别检测程序中各类型的数据占多少字节。例如,int 型变量 a 的字节数为 sizeof(a)或 sizeof(int),用 printf 函数语句输出各类型变量的长度(字节数)。

(2) 设圆半径 $r=1.5$,圆柱高 $h=3$,求圆周长、圆面积、圆球表面积、圆球体积、圆柱体积。编程序,用 scanf 输入数据,输出计算结果。输出时要有文字说明,取小数点后两位数字(第 3 章第 7 题)。

(3) 计算存款利息(第 3 章第 2 题)。

有 1000 元,想存 5 年,可按以下 5 种办法存:

① 一次存 5 年期。

② 先存 2 年期,到期后将本息再存 3 年期。

③ 先存 3 年期,到期后将本息再存 2 年期。

④ 存 1 年期,到期后将本息存再存 1 年期,连续存 5 次。

⑤ 存活期存款。活期利息每一季度结算一次。

银行存款利率:请去银行查当日利率。

计算利息的公式见第 3 章第 2 题。

(4) 编程序将"China"译成密码,密码规律是:用原来的字母后面第 4 个字母代替原来的字母。例如,字母'A'后面第 4 个字母是'E',用'E'代替'A'。因此,"China"应译为"Glmre"。请编一程序,用赋初值的方法使 c1,c2,c3,c4,c5 这 5 个变量的值分别为'C','h','i','n','a',经过运算,使 c1,c2,c3,c4,c5 分别变为'G','l','m','r','e'。分别用 putchar 函数和 printf 函数输出这 5 个字符(第 3 章第 6 题)。

① 输入事先已编好的程序,并运行该程序。分析是否符合要求。

② 改变 c1,c2,c3,c4,c5 的初值为:'T'、'o'、'd'、'a'、'y',对译码规律做如下补充:'W'用'A'代替,'X'用'B'代替,'Y'用'C'代替,'Z'用'D'代替。修改程序并运行。

③ 将译码规律修改为:将一个字母被它前面第 4 个字母代替,例如'E'用'A'代替,'Z'用'U'代替,'D'用'Z'代替,'C'用'Y'代替,'B'用'X'代替,'A'用'V'代替。修改程序并运行。

3. 预习内容

预习教材第 3 章。

19.4 实验 4 选择结构程序设计

1. 实验目的

(1) 掌握逻辑量的方法(以 0 代表"假",以非 0 代表"真")。

Scanned with CamScanner