C语言 程序设计

高级程序设计（C 语言）
实验指导书

张晓民 张凌晓 杨彩霞 王爱兰 常进 编写

刘克成 审定

计算机基础实验中心

2003.02

高级程序设计（C语言）实验指导书

张晓民、张凌晓、杨彩霞、王爱兰、常进　编写

刘克成审定

计算机基础实验中心

2003．02

高级程序设计（C 语言）实验指导书

目　　录

高级程序设计（C语言）实验指导书

目　　录

使用说明

本实验指导书为全院所有开设高级程序设计语言（C 语言）课程的专业使用，各专业根据本专业的教学计划和实验任务书，选做规定的实验项目。全院各专业所做的实验项目具体说明和要求如下：

1、 计算机科学与技术系计算机应用专业（本科）（64/32 学时）必须完成本指导书的全部 16 个实验项目的内容；（2002 级开始执行）

2、 计算机科学与技术系软件专业（56/28 学时）必须完成本指导书的前 14 个实验项目的内容；（2002 级开始执行）

3、 计算机科学与技术系除计算机应用专业（本科）和软件专业（专科）外，其余各专业（40/20 学时）必须完成本指导书的实验一、二、三、五、六、八、九、十、十二、十四，共十个实验项目的内容。（2002 年级开始执行）

4、 全院非计算机专业（40/20 学时）必须完成本指导书的实验一、二、三、五、六、八、九、十、十二、十四，共十个实验项目的内容。（2002 年级开始执行）

计算机科学与技术系

2003-02-17

实验一　C 语言程序初步

一、目的和要求

1、熟悉 C 语言运行环境。

2、掌握 C 语言程序的书写格式和 C 语言程序的结构。

3、掌握 C 语言上机步骤，了解运行一个 C 程序的方法。

4、本实验可在学习完教材第一章后进行。

二、实验内容

1、C 语言上机步骤：

（A）、进入系统（以南阳理工院计算机实验中心 netware4.10 操作系统为例）

F:\>login ks 回车

H:\>ucdos 回车

H:\>tc 回车　　或者 H:\>tc 文件名（回车）

然后进行编辑源程序->编译->连接->执行程序->显示结果

（B）、常用命令

编辑切换（F6），编译（F9），运行（CTRL+F9），显示结果（ALT+F5）其它常用命令见“附录一”。

2、有下面的 C 程序，目的是想计算由键盘输入的任意两个整数的积。

/＊＊＊＊＊＊＊＊＊exl.c＊＊＊＊＊＊＊＊＊/

```
#include "stdio.h"
main()
{
scanf("%x,%y",&x,%y)
p=prodct(x,t)
printf("The product is :",p)
int prodct(int a ,int b )
int c
c=a*b
return c
```

请调试上述程序。

三、实验步骤

1、静态地检查上述程序，改正程序中的错误。

2、在编辑状态下照原样键入上述程序。

3、编译并运行上述程序，记下所给出的出错信息。

4、按照事先静态检查后所改正的情况，进行纠错。

5、再编译执行纠错后的程序。如还有错误，再编辑改正，直到不出现语法错误为止。

6、下面给出6组测试用例，你认为哪几组较好？为什么？

（1）0，　0

（2）0，　9 9

（3）2 0，5 0

（4）3 3 0 0 0，2 0

（5）－5，－2

（6）－5，2

要测试出上述程序中所有错误，你认为应当用几组测试用例？

四、分析与讨论

1、记下在调试过程中所发现的错误、系统给出的出错信息和对策。分析讨论对策成功或失败的原因。

2、总结C程序的结构和书写规则。

实验二　数据类型

一、目的和要求

1、了解C语言中数据类型的意义。

2、本实验可在学习了教材第2.3节后进行。

二、实验内容和步骤

1、下面的程序试图计算由键盘输入的任意两个整数的平均值：

```
#include <stdio.h>
main()
{
    int x,y,a;
    scanf("%x,%y,&x,&y);
    a=(x+y)/2;
    printf("The average is :"a);
}
```

调试无语法错误后，分别使用下列测试用例对上述程序进行测试：

（1）2，6

（2）1，3

（3）－2，－6

（4）－1，－3

（5）－2，6

（6）－1，3

（7）1，0

（8）1，6

（9）32800，33000

（10）－32800，33000

1、分析上述哪几组测试用例较好？通过测试，你发现程序有什么错误了吗？若有错误，请指出错误原因。

2、操作符sizeof用以测试一个数据或类型所占用的存储空间的字节数。请编写一个程序，测试各基本数据类型所占用的存储空间大小。

三、分析与讨论

如何正确地选用数据类型？（提示：给合前面做过的两个实验及书本进行讨论总结）

实验三　运行符与表达式

一、目的和要求

1、理解常用运行符的意义。

2、掌握C语言表达式的运行规则。

3、本实验可在学习了教材第2.3节后进行。

二、实验内容和步骤

1、编写一个C语言程序，测试下列各表达式

i,j

i+1,j+1

i++,j++

++i,++j

i+++++j

++i+++++j++

要求在各表达式中i　和j都分别有相同的初值。在实验中注意下列问题：

（1）　哪些表达式是错误的？为什么？

（2）　理解+，++，++i，i++的意义和优先级别。

2、编写一个程序，测试常用的十个运算符的优先顺序。

四、分析与讨论

分析总结运算符的优先级。

实验四　输入与输出

一、目的和要求

1、掌握C语言程序输入、输出的方法和格式。

2、本实验可在学习教材第二章后进行。

二、实验内容与步骤

1、输入并编辑下面的程序

```
main()
{
    int a,b;
    float c,d;
    long e,f;
    unsigned int u,v;
    char c1,c2;
    scanf("%d,%d",a,b);
    scanf("%f,%f",c,d);
    scanf("%ld,%ld",e,f);
    scanf("%o,%o",u,v);
    scanf("%c,%c",c1,c2);
    printf("\n");
    printf("a=%7d,b=%7d\n",&a,&b);
    printf("c=%10.2f,d=%10.2f\n",&c,&d);
    printf("e=17ld,f=%17ld\n",&e,&f);
    printf("u=%o,d=%o\n",&u,&v);
    printf("c1=%c,d=%c\n",&c1,&c2);
```

这个程序有语法错误吗？为什么？

2、调试上述程序无语法错误后，用下面的测试数据，对程序进行测试：

a=123,b=456,c=17.6,d=71837.65,e=70000,f=2174506,u=62000,v=58765,c1='a',c2='b'

分析运行结果。特别注意输入c1,c2的值是什么？什么原因？

3、将输入e和f的语句改为：

scanf("%d,%d",&e,&f);

再用上述测试数据测试并分析结果。

4、将输入u、v的语句改为：

scanf("%d,%d",&u,&v);

再用上述测试数据测试并分析结果。

5、将输出e　，f　的语句改为：

printf("e=%17d,f=%17d\n",e,f);

再用上述测试数据测试并分析结果。

6、将输出 u、v 的语句改为：

printf("u=%u,v=%u\n",u,v);

或

printf("u=%d,v=%d\n",u,v);

再用上述测试数据测试并分析结果。

7、请读者自己修改程序和改变数据输入的形式，分析各种情况下的输入与输出。

8、在 scanf("%c,%c",&c1,&c2)；语句之前加一个语句：

getchar();

9、验证转义字符\n 与 \r 的意义有何不同。

三、分析与讨论

1、总结在 printf 函数中可以使用的各种格式指定符，并给出样例。

2、总结在 printf 函数中可以使用的各转义字符及其功能。

实验五　选择结构

一、实验目的

1、了解条件与程序流程的关系

2、了解用不同的数据使程序的流程覆盖不同的语句、分支和路径。

3、本实验应在学习教材第 3.2 节后进行。

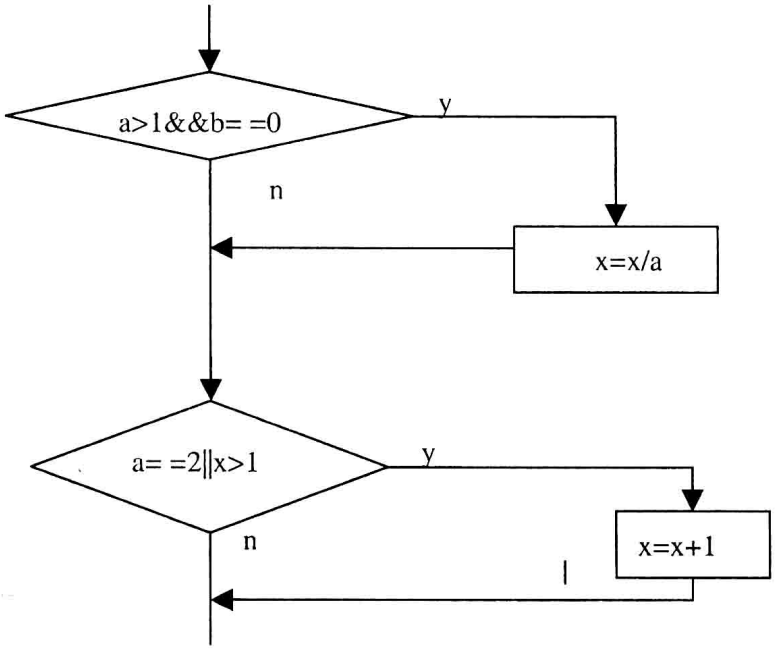
二、实验内容和步骤

1、题目

有如下程序段：

```
{……
        if (a>1&&b= =0) x=x/a;
        if(a==2||x>1) x=x+1;
}
```

为了更容易明白程序的逻辑结构，我们用图 5.1 所示流程图来加以描述。



要求增加一些输入语句和输出语句，以便使上述程序能在不同的 a,b 和 x 值下运行，并且能观察程序流程经过（覆盖）了哪些语句、哪些分支及哪些路径。

2、实验步骤

记下分别使用下列各组数据运行时的操作流程。

（1） a=1,b=1,x=1;

（2） a=1,b=1,x=2;

（3） a=3,b=0,x=1;

（4） a=2,b=1,x=4;

（5） a=2,b=1,x=1;

（6） a=1,b=0,x=2;

（7） a=2,b=1,x=1;

（8） a=3,b=0,x=2。

三、分析讨论

（1） 用哪一组数据就可使程序中的每个处理语句都执行一次？为了找出程序中各条处理语句中的错误，应该使用什么样的数据对程序进行测试？请上机验证自己的结论。

（2） 用哪两组数据就可以使程序段中的每个分支都运行一次？这种情形与上面的讨论有何不同？如果为了找出程序中积压分支中的错误，应该使用什么样的数据对程序进行测试？请上机验证自己的结论。

（3） 用哪两组数据就可以把判定框中的每个条件运算都进行一次？如果为了测试出判定条件中的错误，应使用哪些数据对程序进行测试？请上机验证自己的结论。

（4） 用哪四组数据才可以把各种条件的组合都检测一遍？ 如果为了测试各种条件的组合的情形，应该使用什么样的测试数据？请上机验证自己的结论。

（5） 用哪四组数据才可以把起始到终止的各条路径都覆盖一次？如果为了测试出程序在不同路径下的错误，应该使用什么样的测试数据？请上机验证自己的结论。

四、进一步的实验

1. 题目

从键盘上输入三个数，让它们代表三条线段的长度，请写一个判断这三条线段所组成的三角形属于什么类型（不等边，等腰，等边或不构成三角形）的C程序。

2. 请分别设计下列数据对自己的程序进行测试：

（1） 找出各条语句中的错误。

（2） 找出积压分支中的错误。

（3） 找出各条件中的错误。

（4） 找出各种条件组合中的错误。

（5） 找出各条路径中的错误。

实验六　条件型循环结构

一、目的和要求

1、掌握在程序设计条件型循环结构时，如何正确地设定循环条件，以及如何控制循环的次数。

2、了解条件型循环结构的基本测试方法。

3、本实验可在学习教材 3.3 节后进行。

二、实验内容与步骤

1、下面是一个计算 e 的近似值（使误差小于给定的 δ ）的程序。

```
main()
{double e=1.0,x=1.0,y,detax;
int i=1;
printf("\n please input enter a error:");
scanf("%lf",&detax);
y=1/x;
while(y>=detax)
{
x=x*I;
y=1/x;
e=e+y;
++i;
}
printf("%12.10lf",e);
}
```

2、实验步骤

（1）、阅读上面的程序，写出程序所依据的计算公式。

（2）、当输入的 detax 各是什么值时，能分别使程序按下面的要求运行：

.不进入循环；

.只循环一次；

.只循环两次；

.进入死循环(程序将永远循环下去)。

为了能知道程序循环了多少次，应该在程序中增加一条什么样的语句？

（3）、原程序中 while 语句中的 y>=detax，分别换成 y>detax,y=detax,y<detax,y<=detax，观察程序运行将会有什么变化。

假如不知道机器内的程序中的各语句实际上是什么，分别输入什么样的 detax 来测试出 while 语句的循环条件写错了。

（4）、把原程序中 while 语句之前的 y=1/x 语句去掉，观察程序的运行将会发生什么样

的变化。

假如不知道机器内的程序实际上是怎么写的，输入什么样的 detax 就能测试出少了上述这条语句。

（5）、若把原程序中的++i 换成 i++,观察程序的运行发生了什么变化？

假如不知道这条语句到底是怎么写的,输入什么样的 detax 就能测试出这条语句写错了.

（6）、把原程序中的 while 结构改写成 do—while 结构,再分别按上述的(2)、（3）两步进行实验。

三、分析讨论

总结一下测试条件循环结构的一般方法。