

四川师范大学数学与软件科学学院信息与计算科学专业 《程序设计基础教程----C语言版》课程实验 指导手册

本实验指导手册旨在指导数学与软件科学学院信息与计算科学专业《程序设计基础教程----C语言版》(专业必修课)的实验教学活动过程。其主要使用对象为本课程主讲教师和实验指导教师。它可以作为教学过程中的教学辅助参考资料,也可以作为学生在本门课程的学习中,结合实际教学内容,完成相关实验并填写实验报告的主要依据。具体实施过程中,除开各实验中要求的必做题目外,教师可结合实际教学情况,根据学生接受程度作适当扩展。学生在进行实践设计活动时,也可以结合自身掌握和学习的实际情况进行有选择地实践。

本指导手册仅针对本学院信息与计算科学专业而设置。

本实验手册编写的参考书籍:

1. 《程序设计基础教程----C语言版》,冯山,马廷淮 主编,北京:科学出版社,2008年5月
2. 《C语言程序设计教程(第二版)》,谭浩强,张基温,唐永炎 编著,北京:高等教育出版社,2002年3月
3. 《C语言习题集与上机指导(第二版)》,谭浩强,张基温 主编,北京:高等教育出版社,2002年1月

引言

程序设计是实践性很强的课程。实践也是学好程序设计课程中必不可少的重要环节。程序设计的实践分两个步骤:一是设计和编写解决问题的程序算法流程,并在此基础上编写源程序代码;二是在计算机上对程序进行调试,并使程序完成问题的求解过程,进行相应的逻辑测试。

实验过程中,应当注意如下问题:

(1) 上机实习的主要目的是训练C语言程序的上机调试能力和分析、解决调试过程中遇到的问题的能力。并在此过程中逐步提高程序编写与问题分析能力。而不是简单地对教材上的实例做验证式的操作;

(2) 上机内容可以结合教材内容和习题,或由老师布置专门的上机习题进行;

(3) 为提高上机效果。必须在上机之前准备好上机的基本内容并编写好上机程

序，尽量在上机之前先作好预先的阅读分析工作；

(4) 上机调试的基本方法：语法检查(编译) \Rightarrow 逻辑检查(设计者检查) \Rightarrow 测试。

上机实验的主要步骤：

(1) 了解实验目的和要求，并据此复习或阅读相关教材内容；

(2) 根据题目要求，事先编写好程序，并经过设计者静态检查程序的语法正确性和逻辑正确性，分析可能出现的问题；

(3) 设计好上机时准备测试程序的用例；

(4) 上机调试通过程序，并记录上机调试中出现的问题和测试结论等，写出实验报告。

调试过程中，尤其要注意各种语法错误、逻辑错误调试经验的积累。

实验报告的主要内容：

(1) 实验题目；

(2) 目的要求；

(3) 程序说明(如程序结构、各模块的算法和数据结构等)；

(4) 经调试后的程序清单；

(5) 测试情况说明；

(6) 分析(包含没有通过的程序的原因分析)。

实验报告的具体内容填写格式，参见《四川师范大学实验报告手册》格式的规定。其中，程序清单请附着在相应的实验单元后面，程序结果、说明与分析等写在实验结果处，如不够，可另行附页。

实验一 C 语言程序的运行环境和运行一个 C 语言程序的方法 (2 学时)

实验目的：

(1) 熟悉所用计算机系统的基本操作方法以及在该系统下安装、编辑、编译、连接和运行一个 C 语言源程序的基本操作步骤(以 Turbo C 为基本背景)；

(2) 学习基本的 C 语言程序设计与调试环境设置方法；

(3) 通过运行简单的 C 语言程序，了解 C 语言程序的特点。

实验内容：(要求至少调试 3 个以上的简单程序)

1. 如果系统没有安装 TC 环境，则安装，否则，跳过此步；

2. 配置好 TC 环境，熟悉 TC 编辑器的基本操作命令及其它基本功能；

3. 完成并熟悉程序输入、编译、连接、运行过程，直到程序输出结果正确；

4.学习基本的调试程序方法(包含编译、连接错误信息的正确理解);

5.可参考的程序实例如下(也可以参考教材范例):

1) 最简单的C语言程序:

```
int main(void)
{
    return 0;
}
```

2) 引用C语言函数库进行计算的程序:

```
/*Please try exercises in chapter one of textbook in page 15*/
```

```
int main(void)
{
    float fX;
    fX=sin(2.2922);

    return 0;
}
```

3) 引用自定义函数的C语言程序:

```
/*This is the function of average for data fA,fB,fC*/
```

```
float average(float fA,float fB,float fC)
```

```
{
    float fAvg;
    fAvg=(fA+fB+fC)/3.0;
    return fAvg;
}
```

```
int main(void)
```

```
{
    float x1,x2,x3;
```

```
        fX1,fX2,fX3,average(fX1,fX2,fX3));
```

```
    return 0;
```

```
}
```

4) 求 $1+2+3+\dots+100$

5) 给定5个数, 求其最小者;

6) 求 $a+|b|$ 。

实验准备:

(1) 计算机设备; (2) 程序调试环境的准备, 如TC环境; (3) 实验源程序准备。

实验步骤:

(1) 安装TC, 如果已安装好, 可以跳过此步;

- (2) 进入TC的编辑窗口，进行基本的环境设置；
- (3) 熟悉基本的TC编辑操作方法和文件创建、保存方法；
- (4) 输入一个C语言源程序，进行编辑、编译、连接、运行。完成一个简单的程序设计，运行输出正确结果；
- (5) 输入其它源程序并进行调试(注意错误提示信息)；
- (6) 编写实验报告。

实验结果：

- (1) 对TC的安装、调试、基本操作、环境设置方法等方面进行描述；
- (2) 对TC程序的执行过程和结果进行分析；
- (3) 给出实验过程中出现的主要问题和现象以及解决的方法；
- (4) 程序代码及运行结果描述。

实验二 数据类型、运算符和表达式(2学时)

实验目的：

- (1) 掌握C语言的数据类型的声明、定义、赋值及引用；
- (2) 掌握变量和常量之间的关系，以及各种常见的常量表示方法；
- (3) 掌握C语言的有关运算。算术运算中的自反、自增(++)、自减(--)运算符的使用；关系表达式和逻辑表达式的逻辑值表示方法和基本运算规律；学会在C语言环境下表示条件表达式；
- (4) 掌握不同类型数据之间的转换和赋值方法。

实验内容：

主要参考实例或内容(也可以参考教材或习题实例)：

1. 请变成实现教材P54页题2-4(目的：观察各种数据类型的分配空间大小)；
2. 教材相关内容为基础，考察变量的声明、定义、使用的先后关系；考察不同类型变量之间相互赋值的结果；考察各种形式的常量表示；
3. 基本的C语言运算规律，尤其是算术运算中的自反、自增、自减等运算和关系表达式、逻辑表达式的特殊运算规律的掌握。(可参考教材习题中的2-13, 2-14和2-18, 例2-20)；
4. 关于不同类型数据之间的转换问题，参见教材例2-5~例2-12。

注：由于本章内容是基本的程序设计基础内容，所有这些内容必须要求熟练掌握和运用。因此，其中一些实验实例由教师在课堂实践教学中进行现场示范性讲解，由学生课外自由验证课堂范例、教材范例以及教材习题和相关课外参考书籍和资料，

进行综合性的验证实践，并写出心得报告。具体实验范例不需要确定下来。

实验准备：

分析测试目的并根据它准备好实验内容的各种测试用例。

实验步骤：

- (1) 选择测试目标；
- (2) 准备测试用例和具体的测试数据；
- (3) 上机调试进行测试并做测试记录；
- (4) 进行测试分析；
- (5) 编写实验报告。

实验结果：

- (1) 根据测试目标写出测试报告；
- (2) 测试结果分析说明；
- (3) 测试程序代码清单。

实验三 最简单的 C 语言程序设计(2 学时)

实验目的：

- (1) 掌握顺序结构的程序设计概念、方法及各种赋值语句的使用方法；
- (2) 掌握基本的 C 语言程序设计中的输入、输出语句的使用方法。

实验内容：

1. 熟悉表3-1~表3-5中给出的格式化输入、输出格式控制符及其使用方法，并上机测试教材表例3-1~例3-10，观察 C 语言中的数据格式化输入、输出规律和方法；

2. 输入并编辑以下程序代码：

```
int main(void)
{
    int nA,nB;
    float fC,fD;
    long lE,lF;
    unsigned int u,v;
    char ch1,ch2;
```

```
    return 0;
}
```

请分析和观察代码有无错误，如果有，有哪些？指出错误原因，并修正之，使其可以按照预想的目的输入相应数据并输出结果；

3. 输入华氏温度的值F，要求输出摄氏温度的取值C，其中， $C=5/9*(F-32)$ 。要求输出有文字说明，取两位小数位；

4. 用getchar()函数读入两个字符ch1,ch2，然后分别用putchar()和printf()函数输出这两个字符。试分析思考如下问题：(a) ch1和ch2的数据类型用什么？(b) 要求输出ch1和ch2的ASCII码，应如何处理？是用printf()还是用putchar()？(c) 整型变量是否在任何情况下都可以相互代替？

5. *请编写和实现教材习题3-15,3-16,3-17和3-18，至少要求实现其中一个问题。

实验准备：

(1) 复习相关内容，阅读并分析1、2、3中的相关程序代码；

(2) 对4、5题进行程序代码编写。

实验步骤：

(1) 对照阅读分析结果，上机验证1、2、3的结果，并做出相应分析；

(3) 对4、5题进行上机调试，并进行测试、做测试记录；

(4) 进行实验分析；

(5) 编写实验报告。

实验结果：

(1) 根据测试目标写出测试报告；

(2) 测试结果分析说明；

(3) 测试程序代码清单。

实验四 分支结构的程序设计(4学时)

实验目的：

(1) 掌握C语言程序设计中逻辑量的表示和运用方法；

(2) 掌握C语言程序设计中条件表达式的值的计算方法及其跟程序执行流程之间的逻辑顺序关系；

(3) 掌握C语言中的4种选择结构语句的执行逻辑及其运用方法。

实验内容：

1. if 语句的实验。请分析如下语句段的逻辑关系：

```
int main(void)
{
    ...
    if (nA>1 && nB==0) nX=nX/nA;
    if (nA==2 || nX>1) nX=nX+1;
    ...
    return 0;
}
```

(a) 以算法流程图的形式表述；

(b) 增加输入、输出功能以补充完整该程序段，使之能够运行；

(c) 测试该程序段的各种条件组合下的运行结果。测试数据组如下：

nA	nB	nX
1	1	1
1	1	2
3	0	1
2	0	4
2	1	1
1	0	2
2	1	2
3	0	2

(d) 分析讨论：

1) 用哪组数据就可以使程序段中每个处理语句都执行一次？为了找出程序中的逻辑错误，应使用什么样的数据进行测试？请上机验证自己的分析结果。

2) 用哪两组数据可以使程序段中的每个分支都至少运行一次？为找出各个分支中的逻辑错误，应选用什么样的数据进行逻辑测试？请上机验证自己的分析结果。

3) 用哪四组数据可以将从起点到终点的各条路径都覆盖一次？为了测试程序在各个不同路径上的错误，应选用什么样的测试数据？请上机验证自己的分析结果。

2. if else 语句实验。用 if else 句型，编写求4个数中最大者的程序，并设计测试用例、记录测试结果；

3. 试用 if else if else if ...else 实现求4个数中最大者的程序，并做相应测试；

4. 试用 switch 语句句型实现以上程序，并做相应测试；

5. 请理清习题5-23的逻辑关系，上机进行验证。

实验准备：

- (1) 阅读并分析第1题中的逻辑关系及其分支测试方法；
- (2) 分析并编写2、3、4题(需要绘制流程图)的程序代码。

实验步骤：

- (1) 对照阅读分析结果，上机验证1题的测试结果，并做出相应分析；
- (2) 上机运行和测试所写的2、3、4、5题程序代码；
- (3) 进行实验分析；
- (4) 编写实验报告。

实验结果：

- (1) 根据测试目标写出测试报告；
- (2) 测试结果分析说明；
- (3) 测试程序代码清单。

实验五 循环控制程序的设计(4学时)

实验目的：

- (1) 复习和掌握算法的基本组成要素和描述特点；
- (2) 在实验四的基础上，学习几种常见的循环程序的流程关系和运用方法；
- (3) 不同循环类型之间的转换方法；
- (4) 循环的应用。

实验内容：(1和2必做，3和4选做)

1.循环实验(教材习题5-20)。具体要求：

- (1) 绘制流程图，编制程序代码。
- (2) 分析不进入循环、只循环一次、只循环两次、进入死循环的条件，以及此时各变量的值。
- (3) 分析循环退出时的条件表达式值。
- (4) 如何测试循环的次数？
- (5) 能否将程序改写成另外的形式，如while改写成do while或for语句？
- (6) 总结条件循环的设计、实现步骤和测试方法。

2.计数型循环实验(教材习题5-30)：九九表及其变体形状的打印。具体要求如下：

- (1) 分析并绘制九九表的程序实现代码(右上三角形)；
- (2) 分析程序的执行过程；
- (3)*请在此基础上修改部分程序代码以打印如下一些变形九九表：右下三角、左上三角、左下三角、倒序排列等形状，并观察、分析和总结其规律。
- (4) 分析并总结计数型循环在do while， while和for循环句型中的实现方法

和步骤。

3.其它实验题目：

- (1) 习题5-1(小孩吃梨问题)；
- (2) 习题5-4(水果拼盘问题)；
- (3) 求两个非负整数 u 和 v 的最大公因子；
- (4) 求两个非负整数的最小公约数；
- (5) 用牛顿迭代法求解正实数 a 的平方根；
- (6) 用梯形法求函数的数值积分；
- (7) 把由键盘输入的数列(1,3,5,7,9)按相反的顺序(如...9,7,5,3)输出。

4.习题5-27和习题5-28。

实验准备：

- (1) 编写所选实验题目的程序代码和流程图；
- (2) 检查源程序并给出其可能的运行结果说明。

实验步骤：

- (1) 上机验证所选题目，并做相应测试和分析；
- (2) 进行实验分析；
- (3) 总结循环程序设计、实现本方法及其经验；
- (4) 编写实验报告。

实验结果：

- (1) 根据测试目标写出测试报告；
- (2) 测试结果分析说明；
- (3) 测试程序代码清单。

实验六 基于函数的模块化程序设计(4 学时)

实验目的：

- (1) 了解和掌握C语言环境下模块化程序设计的基础----基于函数模块的程序设计技术；
- (2) 复习C语言程序的基本结构特点，了解基于函数模块的程序模块及其调用关系、过程；
- (3) 函数的声明、定义、使用方法；
- (4) 对各种变量属性的使用方法；
- (5) 对C语言中常用的预编译语句的基本掌握。

第 题任选其两个小题，第 2 题和第 3 题必做)

1.

(1) 编写一个在屏幕上画一条正弦曲线的函数；

(2) 编写一个函数，它可以求出给定数的所有因子，如下形式显示其结果：

72=2*2*2*3*3；

(3) 分别写出一个求N个数的最大公约数和最小公倍数的递归函数；

(4) 试编制n碟片hanoi塔的问题求解函数，并利用它求解4个盘片的结果；

(5) 试编写求n!的函数，并求n=7的结果。

具体要求如下：

(1) 编写程序通过调试获得正确结果；

(2) 分析函数的声明、定义和使用方法；

(3) 分别分析函数调用时的参数传值调用方法。

2.有关变量的属性实验(静态、动态、全局、局部等方面)。具体实例如下：

(1) 仔细分析以下程序的执行结果，然后上机运行如下程序段，并对各个变量的属性及其使用结果进行比较分析：

```
int main(void)
{
    /***(a)***/
    int nX=1;
    /***(b)***/
    void prt(void);
    int nX=3;
    prt();

    /***(c)***/

    return 0;
}
void prt(void)
{
    /***(d)***/
    int nX=5;

    /***(e)***/
}
```

(2) 仔细分析下列程序，是否有错误之处？然后分析其执行结果，上机验证并对各个变量的存储属性及其结果进行比较分析：

```
#define LOW 11
#define HIGH 23
#define CHANGE 12
int i=LOW;
int main(void)
```

```

int workover(int i);
int reset(int i);
int i=HIGH;
reset(i/2.5);

reset(i=i/3);

reset(i/2);

workover(i);

return 0;
}
workover(int i);
{
    i=(i%i++)*((i*i)/(2*i)+4);

    return (i);
}
reset(int i);
{
    i=i<=CHANGE?LOW:HIGH++;
    return (i);
}

```

(3)

指出各个变量的存储属性。

```

/*****file1.c*****/
int last(void);
int new(int);
int reset(void);
exter int i;
int main(void)
{
    int i,j;
    i=reset();
    for (j=1;j<=3;j++)
    {
        printf("next(i)=%d t",next(i));
        printf("last n",last());
        printf( "new(i+j)=%d n",new(i+j) );
    }
int next(void)
{
    return (i++);
}

```

```

}
/*****file2.c*****/
static int i=10;
fast(void)
{
    return (i-=1);
}
new(int i)
{
    static int j=-5;
    return (i=j+=++i);
}
/*****file3.c*****/
extern int i;
reset()
{
    return (i);
}

```

3.

- (1) 一般宏替换实验(教材例7-5);
- (2) 带参数的宏替换(教材例7-7~例7-12, 习题7-8和习题7-9);
- (3) 习题7-11和习题7-12;
- (4) 文件包含(`#include` 和 `#include <>`)。

- (1) 编写所选实验题目的程序代码和流程图;
- (2) 检查源程序并给出其可能的运行结果说明。

实验步骤:

- (1) 上机验证所选题目, 并做相应测试和分析;
- (2) 进行实验分析;
- (3) 总结基于函数、宏等技术的模块化程序设计和实现方法及其经验;
- (4) 编写实验报告。

实验结果:

- (1) 根据测试目标写出测试报告;
- (2) 测试结果分析说明;
- (3) 测试程序代码清单。

实验七 数组(6学时)

实验目的:

C语言环境下数组的声明、定义和使用方法，包含其初始化、赋值、输入、输出和处理方法；

(2) 掌握字符数组和字符串数组的使用方法；

(3) 了解数组作为函数参数或返回值的传递和使用方法。

第1题必做，第2、3、4任选一个)

1.数组的基本使用方法实验。包含数组的声明方法、初始化方法、输入、输出方法、元素的引用方法以及数组作为函数参数的使用方法。

(1) 一维数组的声明、初始化及引用方法；

(2) 二维数组的声明、初始化及引用方法；

(3) 字符数组和字符串数组的初始化及引用方法的差异分析；

)

测试函数声明和定义方法时，可以从以下几个方面进行测试：

用已声明和定义的变量作为数组大小的声明；

2) 用已声明的**const** 变量作为数组变量大小的声明；

3) 用宏名表示声明数组变量的大小；

4) 用表达式计算结果作为数组变量的大小；

5) 用宏的运算结果表示声明数组变量的大小。

(4) 分别按行序和列序顺序返回一个二维数组中各个元素的内存空间地址值，以观察数组元素在计算机内存中的排列规律。

2.一维数组的应用。具体实例如下：

(1) 有一个数组大小为10，内放10个数，请找出其中最大和最小元素来，并分别将其与数组第1个元素和最后1个元素交换。

(2) 有17个人围成一圈，其编号分别为1到17，如果从1号开始报数，凡报到7的倍数的人必须离开该圈子，然后再接着往下数，直到最后只剩一个人为止。请问最后一个人原来的位置序号是多少？（扩展问题：请将每个人按出列顺序重新排列后，其排列的顺序是什么？）

3.二维数组应用。具体实例如下：

(1) 输入一个**n*n**的矩阵。1) 试求除开对角线上元素之外的元素之和；2) 求每行中与相应对角线元素不同的元素之和；

(2) 有**n**个学生，**m**门功课。要求：1) 找出成绩最好的学生及其对应的课程号码；2) 求全部同学每门课程的平均成绩；

4.字符数组应用。具体实例：

(1) (注：单词

以一个或多个空格为间隔。如果一个单词刚好在行末结束，则下一行开头应该有空格，包含句号和逗号后面也应该有空格！)

(2) 输入n个国家的名字，要求按照字母的先后顺序排序，并输出排序前后的结果。

(1) 编写所选实验题目的程序代码和流程图；

(2) 检查源程序并给出其可能的运行结果说明。

实验步骤：

(1) 上机验证所选题目，并做相应测试和分析；

(2) 进行实验分析；

(3) 总结基于数组的程序设计和实现方法及其经验；

(4) 编写实验报告。

实验结果：

(1) 根据测试目标写出测试报告；

(2) 测试结果分析说明；

(3) 测试程序代码清单。

实验八 指针(4学时)

实验目的：

(1) 掌握C语言环境下指针的声明、定义和使用方法；

(2) 掌握指针和变量以及指针和数组的关系；

(3) 掌握指针、数组之间的关系；

(4) 掌握指针、函数之间的关系。

实验内容：

1. 熟悉指针的基本使用方法。

(1) 请仔细分析以下程序段，并上机测试运行结果，对测试结果进行分析说明。

1) 程序段一：

```
int main()
{
    int i=3,j=6,k=9;
    printf("i=%d j=%d k=%d\n",*(&i),j,k);
    return 0;
}
```

2) 程序段二：

```

    {
        int nA,*p;
        float fB,*q;
        p=&nA;
        q=&fB;

        *p=nA>(*q);
        printf("          n", nA, p,*p);
        printf("          n", nA, q,*q);
        printf("p+1=%d      q+1=%d n",p+1,q+1);
        return 0;
    }
3)
int main(void)
{
    int nA,*p,**q,***r;
    p=&nA;q=&p;r=&q;
    printf("Please input      n");
    scanf("%d",&r);
    printf("The results are :");
    printf(" nA value is %d,%d,%d,%d n",nA,*p,**q,***r);
    printf(" nA,p,q,r's value is %d,%d,%d,%d n",nA,p,q,r);
    return 0;
}

```

(2) 设计一个C语言程序，以测试以下表达式的运算顺序。

2. 想使指针变量pt1指向变量nA和nB中的大者，pt2指向其小者，以下程序能否实现此目的？为什么？如果不行，请给出实现的方法。

```

void swap(int *pt1,int *pt2)
{
    int *p;
    p=*pt1;*pt1=*pt2;*pt2=*p;
}
int main(void)
{
    int nA,nB;
    int *pt1,*pt2;
    scanf("%d,%d",&nA,&nB);
    pt1=&nA;pt2=&nB;
    if (nA<nB) swap(pt1,pt2);
    printf("%d,%d n",*pt1,*pt2);
    return 0;
}

```

3.请仔细分析教材例9-2、9-3、9-6~9-11，上机调试之。记录并分析程序运行结

果。

4.请仔细分析并执行以下程序段，并给出输入数据下的运行结果：

```
int main(void)
{
    int fun1();
    int fun2();
    int fun3();
    int fun4();
    int (*function[4])();
    int nA=10,nB=5,i;
    function[0]=fun1;
    function[1]=fun2;
    function[2]=fun3;
    function[3]=fun4;
    for (i=0;i<4;i++)
        printf("fun No.%d      n",i+1,execute( nA,nB,function[i]));
    return 0;
}

int execute(int nX,int nY,int *func()) {return ((*fun)(nX,nY));}
int fun1(int nX,int nY) {return (nX+nY);}
int fun2(int nX,int nY) {return (nX-nY);}
int fun3(int nX,int nY) {return (nX*nY);}
int fun4(int nX,int nY) {return (nX/nY);}
```

对如下main() 函数，假设源程序名为list.c，则对编译连接后的list.exe程序，在DOS界面运行时输入命令行时，其执行结果是什么？为什么？

```
/*Second main() program*/
int main(int argc,char *argv[])
{
    while (argc > 1) printf("%s  n",*++argv);
    return 0;
}
```

实验准备：

- (1) 编写所选实验题目的程序代码和流程图；
- (2) 检查源程序并给出其可能的运行结果说明。

实验步骤：

- (1) 上机验证所选题目，并做相应测试和分析；
- (2) 进行实验分析；
- (3) 总结指针及其程序设计、实现方法和经验；
- (4) 编写实验报告。

实验结果：

- (1) 根据测试目标写出测试报告；
- (2) 测试结果分析说明；

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/008023130042006023>