

三、注意事项及小结

(1) 通过本次实验,大家应该初步掌握 Visual C++ 6.0 集成开发环境的使用,包括集成开发环境的进入、设置、退出,源程序的打开、编辑、保存、编译、运行等内容,能够读懂系统提示的信息。这样在今后的学习中,就可以把主要精力放到程序的调试上,提高上机效率。

(2) 初学 C 语言程序,容易出错的地方是:

① C 语言程序中的标点符号都需是英文输入法下的,如分号、逗号、引号等。

② 语句后漏加分号。如图 2-1-9 中编译出错,因为语句 `printf("我们一起学习写 C 语言程序! \n")` 的最后未加分号。

③ 使用标识符时,混淆了变量中字母的大小写。如初学者可能会认为 `abc` 与 `Abc` 是同一个变量,实际 C 编译系统会把它们看作两个不同的变量。

④ 程序语句中括号不匹配。C 程序中的括号(如 `{}` 和 `()`),引号(如 `"` 和 `"`),都必须匹配,即成对出现,如:

例 2-1-1 中的源程序,若变为:

```
#include <stdio.h>
int main()
{
    printf("我们一起学习写 C 语言程序! \n");
    return 0;
}
```

则会出现如下错误提示:

G:\C 例题\例 2-1-1.c(4) : error C2143: syntax error : missing ')' before ';'

即提示在语句 `"printf("我们一起学习写 C 语言程序! \n");"` 中分号 `;` 之前少了括号 `)`。

实验二 数据类型、变量与表达式

一、实验目的与要求

1. 理解数据类型的基本概念。
2. 熟练掌握变量的使用方法。
3. 熟练掌握运算符与表达式的使用方法。

二、实验内容

1. 写出下列程序的运行结果,并上机验证。

(1)

```
#include <stdio.h>           //编译预处理指令
main()                       //主函数的函数头
{                             //函数体的开始标记
    int x=023;                //定义整型变量并赋初值
    printf("%d\n",--x);       //利用库函数的输出函数在屏幕上输出指定的信息
}                             //函数的结束标记
```

运行结果:

```
(2)
#include<stdio.h>           /*编译预处理指令*/
main()                     /*主函数的函数头*/
{                           /*函数体的开始标志*/
    int a,b,d=241;         /*定义整型变量 a,b,d=241*/
    a=d/100%9;            /*将 d/100%9 赋值给 a*/
    b=(-1)&&(-1);          /*将(-1)&&(-1)赋值给 b*/
    printf("%d,%d\n",a,b); /*利用库函数的输出函数在屏幕上输出 a,b 的值*/
}                           /*函数的结束标志*/
```

运行结果:

2. 找出程序中的错误并修改, 然后上机验证。

(1) 要求以下程序的输出结果是 $a=%d,b=%d$, 请找出错误并修改验证。

```
main()                     /*主函数的函数头*/
{                           /*函数的开始标志*/
    int a=2,c=5             /*定义整型变量 a=2,c=5*/
    printf("a=%d,b=%d\n",a,c); /*利用库函数的输出函数在屏幕上输出 a,c 的值*/
}                           /*函数的结束标志*/
```

(2) 要求下面程序的输出是 1, 请找出错误并修改验证。

```
#include<stdio.h>         /*编译预处理指令*/
main()                   /*主函数的函数头*/
{                         /*函数的开始标志*/
    int x=10,y=3;        /*定义整型变量 x=10,y=3*/
    printf("%d\n,y=x/y); /*利用库函数的输出函数在屏幕上输出 y 的值*/
}                         /*函数的结束标志*/
```

3. 阅读下列程序, 在横线处填入适当的语句, 使程序完整。

(1) 阅读下述程序功能: 将三位整数 n 的十位数的数字变为 0。例如, 输入三位整数为 738, 输出为 708。请将正确答案写在横线处。

```
#include<stdio.h>         /*编译预处理指令*/
void main()              /*主函数的函数头*/
{                         /*函数的开始标志*/
    ①;
    printf("输入一个三位整数: ");
    scanf("%d",&n);
    d2= ②;                /*取出百位数的数字*/
    d0= ③;                /*取出个位数的数字*/
    printf("输出三位整数: %d\n", ④);
}                         /*函数的结束标志*/
```

(2) 输入一个二元一次方程的 a 、 b 、 c 项, 并输出其两个根。

```
#include <stdio.h>       /*编译预处理指令*/
#include < ① >
void main()              /*主函数的函数头*/
```

```

{ /*函数的开始标志*/
    ② a, b, c, delta, p, q, x1, x2;
    scanf(" ③ ", &a, &b, &c); /*利用库函数的输入函数输入 a,b,c 值*/
    printf("%fx^2 + %fx + %f = 0\n", a, b, c); /*利用库函数的输出函数输出 a,b,c 值*/
    delta = ④;
    p = -b / (2 * a);
    q = ⑤;
    x1 = p+q; /* p+q 赋值给 x1*/
    x2 = p-q; /* p-q 赋值给 x2*/
    printf("x1 = %f\nx2 = %f\n", x1, x2); /*利用库函数的输出函数输出 x1, x2 值*/
} /*函数的结束标志*/

```

4. 编写程序。

- (1) 输入一个字符，分别输出其前导字符、该字符、后续字符。
- (2) 编写一个简单的 C 程序，输出以下信息：

```

*****
          C program!
*****

```

三、注意事项及小结

(1) 熟悉 C 语言的数据类型是学习 C 语言的一个主要基础，在此基础上掌握不同数据类型间的混合运算。

(2) 变量定义的意义在于计算机在编译时，对每一个变量分配对应和固定的存储单元（地址），为其程序的使用创造必要条件。变量初始化的意义在于，程序编译完成后，变量已被赋值。

- (3) 在使用各种运算符时，一定要掌握各类运算符的优先级与结合性。

实验三 顺序结构

一、实验目的与要求

1. 理解顺序结构程序设计。
2. 熟练掌握顺序结构程序中语句的执行过程。
3. 学会编写简单的顺序结构程序。

二、实验内容

1. 写出下列程序的运行结果，并上机验证。

(1)

```

#include "stdio.h" /*编译预处理指令*/
main() /*主函数的函数头*/
{ /*函数的开始标志*/
    int a=4; /*定义整型变量 a=4*/
    printf("a=%d",a); /*利用库函数的输出函数输出 a 值*/
}

```

```

    a=5;                /*赋值 a=5*/
    printf("\na=%d",a); /*利用库函数的输出函数输出 a 值*/
}                       /*函数的开始标志*/

```

运行结果:

(2)

```

#include "stdio.h"      /*编译预处理指令*/
main()                 /*主函数的函数头*/
{                       /*函数的开始标志*/
    printf("\n");
    printf("**\n");
    printf("***\n");    /*利用库函数的输出函数输出符号**/
    printf("****\n");
    printf("*****\n");
}                       /*函数的结束标志*/

```

运行结果:

2. 找出程序中的错误并修改, 然后上机验证。

(1) 输入一个小写字母, 将其变成大写字母并显示。请找出错误并修改验证。

```

#include <stdio.h>      /*编译预处理指令*/
main()                 /*主函数的函数头*/
{                       /*函数的开始标志*/
    char c1,c2;        /*定义字符变量 c1,c2*/
    printf("Please enter a lowercase letter:\n"); /*利用库函数的输出函数输出*/
    c1=getchar();      /*利用 getchar 函数输入赋值给 c*/
    c2=c1+32;          /*将 c1+32 结果赋值给 c2*/
    getchar(c2);
}                       /*函数的结束标志*/

```

(2) 要求以下程序的输出结果是 9, 10。请找出错误并修改验证。

```

#include<stdio.h>      /*编译预处理指令*/
main ()               /*主函数的函数头*/
{                       /*函数的开始标志*/
    int i=010,j=10;    /*定义整型变量 i,j 并赋值*/
    printf("%d,%d\n",i++,--j); /*利用库函数的输出函数输出 i++,--j*/
}                       /*函数的结束标志*/

```

3. 阅读下列程序, 在横线处填入适当的语句, 使程序完整。

(1) 下面程序要求先从键盘输入 a, 再在屏幕上输出 a 的值。

```

#include "stdio.h"      /*编译预处理指令*/
main()                 /*主函数的函数头*/
{                       /*函数的开始标志*/
    int a;              /*定义整型变量 a*/
    _____
    printf("a=%d",a);  /*利用库函数的输出函数输出 a*/
}

```

(2) 下面程序是将 a、b 两数对调。

```

#include "stdio.h"          /*编译预处理指令*/
main()                    /*主函数的函数头*/
{                          /*函数的开始标志*/
    int a=4,b=5;          /*定义整型变量 a=4,b=5*/
    int t;                /*定义整型变量 t*/
    printf("a=%d,b=%d",a,b); /*利用库函数的输出函数输出 a,b 的值*/
    t=a;                  /*将 a 的值赋给 t*/

    _____
    b=t;                  /*将 b 的值赋给 t*/
    printf("\na=%d,b=%d",a,b); /*利用库函数的输出函数输出 a,b 的值*/
}

```

4. 编写程序。

(1) 从键盘输入某商品的单价和数量，求出商品的总价并输出。

(2) 不用第三个变量，实现两个数对调。

(3) 一个程序用于水果销售算账。已知苹果每斤 2.5 元，鸭梨每斤 1.8 元，香蕉每斤 2 元，橘子每斤 1.6 元。要求输入各类水果的数量，输出应付的钱数，再根据顾客所付的钱数，输出应找的钱数。

三、注意事项及小结

(1) 使用简单的输入输出语句即可。

(2) 实现两个数对调，通常使用第三个变量，如 3. (2) 题。若要不用这第三个变量，只能通过已有两个数的运算得到第三个变量，如求和后再相减。

(3) 根据题意，依次输入，使用相应的公式计算后输出即可。

(4) 本实验重点要求大家掌握顺序结构程序设计。在顺序结构中，要清楚每一个语句执行结束之后每个变量值的情况。

实验四 简单选择结构程序设计

一、实验目的与要求

1. 理解 C 语言中逻辑量的表示方法。
2. 学会正确使用关系运算符、关系表达式、逻辑运算符和逻辑表达式。
3. 熟练掌握并使用关系表达式或逻辑表达式来表达条件。
4. 熟练掌握并使用 if 分支和 if-else 分支来解决简单的选择问题。

二、实验内容

1. 写出下列程序的运行结果，并上机验证。

(1)

```

#include "stdio.h"
main()

```

```

{
    int y=2,z=2;
    if(y-z)
        printf("####\n");
    else
        printf("****\n");
}

```

运行结果:

(2)

```

#include "stdio.h"
main()
{
    int a,b,c=1;
    a=(-1)|(-1);
    b=(-1)&&(-1);
    if(c=-1)
        printf("a=%d;b=%d;c=%d.",a,b,c);
    else
        printf("no answer.");
}

```

运行结果:

(3)

```

#include "stdio.h"
main()
{
    int a=3,b=4,c=5;
    printf("\n a<b<c =%d",a<b<c);
    printf("\n c>b>a =%d",c>b>a);
    printf("\n a<b && b<c =%d",a<b&&b<c);
}

```

运行结果:

(4)

```

#include "stdio.h"
main()
{
    int a=1,b=2,c=3,d=4,m=2,n=2;
    (m=a>b)&&(n=c>d);
    printf("%d\n",n);
}

```

运行结果:

2. 找出程序中的错误并修改，然后上机验证。

(1) 按照函数 $y = \begin{cases} 2x-1 & 0 < x < 1 \\ x & \text{other} \end{cases}$ ，根据输入 x 的值，计算相应的 y 值。

```
#include "stdio.h"
main()
{
    int x,y;
    scanf("%d",x);
    if(0<x<1)
        y=2x-1;
    else
        y=x;
    printf("x=%d,y=%d",x,y);
}
```

(2)

```
#include "stdio.h"
main()
{
    int m=5;
    if(m++>5);
        printf("%d\n",&m);
    else
        printf("%d\n",&m--);
}
```

3. 阅读下列程序，在横线处填入适当的语句，使程序完整。

(1) 下面程序输出三个数中的最大值。

```
#include "stdio.h"
main()
{
    int a=3,b=4,c=5;
    ①
    if( ② )
        max=a;
    else
        max=b;
    if(max<c)
        ③
    printf("the max is %d.",max);
}
```

(2) 以下程序实现：输入三个整数，按从小到大的顺序进行输出。

```
#include "stdio.h"
main()
{
```

```

int a,b,c,x;
scanf("%d %d %d",&a,&b,&c);
if(①)
{
    x=a;
    a=b;
    b=x;
}
if(②)
{
    x=a;
    a=c;
    c=x;
}
if(③)
{
    x=b;
    b=c;
    c=x;
}
printf("the order is %d,%d,%d.",a,b,c);
}

```

(3) 以下程序用于判断用户输入的数是奇数还是偶数。

```

#include "stdio.h"
main()
{
    int n,remainder;
    printf("Enter your number to be tested.\n");
    scanf("%d",&n);
    remainder=n%2;
    if(①)
        printf("%d is even.\n",n);
    if(②)
        printf("%d is odd.\n",n);
}

```

4. 编写程序。

(1) 输入一个字符，若是小写字母，则转换成大写字母输出；若是大写字母，则转换成小写字母输出。

提示：大写字母和小写字母的 ASCII 码值相差 32，例如 a 的码值是 97，A 的码值是 65，所以大写字母转换成小写字母相当于 ASCII 码值加 32。

(2) 编写程序，输入两个整数和一个字符，如果该字符是 y，则进行两个整数的交换并输出交换后的结果，否则，输出字符串 “No swap!”

提示：两个数进行交换需要引入第三个变量。

(3) 有一函数

$$y = \begin{cases} x & x < 0 \\ 2x - 1 & 0 \leq x < 10 \\ 3x - 1 & x \geq 10 \end{cases}$$

编程要求:

- ①输入 x , 数据类型为浮点型。
- ②输出 y 值, 要求屏幕打印出: $x=??$ 时, $y=###.##$ 。(??表示输入值, ###.##表示输出结果, 小数部取两位)。
- ③程序中加入三处注释, 说明 x 取不同值时, 相应的计算公式。

三、注意事项及小结

- (1) 当 if 语句后面的表达式的值非 0 时, 都代表条件成立, 只有为 0 时, 条件不成立。
- (2) 逻辑运算符 && 只有在左边为真时, 才进行右边的计算。
- (3) 执行复合语句时, 一定要写 {}, 把要执行的语句括起来。

实验五 多分支选择结构与选择嵌套程序设计

一、实验目的与要求

1. 学会使用 if 条件语句中的多分支形式: if-else if-else。
2. 熟练掌握 if 条件语句三种形式的嵌套使用。
3. 熟练掌握 switch 语句。
4. 在 if 多分支、if 嵌套和 switch 语句执行过程中, 要注意程序段中语句的执行过程。

二、实验内容

1. 写出下列程序的运行结果, 并上机验证。

(1) 运行程序时, 输入 3, 1<回车>

```
#include "stdio.h"
main()
{
    int a,b,s,t;
    scanf("%d,%d",&a,&b);
    s=t=1;
    if(a>0)
        s=s+1;
    if(a>b)
        t=s+t;
    else if(a==b)
        t=5;
    else
        t=2*s;
```

```

        printf("s=%d,t=%d",s,t);
    }

```

运行结果:

(2)

```

#include "stdio.h"
main()
{
    int a=100,x=10,y=20,m=5,n=0;
    if(x<y)
        if(y!=m)
            a=1;
        else
            if(n)
                a=10;
    a=-1;
    printf("%d\n",a);
}

```

运行结果:

(3) 运行 5 次, 分别输入 1、2、3、4、5。

```

#include "stdio.h"
main()
{
    int i;
    printf("enter i:");
    scanf("%d",&i);
    switch(i)
    {
        case 1:
        case 2:printf("i=%d\n",i);
        case 3:printf("i=%d\n",i);break;
        default:printf("default!\n");
    }
}

```

运行结果:

(4)

```

#include "stdio.h"
main()
{
    int x=1,a=0,b=0;
    switch(x)
    {
        case 0: b++;

```

```

        case 1: a++;
        case 2: a++;b++;
    }
    printf("a=%d,b=%d\n",a,b);
}

```

运行结果:

2. 找出程序中的错误并修改, 然后上机验证。

(1) 按照函数 $y = \begin{cases} 0 & x < 0 \\ 1/x & 0 < x < 10 \\ 1 & \text{other} \end{cases}$, 根据输入 x 的值, 计算相应的 y 值。

```

#include "stdio.h"
main()
{
    float x=2.0,y;
    scanf("%f",x);
    if(x<0.0);
        y=0.0;
    else if(x<10.0)
        y=1.0/x;
    else
        y=1.0;
    printf("%f\n",y);
}

```

(2)

```

#include "stdio.h"
main()
{
    int a=2,b=7,c=5;
    switch(a+b);
    {
        case 1.0: printf("@");break;
        case 2: printf("!");break;
        case 3,4: printf("*");break;
        default:printf("?");break;
    }
    printf("\n");
}

```

3. 阅读下列程序, 在横线处填入适当的语句, 使程序完整。

(1) 根据以下函数关系, 对输入的几个 x 值, 计算相应的 y 值。

$$y = \begin{cases} 0 & x < 0 \\ x(x+1) & 0 \leq x \leq 10 \\ x-1 & 10 < x \leq 20 \\ 2x & x > 20 \end{cases}$$

```

#include "stdio.h"
main()
{
    int x,y;
    scanf("%d",&x);
    if(①)
    {
        y=0;
        printf("x<0,y");
    }
    else if(②)
    {
        y=x*(x+1);
        printf("y=x(x+1)");
    }
    else if(③)
    {
        y=x-1;
        printf("y=x-1");
    }
    else
    {
        ④;
        printf("y=2x");
    }
    printf("=%d",y);
}

```

(2) 以下程序的功能是计算某年某月有几天。

```

#include "stdio.h"
main()
{
    int y,m,len;
    printf("year,month=");
    scanf("%d,%d",&y,&m);
    switch(m)
    {
        case 1:
        case 3:
        case 5:
        case 7:
        case 8:
        case 10:
        case 12: ① break;
        case 4:
        case 6:
        case 9:

```

```

        case 11:len=30;break;
    case 2:
        if(②)
            len=29;
        else
            len=28;
        ③
    default: printf("the month is error!");
}
printf("the length of %d year %d month is %d\n",y,m,len);
}

```

(3) 以下程序根据输入的三角形的三边判断是否能组成三角形, 若可以则输出它的面积和三角形的类型。

```

#include "stdio.h"
#include "math.h"
main()
{
    float a,b,c;
    float s,area;
    scanf("%f%f%f",&a,&b,&c);
    if(①)
    {
        s=(a+b+c)/2;
        area=sqrt(s*(s-a)*(s-b)*(s-c));
        printf("%f",area);
        if(②)
            printf("等边三角形");
        else if(③)
            printf("等腰三角形");
        else if((a*a+b*b==c*c)||((a*a+c*c==b*b)||((c*c+b*b==a*a)))
            printf("直角三角形");
        else
            printf("一般三角形");
    }
    else
        printf("不能组成三角形");
}

```

4. 编写程序

(1) 从键盘输入一个字符, 当该字符是+、-、*或/时, 显示其对应的英文单词: plus、minus、multiplication 或 division。输入其他字符时, 显示 Error!。

提示: 判断是否相等用“==”, 字符用'给出, 例如: if(c=='+')。

(2) 某物品原有价值为 p, 由于使用其价值降低, 价值的折扣率根据时间 t (月数) 确定如下:

$$\left\{ \begin{array}{ll} t < 3 & \text{无折扣} \\ 3 \leq t < 6 & \text{2\%折扣} \\ 6 \leq t < 12 & \text{5\%折扣} \\ 12 \leq t < 21 & \text{8\%折扣} \\ t \geq 21 & \text{10\%折扣} \end{array} \right.$$

编写程序，输入时间和原有的价值计算物品的现有价值。

提示：可使用 switch 语句编程，将时间 t 与折扣率之间的关系转换成某些整数与折扣率的关系。

(3) 某地出租车的收费方法如下：起步价 8 元，最多可行驶 3 公里（不包含 3 公里）；3~8 公里（不包含 8 公里）按 1.8 元/公里计算（不足 1 公里，按 1 公里计算），8 公里以后按 2.0 元/公里计算（不足 1 公里，按 1 公里计算）。编写程序，计算并输出车费。

提示：可使用 if-else 语句编程，根据题意计费分为三个区间，分别是(0,3)，[3,8)和大于 8。

三、注意事项及小结

本实验是对选择结构的综合应用，同时与现实生活中的情况紧密结合。在练习时注重对问题的分析，多分支形式一定要把每个分支弄清楚后，再进行编程。

实验六 简单循环结构程序设计

一、实验目的与要求

1. 熟练掌握 while、do-while 和 for 三种循环语句的使用。
2. 掌握利用循环语句求解问题的常用算法。
3. 练习程序的跟踪调试技术。

二、实验内容

1. 写出下列程序的运行结果，并上机验证。

(1)

```
#include<stdio.h>
main()
{
    int num=0;
    while(num<=2)
    {
        num++;
        printf("%d\n",num);
    }
}
```

运行结果：

第 3 部分 上机实验答案及参考程序

实验二 答案及参考程序

1.

(1) 18

(2) 2,1

2.

(1) 程序开始缺头文件#include<stdio.h>;

缺分号, 应将 int a=2,c=5 改为 int a=2,c=5;。

(2) 实际上是因为 printf 函数中缺失了一个双引号", 使得系统无法正确判断, 提示缺少函数及语句结束标志。将"%d\n 改为"%d\n", 将 y=x/y 改为 y=x%y。

3.

(1) ①int n,d2,d0 ②n/100 ③n%10 ④d2*100+d0

(2) ①math.h ②float ③%f%f%f ④sqrt(b*b-4*a*c) ⑤delta/(2*a)

4.

(1)

```
#include <stdio.h>
void main()
{
    char c;
    printf("输入一个字符:"); scanf("%c",&c);
    printf("前导字符: %c\n  该字符: %c\n  后续字符: %c\n",c-1,c,c+1 );
}
```

(2)

```
#include <stdio.h>
void main ( )
{
    printf("*****\n");
    printf("    C program!\n");
    printf("*****\n");
}
```

实验三 答案及参考程序

1.

(1)

a=4

a=5

(2)

*

**

2.

(1) 将 `c2=c1+32`;改为 `c2=c1-32`;将 `getchar(c2)`;改为 `putchar(c2)`;(2) 将程序中 `printf("%d,%d\n",i++,--j)`;改为 `printf("%d,%d\n",++i,j--)`;

3.

(1) `scanf("%d",&a)`;(2) `a=b`;

4.

(1)

```
#include "stdio.h"
main()
{
    int p,x;
    scanf("%d",&x);
    scanf("%d",&p);
    printf("the total is %d",x*p);
}
```

(2)

```
#include "stdio.h"
main()
{
    int a=4,b=5;
    printf("a=%d,b=%d",a,b);
    a=a+b;
    b=a-b;
    a=a-b;
    printf("\na=%d,b=%d",a,b);
}
```

(3)

```
#include "stdio.h"
```



```

main()
{
    float apple,pear,banana,orange,cost,payin,payout;
    printf("apple,pear,banana,orange =");
    scanf("%f,%f,%f,%f",&apple,&pear,&banana,&orange);
    cost=2.5*apple+1.8*pear+2*banana+1.6*orange;
    printf("Pay=%.2f",cost);
    scanf("%f",&payin);
    payout=payin-cost;
    printf("payin =%.2f, payout =%.2f ",payin,payout);
}

```

实验四 答案及参考程序

1.

(1) ****

(2) a=1;b=1;c=-1.

(3)

a<b<c =1

c>b>a =0

a<b && b<c =1

(4) 2

分析: 首先计算 $m=a>b$, 因为关系运算符 $>$ 优先级高于赋值运算符 $=$, $a>b$ 的结果为假 (0), 赋值后 m 的值为 0。因为 C 语言计算逻辑表达式 $0 \&\&(n=c>d)$ 时, 已经知道结果为 0, 因此 $n=c>d$ 就不计算了。 n 的值仍为 2。

2.

(1) #include "stdio.h"

```

main()
{
    int x,y;           ①
    scanf("%d",x);    ②
    if(0<x<1)         ③
        y=2x-1;      ④
    else
        y=x;
    printf("x=%d,y=%d",x,y);
}

```

错误答案:

① int x,y;

② scanf("%d",&x);

③ if(0<x&& x<1)

④ y=2*x-1;

(2)

```
#include "stdio.h"
main()
{
    int m=5;
    if(m++>5);           ①
        printf("%d\n",&m); ②
    else
        printf("%d\n",&m--); ③
}
```

错误答案:

- ① if(m++>5)
- ② printf("%d\n",m)
- ③ printf("%d\n",m--);

3.

- (1) ① int max; ② a>b ③ max=c;
- (2) ① a>b ② a>c ③ b>c
- (3) ① remainder==0; ② remainder!=0

4.

(1)

```
#include "stdio.h"
main()
{
    char ch;
    printf("Please enter a character:\n");
    scanf("%c",&ch);
    if(ch>='A'&&ch<='Z')
        printf("%c\n",ch+32);
    if(ch>='a'&&ch<='z')
        printf("%c\n",ch-32);
}
```

(2)

```
#include "stdio.h"
main()
{
    char ch;
    int a,b,t;
    printf("Please enter two number and a character:\n");
    scanf("%d %d %c",&a,&b,&ch);
    printf("ch=%c,a=%d,b=%d\n",ch,a,b);
    if(ch=='y')
    {
        t=a;a=b;b=t;
        printf("a=%d,b=%d\n",a,b);
    }
}
```

```

        }
        else
            printf("No swap!\n");
    }
(3)
#include "stdio.h"
main()
{
    float x,y;
    printf("Please enter a float number :\n");
    scanf("%f",&x);
    if(x<0)
        y=x; //x<0,y=x
    if(0<=x&&x<10)
        y=2*x-1;//0<=x<10,y=2x-1
    if(x>=10)
        y=3*x-1;//x>=10,y=3x-1
    printf("x=%f,y=%.2f\n",x,y);
}

```

实验五 答案及参考程序

1.

(1) s=2,t=3

(2) -1

(3)

输入 1, 运行结果为

i=1

i=1

输入 2, 运行结果为

i=2

i=2

输入 3, 运行结果为

i=3

输入 4, 运行结果为

default!

输入 5, 运行结果为

default!

(4)

a=2,b=1

2.

(1)

```
#include "stdio.h"
main()
{
    float x=2.0,y;
    scanf("%f",x);      ①
    if(x<0.0);          ②
        y=0.0;
    else if(x<10.0)
        y=1.0/x;
    else
        y=1.0;
    printf("%f\n",y);
}
```

错误答案:

① scanf("%f",&x);

② if(x<0.0)

(2)

```
#include "stdio.h"
main()
{
    int a=2,b=7,c=5;
    switch(a+b);        ①
    {
    case 1.0: printf("@");break; ②
    case 2: printf("!");break;
    case 3,4: printf("*");break; ③
    default:printf("?");break;
    }
    printf("\n");
}
```

错误答案:

① switch(a+b)

② case 1: printf("@");break;

③ case 3:

case 4: printf("*");break;

3.

(1) ①x<0 ②x<=10 ③x<=20 ④y=2*x;

(2) ①len=31; ②y%4==0&& y%100!=0||y%400==0 ③break;

(3) ①a+b>c&& b+c>a&& a+c>b ②a==b&& b==c ③a==b||b==c||a==c

4.

(1)

```
#include "stdio.h"
main()
{
    char c;
    c=getchar();
    if(c=='+')
        printf("plus\n");
    else if(c=='-')
        printf("minus\n");
    else if(c=='*')
        printf("multiplication\n");
    else if(c=='/')
        printf("division\n");
    else
        printf("Error!\n");
}
```

(2)

```
#include "stdio.h"
main()
{
    int t,d;
    float p;
    scanf("%d,%f",&t,&p);
    switch(t/3)
    {
        case 0:d=0;break;
        case 1:d=2;break;
        case 2:
        case 3:d=5;break;
        case 4:
        case 5:
        case 6:d=8;break;
        default:d=10;
    }
    printf("Price=%f\n",p*(1-d/100.0));
}
```

(3)

```
#include "stdio.h"
main()
{
    float distance,fee;
    printf("Please enter the distance:\n");
    scanf("%f",&distance);
    if(distance<0)
```

```

        printf("Enter error!\n");
    else if(distance<3)
        fee=8;
    else if(distance<8)
        fee=8+(int)(distance-2)*1.8;
    else
        fee=8+5*1.8+(int)(distance-7)*2.0;
    printf("The taxi fee is %.2f\n",fee);
}

```

实验六 答案及参考程序

1.

(1)

1
2
3

(2)

0

(3)

#&
&
&*

2.

(1) 程序中有以下几处错误:

①语句 `while(k=0)` 中 `k=0` 是赋值语句, 不是逻辑判断语句, 这个语句在每次执行时都将 `k` 的值赋值为 0, 然后判断不满足条件, 结束循环。在本题中循环的条件是当 `k` 值不为 0 时执行循环, 因此应该为 `while(k!=0)`

②在循环中没有改变循环控制变量的取值, 根据本题题意, 应在循环体中添加 `k--;` 语句。
修改后参考程序:

```

#include <stdio.h>
main()
{
    int k=100;
    while(k!=0)
    {
        printf("%5d",k);
        k--;
        if(k%10==0)
            printf("\n");
    }
}

```