

第五十五节： 函数的作用和四种常见书写类型。

【55.1 函数和变量的命名规则。】

函数的名字和变量的名字一样，一般是由“字母，数字，下划线”三者组成。第1个字符不能是数字，必须是字母或者下划线“_”，后面紧跟的第2个字符开始可以是数字。在C语言中名字所用的字母是区分大小写的。可以用下划线“_”，但是不可以用横杠“-”。名字不能跟C编译系统已经征用的关键字重名，比如不能用“unsigned”，“char”，“static”等系统关键词，跟古代时不能跟皇帝重名一样，要避尊者讳。

【55.2 函数的作用和分类。】

函数的作用。通常把一些可能反复用到的算法或者过程封装成一个函数，函数就是一个模块，给它输入特定的参数，就可以输出想要的结果，比如一个加法函数，只要输入加数和被加数，然后就会输出相加结果之和，里面具体的算法过程只要写一次就可以重复调用，极大的节省单片机程序容量，也节省程序开发人员的工作量。还有一类函数，它从封装上看无所谓“输入输出”，这类函数往往是针对某一种可能重复使用的“过程”。

函数的分类。暂时排除指针的情况下（指针的内容后续章节会讲到），从输入输出的角度来看，有四种常见的书写类型。分别是“无输出无输入，无输出有输入，有输出无输入，有输出有输入”。“输出”是看函数名的前缀，前缀如果是void表示“无输出”，否则就是“有输出”。“输入”是看函数名括号里的内容，如果是void或者是空着就表示“无输入”，否则就是“有输入”。“输出”和“输入”是比较通俗的说法，专业一点的说法是，“有输出”表示函数“有返回”，“无输出”表示函数“无返回”。“有输入”表示函数“有形参”，“无输入”表示函数“无形参”。下面举一个加法函数的例子，分别用四种不同的函数类型来实现，通过对比它们之间的差别，来体会它们在书写方面有哪些不同，又有哪些规律。

【55.3 第1类：“无输出”“无输入”的函数。】

```
unsigned char a; //此变量用来接收最后相加结果的和。
unsigned char g=2;
unsigned char h=3;
void HanShu(void) //“无输出”“无输入”函数的定义。
{
    a=g+h;
}
main()
{
    HanShu(); //函数的调用。此处括号内的形参 void 要省略，否则编译不通过。
}
```

分析：void HanShu(void)，此函数名的前缀是void，括号内也是void，属于“无输出”“无输入”函数。这类函数表面看是“无输出”“无输入”，其实内部是通过全局变量来输入输出的，比如上面的例子就是靠a, g, h这三个全局变量来传递信息，只不过这类表达方式比较隐蔽，没有那么直观。

【55.4 第2类：“无输出”“有输入”的函数。】

```

unsigned char b; //此变量用来接收最后相加结果的和。
void HanShu(unsigned char i,unsigned char k) //“无输出”“有输入”函数的定义。
{
    b=i+k;
}
main()
{
    HanShu(2,3); //函数的调用。
}

```

分析：void HanShu(unsigned char i,unsigned char k)，此函数名的前缀是 void，括号内是(unsigned char i,unsigned char k)，属于“无输出”“有输入”的函数。括号的两个变量 i 和 k 是函数内的局部变量，也是跟对外的桥梁接口，它们有一个专业的名称叫形参。外部要调用此函数时，只要给括号填入对应的变量或者数值，这些变量和数值就会被复制一份传递给作为函数形参的局部变量（比如本例子中的 i 和 k），从而外部调用者跟函数内部就发生了数据信息的传递。这种书写方式的特点是把输入接口封装了出来。

【55.5 第3类：“有输出”“无输入”的函数。】

```

unsigned char c; //此变量用来接收最后相加结果的和。
unsigned char m=2;
unsigned char n=3;
unsigned char HanShu(void) //“有输出”“无输入”函数的定义。
{
    unsigned char p;
    p=m+n;
    return p;
}
main()
{
    c=HanShu(); //函数的调用。此处括号内的形参 void 要省略，否则编译不通过。
}

```

分析：unsigned char HanShu(void)，此函数名的前缀是 unsigned char 类型，括号内是 void，属于“有输出”“无输入”的函数。函数前缀的 unsigned char 表示此函数最后退出时会返回一个 unsigned char 类型的数据给外部调用者。而且这类函数内部必须有一个 return 语句配套，表示立即退出当前函数并且返回某个变量或者常量的数值给外部调用者。这种书写方式的特点是把输出接口封装了出来。

【55.6 第4类：“有输出”“有输入”的函数。】

```

unsigned char d; //此变量用来接收最后相加结果的和。
unsigned char HanShu(unsigned char r,unsigned char s) //“有输出”“有输入”函数的定义
{
    unsigned char t;
    t=r+s;
}

```

```

        return t;
    }
    main()
    {
        d=HanShu(2,3); //函数的调用。
    }

```

分析：unsigned char HanShu(unsigned char r,unsigned char s)，此函数名的前缀是 unsigned char 类型，括号内是(unsigned char r,unsigned char s)，属于“有输出”“有输入”的函数。输入输出的特点跟前面介绍的函数一样，不多讲。这种书写方式的特点是把输出和输入接口都封装了出来。

【55.7 函数在被“调用”时需要注意的地方。】

函数的三要素是“声明，定义，调用”。函数在被“调用”的时候，对于“无输入”的函数，形参的 void 关键词要省略，否则编译不通过，这里仅仅是指在函数在被“调用”的时候。

【55.8 例程练习和分析。】

现在编写一个练习程序，要求编写 4 个不同“输入输出”封装的函数，它们每个函数所实现的功能都是一样的，都是加法的算法函数，它们之间仅仅是外观的封装接口不同而已。

```

/*---C 语言学习区域的开始。-----*/

void hanshu_1(void);
void hanshu_2(unsigned char i,unsigned char k);
unsigned char hanshu_3(void);
unsigned char hanshu_4(unsigned char r,unsigned char s);

unsigned char a;    //此变量用来接收第 1 个函数最后相加结果的和。
unsigned char g=2;
unsigned char h=3;

unsigned char b;    //此变量用来接收第 2 个函数最后相加结果的和。

unsigned char c;    //此变量用来接收第 3 个函数最后相加结果的和。
unsigned char m=2;
unsigned char n=3;

unsigned char d;    //此变量用来接收第 4 个函数最后相加结果的和。

void hanshu_1(void) //第 1 类：“无输出”“无输入”。
{
    a=g+h;
}

```

```

void hanshu_2(unsigned char i,unsigned char k) //第2类：“无输出”“有输入”。
{
    b=i+k;
}

unsigned char hanshu_3(void) //第3类：“有输出”“无输入”。
{
    unsigned char p;
    p=m+n;
    return p;
}

unsigned char hanshu_4(unsigned char r,unsigned char s) //第4类：“有输出”“有输入”。
{
    unsigned char t;
    t=r+s;
    return t;
}

void main() //主函数
{
    hanshu_1(); //第1类：“无输出”“无输入”的函数调用。这里的形参的void要省略。
    hanshu_2(2,3); //第2类：“无输出”“有输入”的函数调用。
    c=hanshu_3(); //第3类：“有输出”“无输入”的函数调用。这里的形参的void要省略。
    d=hanshu_4(2,3); //第4类：“有输出”“有输入”的函数调用。
    View(a); //把a发送到电脑端的串口助手软件上观察。
    View(b); //把b发送到电脑端的串口助手软件上观察。
    View(c); //把c发送到电脑端的串口助手软件上观察。
    View(d); //把d发送到电脑端的串口助手软件上观察。
    while(1)
    {
    }
}

/*---C语言学习区域的结束。-----*/

```

在电脑串口助手软件上观察到的程序执行现象如下：

开始...

第1个数

十进制:5

十六进制:5

二进制:101

第 2 个数
十进制:5
十六进制:5
二进制:101

第 3 个数
十进制:5
十六进制:5
二进制:101

第 4 个数
十进制:5
十六进制:5
二进制:101

分析:

变量 a 为 5。

变量 b 为 5。

变量 c 为 5。

变量 d 为 5。

【55.9 如何在单片机上练习本章节 C 语言程序？】

直接复制前面章节中第十一节的模板程序，练习代码时只需要更改“C 语言学习区域”的代码就可以了，其它部分的代码不要动。编译后，把程序下载进带串口的 51 学习板，通过电脑端的串口助手软件就可以观察到不同的变量数值，详细方法请看第十一节内容。