

第六十四节： 指针让函数具备了多个相当于 return 的输出口。

【64.1 函数的三类输出渠道。】

函数是模块，模块必须具备输入和输出的接口，从输入和输出的角度分析，函数对外部调用者传递信息主要有三类渠道，第一类是全局变量，第二类是 return 返回值，第三类是用指针。全局变量太隐蔽，没有那么直观，可读性稍差。return 可读性强，缺点是一个函数只能有一个 return，如果一个函数要输出多个结果，return 就力不从心。指针作为函数的输出接口，就能随心所欲了，不但可读性强，而且输出的接口数量不受限制。

【64.2 只有一个输出接口的时候。】

现在举一个例子，要用函数实现一个加法运算，输出“一个”加法运算的和，求 3 加上 5 等于 8。下面三个例子中分别使用“全局变量，return, 指针”这三类输出接口。

第一类：全局变量。

```
unsigned char DiaoYongZhe; //调用者
unsigned char BeiJiaShu;   //被加数
unsigned char JiaShu;      //加数
unsigned char He;          //输出的接口，加法运算的“和”。

void JiaFa(void)
{
    He=BeiJiaShu+JiaShu;
}

void main()
{
    BeiJiaShu=3;           //填入被加数 3
    JiaShu=5;              //填入加数 5
    JiaFa();               //调用一次加法运算的函数
    DiaoYongZhe=He;        //把加法运算的“和”赋值给调用者。
}
```

第二类：return。

```
unsigned char DiaoYongZhe; //调用者

unsigned char JiaFa(unsigned char BeiJiaShu,unsigned char JiaShu)
{
    unsigned char He;
    He=BeiJiaShu+JiaShu;
    return He;
}
```

```
void main()
{
    DiaoyongZhe=JiaFa(3,5);    //把加法运算的“和”赋值给调用者，一气呵成。
}
```

第三类：指针。

```
unsigned char DiaoyongZhe;    //调用者

void JiaFa(unsigned char BeiJiaShu,unsigned char JiaShu,unsigned char *pu8He)
{
    *pu8He=BeiJiaShu+JiaShu;
}

void main()
{
    JiaFa(3,5,&DiaoyongZhe);    //通过指针这个输出渠道，把加法运算的“和”赋值给调用者，一气呵成。
}
```

【64.3 有多个输出接口的时候。】

现在举一个例子，要用函数实现一个除法运算，分别输出除法运算的商和余数这“两个”数，求 5 除以 3 等于 1 余 2。因为 return 只能输出一个结果，所以这里不列举 return 的例子，只使用“全局变量”和“指针”这两类输出接口。

第一类：全局变量。

```
unsigned char DiaoyongZhe_Shang; //调用者的商
unsigned char DiaoyongZhe_Yu;    //调用者的余数

unsigned char BeiChuShu;          //被除数
unsigned char ChuShu;            //除数
unsigned char Shang;             //输出的接口，除法运算的“商”。
unsigned char Yu;                //输出的接口，除法运算的“余”。

void ChuFa(void)
{
    Shang=BeiChuShu/ChuShu;      //求商。假设除数不会为 0 的情况。
    Yu=BeiChuShu%ChuShu;        //求余数。假设除数不会为 0 的情况。
}

void main()
{
    BeiChuShu=5;                //填入被除数 5
    ChuShu=3;                   //填入除数 3
}
```

```

ChuFa();           //调用一次除法运算的函数
DiaoYongZhe_Shang=Shang; //把除法运算的“商”赋值给调用者的商。
DiaoYongZhe_Yu=Yu;   //把除法运算的“余数”赋值给调用者的余数。
}

```

第二类：return。

return 只能输出一个结果，力不从心，所以这里不列举 return 的例子。

第三类：指针。

```

unsigned char DiaoYongZhe_Shang; //调用者的商
unsigned char DiaoYongZhe_Yu;    //调用者的余数

void ChuFa(unsigned char BeiChuShu,
            unsigned char ChuShu,
            unsigned char *pu8Shang,
            unsigned char *pu8Yu)
{
    *pu8Shang=BeiChuShu/ChuShu; //求商。假设除数不会为 0 的情况。
    *pu8Yu=BeiChuShu%ChuShu;   //求余数。假设除数不会为 0 的情况。
}

void main()
{
    ChuFa(5, 3, &DiaoYongZhe_Shang, &DiaoYongZhe_Yu); //通过两个指针的输出接口，一气呵成。
}

```

【64.4 例程练习和分析。】

现在编一个练习程序。

```

/*---C 语言学习区域的开始。-----*/
void ChuFa(unsigned char BeiChuShu,
            unsigned char ChuShu,
            unsigned char *pu8Shang,
            unsigned char *pu8Yu); //函数声明

unsigned char DiaoYongZhe_Shang; //调用者的商
unsigned char DiaoYongZhe_Yu;    //调用者的余数

void ChuFa(unsigned char BeiChuShu,
            unsigned char ChuShu,
            unsigned char *pu8Shang,
            unsigned char *pu8Yu) //函数定义
{

```

```

        *pu8Shang=BeiChuShu/ChuShu;    //求商。假设除数不会为 0 的情况。
        *pu8Yu=BeiChuShu%ChuShu;      //求余数。假设除数不会为 0 的情况。
    }

void main() //主函数
{
    ChuFa(5, 3, &DiaoYongZhe_Shang, &DiaoYongZhe_Yu); //函数调用。通过两个指针的输出接口，一气呵成。

    View(DiaoYongZhe_Shang); //把第 1 个数 DiaoYongZhe_Shang 发送到电脑端的串口助手软件上观察。
    View(DiaoYongZhe_Yu);    //把第 2 个数 DiaoYongZhe_Yu 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:1

十六进制:1

二进制:1

第 2 个数

十进制:2

十六进制:2

二进制:10

分析：

DiaoYongZhe_Shang 为 1。

DiaoYongZhe_Yu 为 2。

【64.5 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。