

第四十五节： 组合判断 if...else if...else。

【45.1 三类组合判断语句的概述。】

if 的英文含义是“如果”，else 是“否则”，else if 是“否则如果”。在 C 语言里，if, else if, else 所表达的含义，跟英文也是一样的。

if, else if, else 可以组成三种组合判断语句，第一种是“二选一”，第二种是“多选一”，第三种是“多选一或者什么都不选”。

这类组合判断语句可以这样解读：在众多条件判断中，先从第一个 if 条件开始判断，如果第一个 if 条件是真，那么不管后面的条件是否为真，都不再判断，直接执行第一个 if 条件后面大括号的语句，组合语句中其它剩余的条件不再判断直接跳过，否则，就挨个条件往下判断，只要其中一个条件满足，就不再判断剩余的条件，也就是我们日常所说的多选一，甚至在某些组合语句如果所有条件都不满足，那么什么也不选。总之，在如此众多的条件中，最多只能执行一个条件后面大括号的语句。组合语句还有一个规律：if 语句只能出现在第一个条件判断，而且只能出现一次；else 只能出现在最后，而且也只能出现一次；而 else if 语句总是出现在中间，绝对不能出现在第一个条件判断，如果没有 else，也可以出现在最后的条件判断。多说一句，在上述所提到的“只能出现一次”的概念仅仅局限于在一个组合判断语句的范围内，而组合判断语句在整个程序的出现次数是不受限制的。

【45.2 二选一的组合判断。】

先讲第一种“二选一”的书写格式，如下：

书写格式如下：

```
if(条件 1)    //if 只能出现第一个条件，并且只能出现一次
{
    语句 1;
}
else          //else 只能出现最后，并且也只能出现一次。
{
    语句 2;
}
语句 3;
```

这类语句的书写特点是：第一个是 if 判断语句，最后一个是 else 语句，中间没有 else if 判断语句。

这类语句的执行顺序是：先判断第一个的 if 里面的（条件 1），如果（条件 1）满足而为真，就执行该（条件 1）后面紧跟的大括号里面的“语句 1”，执行完该大括号内的所有语句之后，就直接跳出整个组合判断的语句，不再判断也不再执行剩下的 else 那部分的代码，直接跳到“语句 3”处，从“语句 3”处（包括“语句 3”）继续往下执行。但是，如果第一个的 if 里面的（条件 1）不满足而为假，那么就执行 else 后面大括号内的语句。也就是说，else 是在 if 条件不满足时才执行的，所以叫“二选一”，在 if 和 else 之间二选一。

【45.3 多选一的组合判断。】

接着讲第二种书写格式的“多选一”，这种书写格式，跟第一种对比，是在 if 与 else 的中间多插入了 N 个 else if 的判断语句。书写格式如下：

```
if(条件 1)          //if 只能出现第一个条件，并且只能出现一次
```

```

{
    语句 1;
}
else if(条件 2)    //else if 只能出现中间或最后，可以出现多次
{
    语句 2;
}
...
else if(条件 N)    //else if 只能出现中间或最后，可以出现多次
{
    语句 N;
}
else                //else 只能出现最后，并且也只能出现一次。
{
    语句 N+1;
}
语句 N+2;

```

这类语句的书写特点是：第一行是 if 开始，最后一行以 else 结束，中间是 N 个 else if 判断语句。

这类语句的执行顺序是：跟第一种“二选一”对比，判断顺序和规律大致也是一样的，也是从第一个 if 开始，往下逐个判断，然后到中间的 else if，只要发现一个条件满足，就执行该条件后面的大括号内的代码，之后就马上结束整个组合判断语句，不再判断剩下的组合判断语句。但是，如果万一前面第一个 if 和中间所有的 else if 的条件都不满足而为假，就直接执行最后一个 else 大括号内的语句。所以叫“多选一”，在“第一个 if、中间的 else if、最后一个 else”之间多选一。

【45.4 多选一或者什么都不选的组合判断。】

最后讲第三种书写格式的“多选一或者什么都不选”，这种书写格式，跟第二种对比，只有第一个 if 和其它的 else if 语句，没有最后那个 else 语句。书写格式如下：

```

if(条件 1)                //if 只能出现第一个条件，并且只能出现一次
{
    语句 1;
}
else if(条件 2)    //else if 只能出现中间或最后，可以出现多次
{
    语句 2;
}
...
else if(条件 N)    //else if 只能出现中间或最后，可以出现多次
{
    语句 N;
}
语句 N+1;

```

这类语句的书写特点是：第一行是 if 开始，中间是 N 个 else if 判断语句，没有最后一个 else 语句。

这类语句的执行顺序是：跟第二种“多选一”对比，判断顺序和规律大致也是一样的，也是从第一个 if

开始，往下逐个判断，然后到中间的 else if，只要发现一个条件满足，就执行该条件后面的大括号内的代码，之后就马上结束整个组合判断语句，不再判断剩余的组合判断语句。但是，如果万一前面第一个 if 和中间所有的 else if 的条件都不满足而为假，因为此时没有 else 语句，就意味着整个组合判断语句都没有条件满足，因此就没有相关满足的代码被执行到。所以把这种情况称为“多选一或者什么都不选”。

【45.5 例程练习和分析。】

现在编写一个实验程序。

程序代码如下：

```
/*---C 语言学习区域的开始。-----*/

//x 这个变量作为条件判断的变量
unsigned char x=5;

//a,b,c 这 3 个变量作为输出判断结果,0 代表什么语句都没执行，1 代表执行了语句 1，
//2 代表执行语句 2，3 代表执行语句 3。
unsigned char a=0;
unsigned char b=0;
unsigned char c=0;

void main() //主函数
{

    //第一种“二选一”
    if(x>6)
    {
        a=1;    //1 代表执行了“语句 1”
    }
    else
    {
        a=2;    //2 代表执行了“语句 2”
    }

    //第二种“多选一”
    if(x>6)
    {
        b=1;    //1 代表执行了“语句 1”
    }
    else if(7==x)
    {
        b=2;    //2 代表执行了“语句 2”
    }
    else
```

```

    {
        b=3;  //3 代表执行了 “语句 3”
    }

    //第三种 “多选一或者什么都不选”
    if(x>6)
    {
        c=1;  //1 代表执行了 “语句 1”
    }
    else if(7==x)
    {
        c=2;  //2 代表执行了 “语句 2”
    }
    else if(8==x)
    {
        c=3;  //3 代表执行了 “语句 3”
    }

    View(a);  //把第 1 个数 a 发送到电脑端的串口助手软件上观察。
    View(b);  //把第 2 个数 b 发送到电脑端的串口助手软件上观察。
    View(c);  //把第 3 个数 c 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:2

十六进制:2

二进制:10

第 2 个数

十进制:3

十六进制:3

二进制:11

第 3 个数

十进制:0

十六进制:0

二进制:0

分析:

变量 a 为 2。(2 代表执行了语句 2)

变量 b 为 3。(3 代表执行了语句 3)

变量 c 为 0。(0 代表什么语句都没执行)

【45.6 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。