|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 时间：10月12日 | |
| C语言程序设计学习报告 | | | | |
| 图像 | | | | |
| 学生姓名 卢济渊 | |  |  |  |

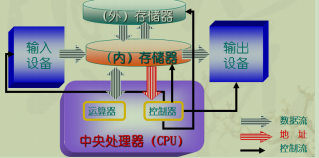
知识点总结



一，计算机原理

(1)工作原理

计算机以二进制表示数据和指令，并将程序存入存储器中，最后由控制器读取并执行。其工作原理如下图



**（2）编程机制**

**C语言编程的基本策略是用程序把源代码转换为可执行文件。典型的C语言通过编译和链接实现这一过程。编译器把源代码转换成中间代码，链接器把中间代码和其他代码合并，生成可执行文件。**

**（3）数值在计算机中的表示**

**原码：正数符号为1，负数符号为0，其他位按一般方法表示其绝对值。**

**[+1]原= 0000 0001**

**[-1]原= 1000 0001**

**第一位是符号位，因为第一位是符号位，所以8位二进制数的取值范围就是：（即第一位不表示值，只表示正负。）**

**[1111 1111 , 0111 1111]**

**即**

**[-127 , 127]**

**反码：正数的反码与原码相同，负数的反码是符号位不变，其他位取反。**

**补码：正数的补码与原码相同，负数的补码是反码的最末位加一。**

1. **数据的表现形式**

**所谓形式就是对数据分配存储单元的安排，数据的表现形式多种多样，总的来说已学过的有以下几种**

**（1）整型变量**

**1.1 基本整型：整数常量是不带小数的数值，用来表示正负数。C语言的整数常量有三种形式：十进制、八进制、十六迸制。十进制整数是由不以0开头的0～9的数字组成的数据。八进制整数是由以0开头的0～7的数字组成的数据。十六进制整数是由以0x或0x开头的0～9的数字及A～F的字母(大小写字母均可)组成的数据。**

**重点基本整型通常用int表示**

**1.2 Short类型；short类型说明一个带符号的16位整型变量。**

**重点Short类型通常用short int表示**

**1.3 Long类型：long类型说明一个带符号的64位整型变量。对于大型计算，常常会遇到很大的整数，并超出int所表示的范围，这时要使用long类型。**

**重点Long类型通常用long int表示**

**（2）浮点类型**

**浮点型数据是用来表示具有小数点的实数的。浮点数包括单精度浮点数和双精度浮点数。**

**2.1 Float型：系统为每一个float型变量分配四个字节，数值以常规的二进制指数形式存放在存储单元中。在存储时，系统将整型变量分成小数部分和指数部分两个部分，分别存放。**

**重点Float型变量常用float定义数据类型**

**2.2 Double型；为了能扩大数据的表现范围，用8个字节存储double型数据，可以得到15位有效数字。为了提高精度，在c语言计算时，会将float型数据转换为double型数据进行运算。**

**重点Double型变量常用double定义数据类型**

**三．数据的运算**

**几乎每一个程序都需要进行运算，对数据进行加工处理，否则程序就没有意义了。要进行运算，就需要规定可以使用的运算。C语言的运算符范围很宽，把除了控制语句和输入输出语句以外基本操作都为运算符处理。**

（1） **算数运算**

**1.1常见的算数运算符**

**常见运算符如下图所示；**

****

**特殊的**

**自增N++，自减N--，运算符的作用是使变量加一或减一，例如；**

**++i，--i（在使用i之前先使i的值加（减））**

**i++，i-- (在使用i之后先使i的值加（减））**

**1.2 算数表达式和运算符的优先级**

**用算数运算式将运算对象链接起来的，符合C语法规则的式子称为C算数表达式。运算的对象包括常量，变量，函数。C语言中规定了运算符的优先级，还规定了运算符的结合性。在运算中通常按照数学中四则运算的原则，有括号先算括号，需要注意的是=在C语言中也算是一种运算，称为赋值运算，即将某个数字的值赋给另一个变量。**

**特殊的**

**上述的加减乘除和=运算都涉及两个变量，这种运算称为双目运算，而类似于+1，-1之类的运算被陈为单目运算。一般来说，单目运算的优先级都会高于双目运算。**

**（2）逻辑运算**

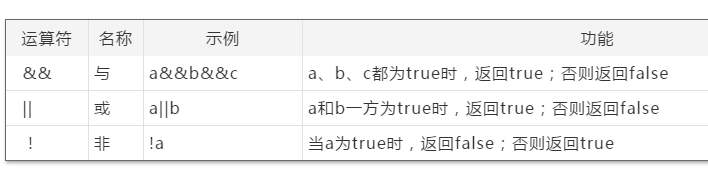
**2.1 简介**

**逻辑运算只有两种结果，真为1，假为0。在C语言中用1代表所有非零的数。**

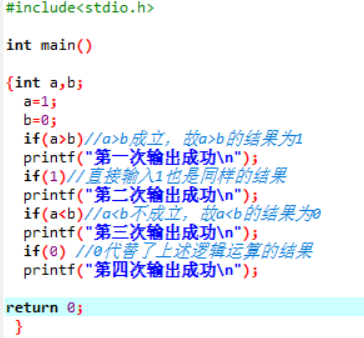
**例;if（2） {printf(“a”}**

**上述语句因为if（）内的值是非零的所以该语句运行时会输出a 。**

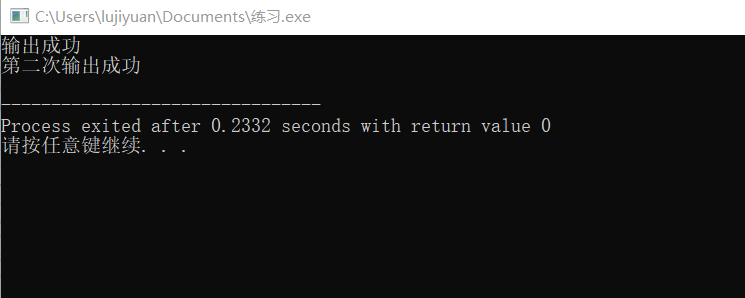
**2.2 逻辑运算符**

**逻辑运算符分为三种，如下图所示。**

**关于逻辑运算的例子具体可以参照下例；**

****

**上述程序的输出结果是**

****

**四．输入与输出**

**（1）输入**

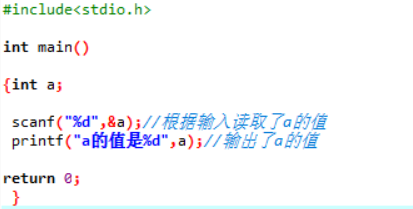
**Scanf是最常见的输入输出格式。其使用格式为scanf（“%d”,&a）。**

**其中该语句由三部分组成:**

* **格式说明符：前缀为%,用于告诉方法下次要读入何种数据类型的数据,并顺次放到方法后的变量中.**
* **空白符:：由空格(" ")、制表符("\t")和新行符("\n")表示,让方法在输入流中忽略一个或多个空白符（只要存在一个就可以忽略多个）。控制串中的空白符使 scanf() 在输入流中读，但不保存结果，直到发现非空白字符为止。**
* **非空白符：除去格式说明符和空白符以外的其他字符，如逗号，分号，于空白符相同，scanf()在输入流中读，但不保存结果。**

1. **输出**

**Printf是最为常见的输出语句，printf()方法和scanf()基本对应，不过在输入变量的时候，不需要指定地址（也就是不需要添加&）。该语句其基本格式是printd(“%d”,a)。“”内的字符即是需要输出的量。**

**有关输入输出可见下例；**

**上述程序运行后如下**

****

**五，选择与循环**

**(1)选择**

* 1. **if语句**

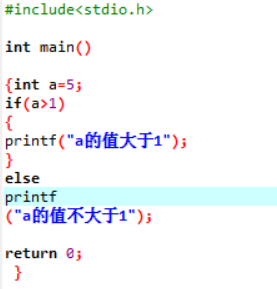
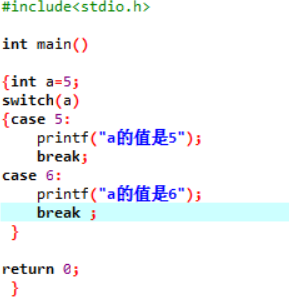
**用if语句可以实现选择结构。if后面的小括号中需要写入一个条件表达式，if语句会根据这个条件表达式结果的真假，让程序产生不同的结果，也就是程序分支。具体来说，就是当条件表达式的值是真时，程序执行if下面大括号中的语句，而当条件表达式的值是假时，程序执行else下面大括号中的语句。也就是说，if和else后面的程序区块中只有一个会被执行，要么是if后面大括号中的语句，要么是else后面大括号中的语句。所以程序在执行到if语句时就产生了逻辑分支。当多个if语句嵌套时便可以实现多分支的选择。**

**重点 if语句的格式 if（条件）{执行语句}**

**1.2 switch语句**

**switch语句可以算是if else语句的一种特殊情况。当我们使用if else语句来做判断时，如果我们只想对条件表达式的结果做一些特定值的判断，那么就可以使用switch语句。switch右侧的小括号中的变量或表达的值的结果是一个确定的值。大括号中可以有多个case语句，case右侧的值如果等于switch右侧小括号中的值，那么程序将执行当前case大括号下的内容（case下的大括号可以省略，但建议保留），在大括号中的最后一行需要加入break语句，表示执行完当前条件分支的case之后程序跳出当前switch结构，如果没有break则程序会顺序执行下面的case语句代码，即使case后的值不等于switch的条件值。如果switch中所有行default后的大括号中的内容，default分支即为默认条件分支，也就是说当所有的case分支都没有起作用时程序将执行此默认分支。（default分支可以省略）**

**有关if，switch的使用示例如下；**

** **

**重点 case并不具有分隔语句的功能，需要用brake使其退出switch**

**（2）循环**

**2.1 while语句**

**while循环语句是C语言中最基本的循环方式，while是C语言的一个关键字，其后是使用一个小括号中的条件表达式来做为执行循环的条件，也就是说当条件表达式的结果为真时执行大括号里面的的程序内容，而当条件表达式的结果为假时不执行大括号中的内容。其实这与if语句的语法有些类似，当条件表达式为真时if后的执行程序只执行一次，而while则会循环执行下去，直到条件表达式的结果为假时结束。**

**它的语法结构如下：**

**while (条件表达)  
{  
//循环体  
}**

**2.2 do while语句**

**do while语句也同样是用于完成程序循环的一种方式不同的是，do while循环在进入循环前会先做一遍do（）中的语句。换一个角度说，即是无论初始条件是什么do while循环都会先进行一遍循环体内的语句。这也是do while循环与while循环最大的差别。**

**do while循环的基本格式如下**

**do  
{  
//循环体  
}  
while (条件表达式);**

**2.3 for循环**

**除了可以用while和do while语句实现循环以外，C语言还提供了for语句实现循环，而且for语句更加灵活，不仅可以用于循环次数已经确定的情况，还可以用于循环不确定只给出循环结束条件都情况。**

**For的格式如下**

**For（初始条件，进入循环的条件，每次循环结束后的动作）**

**重点 continue可以提前结束本次循环进入下一轮循环**

**学习心得总结;**

**计算机导论与程序设计是一门讲究实践与理论共同结合的学科。在平时学习的过程中，不仅需要扎实基础，理解各种语句的意义和用法更要狠抓实践，在实践中找到自己只是的漏洞。理论为实践打下基础，实践又可以补充理论知识的漏洞，两者相辅相成，缺一不可。**