**C语言知识复习资料**

**第一章 C语言基本知识**

**【考点1】C程序**

用C语言编写的程序称为C语言源程序，源程序文件的后缀名为“.c”。源程序经编译后生成后缀名为“.obj”的目标文件，再把目标文件与各种库函数连接起来，生成“.exe”可执行文件。C语言有三种基本结构：顺序结构、选择结构、循环结构。

**【考点2】main函数**

又称主函数，是C程序的入口。main后面跟一对小括号和一对花括号，花括号括起来的部分称为main函数的函数体。一个C程序从main函数开始执行，到main函数体执行完结束，而不论main函数在整个程序中的位置如何。每一个程序有且仅有一个main函数，其他函数都是为main函数服务的。

**【考点3】存储形式**

计算机在电脑中保存数据是采用二进制形式，由0或1构成的二进制称为位（bit），八个位构成一个字节（Byte），1个Byte=8个bit。二进制、八进制、十六进制转化为十进制采用乘法，十进制转化为二进制、八进制、十六进制采用除法。数据的存放位置就是它的地址。

**【考点4】注释**

是对程序的说明，可出现在程序中任意合适的地方，注释从“/\*”开始到最近一个“\*/”结束，其间任何内容都不会被计算机执行，注释不可以嵌套。

**【考点5】书写格式**

每条语句的后面必须有一个分号，分号是语句的一部分。一行内可写多条语句，一个语句可写在多行上。

**【考点6】标识符**

是标识名字的有效字符序列，可以理解为C程序中的单词。

标识符的命名规则是：

（1）标识符只能由字母、数字和下划线组成，字母区分大小写。

（2）标识符的第一个字符必须是字母或下划线，不能为数字。

C语言标识符分如下3类

（1）关键字。它们在程序中有固定的含义，不能另作他用。如int、for、switch等。

（2）预定义标识符。预先定义并具有特定含义的标识符。如define、include等。

（3）用户标识符。用户根据需要定义的标识符，符合命名规则且不与关键字相同。

**【考点7】常量与变量**

常量是指在程序运行过程中，其值不能改变的量。常量分为整型常量、实型常量、字符常量、字符串常量、符号常量5种。在程序运行过程中其值可以改变的量称为变量。C语言中没有字符串变量。存放字符串使用字符数组。

**【考点8】整型数据**

整型常量有十进制、八进制、十六进制三种表示形式，没有二进制形式。八进制整型常量加前导数字0，十六进制常量加前导0X，八进制常量中不会出现8。

整型变量可分为基本整型（int）、短整型（short）、长整型（long）、和无符号整型（unsigned）。一个基本整型占4个字节。

**【考点9】实型数据**

实型数据有两种表示形式：小数形式和指数形式。掌握判定指数形式合法性。

实型变量分为单精度型（float）和双精度型（double），单精度型占四个字节。

**【考点10】算术运算**

算术运算符一共有+、—、\*、/、%这五个。求余运算要求运算对象只能为整型，除法运算符两边运算对象都为整型时，运算结果也为整型即舍掉小数部分。

**【考点11】强制类型转换**

将一个运算对象转换成指定类型，格式为（类型名）表达式，注意小括号位置。

**【考点12】赋值**

赋值运算符为“=”，不同于关系等于“= =”。赋值表达式格式为：变量名=表达式，赋值运算符左边必须为变量，赋值运算是把赋值运算符右边表达式的值赋给左边变量。

复合赋值运算符是将算术运算符或位运算符与赋值运算符组合在一起组成的运算符，掌握复合赋值表达式转化为赋值表达式的方法。如n+=100可转化为n=n+100。

**【考点13】自加自减运算**

自加运算符“++”与自减运算符“--”是单目运算符，运算对象必须是变量。自增自减运算分前缀运算和后缀运算，它们所对应的表达式的值是有区别的，如j=i++;等价于j=i;i=i+1;而j=++i;等价于i=i+1;j=i;。

**【考点14】逗号运算**

逗号运算符运算优先级最低，可将多个表达式构成一个新的表达式。

**第二章 顺序结构**

**【考点1】运算符、表达式、语句**

运算对象加运算符构成表达式，表达式加分号构成表达式语句，运算对象可以是表达式、常量、变量。如赋值运算符加运算对象构成赋值表达式，赋值表达式加分号又可构成赋值语句。

**【考点2】运算符的优先级和结合顺序**

运算符按参加运算的对象数目可分为单目运算符、双目运算符和三目运算符。初等运算符的优先级别最高，然后依次是单目运算符、算术运算符、关系运算符、逻辑运算符（除逻辑非！）、条件运算符、赋值运算符、逗号运算符。位运算符优先级介于算术运算符与逻辑运算符之间。结合顺序大多为自左向右，而自右向左的有三个：单目运算符、条件运算符和赋值运算符。

**【考点3】printf函数**

格式为：printf(输出控制，输出列表)。输出控制是用一对双引号括起来的，包含格式说明和原样信息。输出列表包含若干输出项。

**【考点4】printf函数中格式说明**

%d对应整型，%f对应单精度实型，%c对应字符型，%o对应八进制无符号整型，%x对应无符号十六进制整型，%u对应无符号整型，%e对应指数型，%s对应字符串型。可在%和格式字符之间加一个数来控制数据所占的宽度和小数位数。

**【考点5】scanf函数**

输入项要求带取地址符&。当用键盘输入多个数据时，数据之间用分隔符。分隔符包括空格符、制表符和回车符，但不包括逗号。

**【考点】6如何交换两个变量**

要使用中间变量，语句为：t=x; x=y; x=t;。

**第三章 选择结构**

**【考点1】关系运算**

C语言用非0表示逻辑真，用0表示逻辑假。关系运算符有6个，分别是>，>=，<，<=，==，!=，前四种优先级高于后两种。关系表达式真时为1，假时为0。注意a<b<c是不可以的，可用(a<b)&&(b<c)来表示。

**【考点2】逻辑运算**

逻辑运算符共有3个：逻辑与（&&），逻辑或（||），逻辑非（!）。注意短路现象，例a++||b++，如果表达式a++的值非零，则表达式b++不再执行。

**【考点3】if语句**

可以单独出现，也可以与else匹配出现。if语句可以嵌套，这时else总是与离它最近的且没有与else匹配的if匹配。

**【考点4】条件运算**

是唯一的三目运算符，格式为：表达式1?表达式2:表达式3。表达式1值为非0时，整个表达式值为表达式2的值，表达式1值为0时，整个表达式值为表达式3的值。

**【考点5】switch语句**

要注意每条case后有没有break语句的区别。还要注意switch后小括号里面的表达式不能为实型，case后表达式不能有变量。

**第四章 循环结构**

**【考点1】三种循环结构**

三种循环结构分别为：while，do-while，for。注意for循环中的小括号中必须是两个分号；循环一定要有结束条件，否则成了死循环；do-while()循环最后的while()；后一定要有分号。

**【考点2】break与continue**

break是终止所在整个循环，而continue是提前结束本轮循环。break语句可出现在循环结构与switch语句中，continue只出现在循环结构中。

**【考点3】循环的嵌套**

就是循环里面还有循环，计算要一层一层分析，一般只考查两层嵌套，循环嵌套通常是处理二维数组。

**【考点4】循环结构的复习**

循环结构是重点，应用性很强。要求学员重点理解并多加练习，领会掌握。

**第五章 字符型数据 位运算**

**【考点1】字符常量**

一个字符常量用一对单引号括起来，字符常量只能包括一个字符，'ab'是非法的。空格常用'□'来表示（此处用□来代表一个空格，不能写成''两个连续的单引号）。字符常量可用对应的ASCII码表示，需记住：'0'的ASCII码为48，'A'的ASCII码为65，'a'的ASCII码为97。

**【考点2】转义字符**

一对单引号中以一个反斜线后跟一个特定字符或八进制、十六进制数来构成转义字符。比如'\n'表示换行，'\101'或'\x41'表示ASCII码为65的字符'A'。

**【考点3】字符型数据可以和整型数据相互转换**

如：'0'-0=48 'A'+32='a' char a=65;printf("%d%c",a,a);结果为65A

**【考点4】位运算符**

C语言提供6种位运算符：按位求反~，按位左移<<，按位右移>>，按位与&，按位异或|，按位或^。一般情况下需要先转化进制。异或运算的规则：0异或1得到1，0异或0得到0，1异或1得到0。可记为“相同为0，不同为1”。

**【考点5】putchar与getchar函数**

可用于输出或输入单个字符，这两个函数是stdio.h文件中的库函数，它们是printf与scanf函数的简化。

**第六章 函数**

**【考点1】函数的定义**

函数是具有一定功能的一个程序块。函数的首部为：函数类型 函数名（类型1 形参1，类型2 形参2，……）。在函数定义中不可以再定义函数，即不能嵌套定义函数。函数类型默认为int型。

**【考点2】库函数**

调用C语言标准库函数时要包含include命令，include命令行以#开头，后面是""或<>括起来的后缀为”.h”的头文件。以#开头的一行称为编译预处理命令行，编译预处理不是C语言语句，不加分号，不占运行时间。

**【考点3】函数的返回值**

函数通过return语句返回一个值，返回的值类型与函数类型一样。return语句只执行一次，执行完或函数体结束后退出函数。

**【考点4】函数的声明**

函数要“先定义后调用”，或“先声明再调用后定义”。函数的声明一定要有函数名、函数返回值类型、函数参数类型，但不一定要有形参的名称。

**【考点5】函数的调用**

程序从上往下执行，当碰到函数名后，把值传给调用函数，当程序得到了返回值或调用函数结束，再顺序往下执行。

**【考点6】函数的参数及值传递**

形式参数简称形参，是定义函数时函数名后面括号中的参数。实在参数简称实参，是调用函数时函数名后面括号中的参数。实参和形参分别占据不同的存储单元。实参向形参单向传递数值。

“传值”与“传址”的区别：传数值的话，形参的变化不会改变实参的变化。传地址的话，形参的变化就有可能改变实参所对应的量。

**【考点7】函数的递归调用**

函数直接或间接地调用自己称为函数的递归调用。递归调用必须有一个明确的结束递归的条件。在做递归题时可把递归的步骤一步步写下来，不要弄颠倒了。

**【考点8】要求掌握的库函数**

sqrt()算术平方根函数，fabs()绝对值函数，pow()幂函数，sin()正弦函数

**第七章 指针**

**【考点1】指针变量**

指针变量是用来存储地址的，而一般变量是存储数值的。指针变量可指向任意一种数据类型。

**【考点2】指针变量的定义**

格式为：类型名 \*指针变量名。二维指针int \*\*p;可以理解为基类型为(int \*)类型。

**【考点3】指针变量的初始化**

指针变量在使用前必须要初始化，把一个具体的地址赋给它，否则引用时会有副作用，如果不指向任何数据就赋“空值”NULL。

**【考点4】指针变量的引用**

&是取地址符，\*是间接访问运算符，它们是互逆的两个运算符。在指针变量名前加间接访问运算符就等价它所指向的量。

**【考点5】指针的运算**

\*p++和(\*p)++之间的差别：\*p++是地址变化，(\*p)++是指针变量所指的数据变化。一个指针变量加一个整数不是简单的数学相加，而是连续移动若干地址。当两个指针指向同一数组时，它们可以比较大小进行减法运算。

**第八章 数组**

**【考点1】数组的定义**

数组是一组具有相同类型的数据的集合，这些数据称为数组元素。格式为：类型名 数组名[常量表达式]。数组的所占字节数为元素个数与基类型所占字节数的乘积。

**【考点2】数组的初始化**

第一维长度可以不写，其它维必须写。int a[]={1,2};合法，int a[][3]={2,3,4};合法，int a[2][]={2,3,4};非法。数组初始化元素值默认为0，没有初始化元素值为随机。如在int a[5]={0,1,2};中，元素a[4]值为0；而在int a[5];中，元素a[4]值为一个不确定的随机数。

**【考点3】元素的引用**

数组元素的下标从0开始，到数组长度减1结束。所以int a[5];中数组最后一个元素是a[4]。要把数组元素看作一个整体，可以把a[4]当作一个整型变量。

**【考点4】二维数组**

数组a[2][3]={1,2,3,4,5,6};中含6个元素，有2行3列。第一行为a[0]行，第2行为a[1]行，a[0]、a[1]叫行首地址，是地址常量。\*(a[0]+1)是第一行第一个元素往后跳一列，即元素a[0][1]值为2，\*(a[0]+3)是第一行第一个元素往后跳三个，即元素a[1][0]值为4。

**【考点5】行指针**

是一个指针变量，占四个字节，行指针指向一行连续数据，形式为：int (\*p)[2];，p只能存放含有两个整型元素的一维数组的首地址。注意(\*p)两边的小括号不能省略，否则就成了指针数组，是若干指针元素的集合。

**【考点6】数组名**

数组名是数组的首地址。数组名不能单独引用，不能通过一个数组名代表全部元素。数组名是地址常量，不能对数组名赋值，所以a++是错误的。但数组名可以作为地址与一个整数相加得到一个新地址。

**【考点7】元素形式的转换**

助记： a[2]变成\*(a+2)，a[2][3]变成\*(a+2)[3]再可变成\*(\*(a+2)+3)。

**第九章 字符串**

**【考点1】字符串常量及表示**

字符串常量是由双引号括起来的一串字符，如"ABC"。在存储字符串时，系统会自动在其尾部加上一个空值'\0'，空值也要占用一个字节，也就是字符串"ABC"需要占四个字节。

**【考点2】字符数组**

C语言没有字符串变量，只能采用字符数组来存储字符串。数组的大小应该比它将要实际存放的最长字符串多一个元素，从而存放'\0'。

**【考点3】字符串赋值**

可以用下面的形式进行赋值：char str[]="Hello! ";或char \*p;p="Hello! ";，但不能用下面的形式：char str[10];str="Hello";因为str是一个地址常量，不能进行赋值操作。

**【考点4】字符串的输入与输出**

可以用scanf和printf函数，如scanf("%s",str);，也可用专门处理字符串的两个函数gets和puts函数，还可以对字符数组逐个元素进行赋值，但一定要在最后赋一个'\0'。使用gets函数可以接收空格，使用puts函数在最后输出一个换行。

**【考点5】字符串函数**

要掌握的四个字符串函数：字符串拷贝函数strcpy（），求字符串长度函数strlen（），字符串链接函数strcat（），字符串比较函数strcmp（）。使用这些函数需在预处理部分包含头文件”string.h”。

字符串长度要小于字符数组的长度，例：char str[10]= "Hello";sizeof(str)的值为10（数组长度），strlen(str)的值为5（字符串长度）。这些函数是考试常用到的函数，大家一定要熟练应用这几个函数。

**第十章 结构体与共用体**

**【考点1】结构体类型的说明**

结构体是若干个类型数据的集合，结构体类型说明格式如下：struct 类型名 {类型1 成员名1;类型2 成员名2;……};，以上整个部分是一个数据类型，与整型的int是同样地位。可用typedef把结构体类型替换成一个只有几个字母的简短标识符。

**【考点2】结构体变量的定义**

结构体变量是用说明的结构体类型所定义的一个变量，与结构体类型不是一回事。一个结构体变量所占字节数为其所有成员所占字节数之和。如struct stu{char name[10];int age;} a,b;则表明定义了两个结构体变量a,b,每个变量占14个字节。a,b与int i,j;中的变量i,j是同样地位。

**【考点3】结构体成员的引用**

引用成员可用以下3种方式：（1）结构体变量名.成员名；（2）指针变量名->成员名：（3）（\*指针变量名）.成员名。点（.）称为成员运算符，箭头（->）称为结构指向运算符。

**【考点4】链表**

链表是由一个个结点构成的，一个结点就是一个结构体变量。每个结点可以分为数据域与指针域两个部分，数据域用来存放要存储的数据，指针域用来指向下一个结点。链表是考试中的难点，在C语言和公共基础部分都会考到，要领悟掌握。

**【考点5】共用体**

共用体的使用格式与结构体相似，共用体定义的关键字为union，共用体所占字节数是所有成员中字节数最大的那个。

**第十一章 文件**

**【考点1】文件类型指针**

文件指针是一个指向结构体类型的指针，定义格式为：FILE \*指针变量名。在使用文件时，都需要先定义文件指针。

**【考点2】文本文件与二进制文件**

文本形式存放的是字符的ASCII码，二进制形式存放的是数据的二进制。例如“100”如果是文本形式就是存储’1’、 ’0’、 ’0’三个字符的ASCII码（00110001 00110000 00110000），如果是二进制形式就把100转化成二进制（01100100）。

**【考点3】打开文件**

文件的打开形式如下：FILE \*fp; fp=fopen("c:\\lab.c","rb");。fopen函数的前面一部分为文件名，后面一部分为文件的使用方式。其中r代表读，w代表写，a代表添加，b代表二进制位的。

**【考点4】文件函数**

判断文件结束feof函数，移动文件指针位置fseek函数，获得文件位置ftell函数，文件位置移到开头rewind函数，文件字符输入输出fgetc函数和fputc函数，文件输入输出fscanf函数和fprintf函数，文件字符串输入输出fgets函数和fputs函数，读写二进制文件fread函数和fwrite函数。

以上函数要求知道格式会用，清楚是用于二进制文件还是文本文件，要把教材文件这章仔细复习下，不要在考试的时候把这些文件函数搞混了。

**第十二章 深入讨论**

**【考点1】编译预处理**

凡以#开头的这一行，都是编译预处理命令行，编译预处理不加分号，不占运行时间。宏替换仅是简单的文本替换，如#define f(x) (x)\*(x)和#define f(x) x\*x替换f(2+2)时就有区别，前者展开为(2+2)\*(2+2)，后者为2+2\*2+2。

如果源文件f2.c中有#include"f1.c"可以理解为把源文件f1.c原样包含到f2.c中，使f1.c和f2.c融合到一起成为一个C程序编译。所以一个C程序必有主函数，但一个C源文件未必有主函数。

**【考点2】标识符作用域**

局部变量是在函数内或复合语句内定义的变量，作用域为定义它的函数内。局部变量有三种类型：自动auto，寄存器register和静态static。

自动变量随着函数的使用与否创建消失；寄存器变量分配在cpu中，没有内存地址；静态变量占用固定存储单元，在程序执行过程不释放，直到程序运行结束。

全局变量是在函数外定义的变量，作用域从定义它的位置到整个源文件结束为止，生存期为整个程序运行期间。全局变量都是静态变量。

**【考点3】动态存储分配**

malloc(size)用来创建连续size个字节存储区，返回值类型为void \*型。malloc函数常用于动态创建链表结点，如int \*p; p=(int \*)malloc(sizeof(int));。

calloc（n,size）创建n个同一类型的存储空间，可以理解为n个malloc。

free(p)释放动态分配的存储单元。