

100条经典C语言笔试题目

上嵌基地：朱中浪

QQ: 1729366371

本课件是服务应届毕业生，如有不懂部分，请QQ联系我，另请各位帮忙收集更多的面试笔试题目

- 题目来源：

- 1、中兴、华为、慧通、英华达、微软亚洲技术中心等中外企业面试题目；
- 2、C语言面试宝典(林锐 《高质量编程第三版》)。

- 说明：

- 1、部分C语言面试题中可能会参杂部分和C++相关的知识，为了保持题目的灵活性故保留，但选题最终还是会以C语言题目为主体；
- 2、以上公司的面试题目已成为国内中小型企业公司出题模板；
- 3、由于本人的能力有限加上时间仓促，本课件肯定存在不足之处，恳请各位同学批评指正。

● 题目类型

基本数据类型的考察(1-4)

数组、指针、内存、结构体常考点(5-28)

宏、头文件、关键字的考察(29-54)

程序阅读、改错(55-75)

编程练习(76-100)

- 1、请填写 bool , float, 指针变量 与“零值”比较的 if 语句。

提示：这里“零值”可以是 0, 0.0 , FALSE 或者“空指针”。例如 int 变量 n 与“零值”比较的 if 语句为：

```
if ( n == 0 )
```

```
if ( n != 0 )
```

以此类推。

- (1) 请写出 bool flag 与“零值”比较的 if 语句:
【标准答案】if (flag) if (!flag)

如下写法均属不良风格，不得分。

```
if (flag == TRUE)
```

```
if (flag == 1 )
```

```
if (flag == FALSE)
```

```
if (flag == 0)
```

- (2) 请写出 float x 与“零值”比较的 if 语句:

【标准答案】 `const float EPSINON = 0.00001;`
`if ((x >= - EPSINON) && (x <= EPSINON))`
不可将浮点变量用“==”或“!=”与数字比较，应该设法转化成“>=”或“<=”此类形式。

如下是错误的写法，不得分。

```
if (x == 0.0)
```

```
if (x != 0.0)
```

- (3) 请写出 char *p 与“零值”比较的 if 语句

【标准答案】 if (p == NULL) if (p != NULL)

如下写法均属不良风格，不得分。

```
if (p == 0)
```

```
if (p != 0)
```

```
if (p)
```

```
if (!)
```

- 2、以下为 Linux下的 32 位 C程序，请计算 sizeof 的值。

```
char str[] = "Hello";
```

```
char *p = str;
```

```
int n = 10;
```

请计算

(1) sizeof (str) =

(2) sizeof (p) =

(3) sizeof (n) =

【标准答案】 (1) 6、 (2) 4、 (3) 4

● (4) void Func (char str[100])
{
 ;
}

请计算 sizeof(str) =

(5) void *p = malloc(100);

请计算sizeof (p) =

【标准答案】 (4) 4、 (5) 4

4、用变量a给出下面的定义

e) 一个有10个指针的数组，该指针是指向一个整型数的；

f) 一个指向有10个整型数数组的指针；

g) 一个指向函数的指针，该函数有一个整型参数并返回一个整型数；

h) 一个有10个指针的数组，该指针指向一个函数，该函数有一个整型参数并返回一个整型数；

【标准答案】 e) `int * a[10];` f) `int (*a)[10]`
g) `int (*a)(int);` h) `int (*a[10])(int)`

- 5、设有以下说明和定义：

```
typedef union {long i; int k[5]; char c;} DATE;  
struct data { int cat; DATE cow; double dog;} too;  
DATE max;
```

则语句 `printf("%d",sizeof(struct data)+sizeof(max));` 的执行结果是：_____

【标准答案】 DATE是一个union, 变量公用空间. 里面最大的变量类型是int[5], 占用20个字节. 所以它的大小是20

data是一个struct, 每个变量分开占用空间. 依次为int4 + DATE20 + double8 = 32.

所以结果是 $20 + 32 = 52$.

当然...在某些16位编辑器下, int可能是2字节,那么结果是 $\text{int}2 + \text{DATE}10 + \text{double}8 = 20$

- 6、请问以下代码有什么问题：

```
int main()
{
    char a;
    char *str=&a;
    strcpy(str,"hello");
    printf(str);
    return 0;
}
```

【标准答案】 没有为str分配内存空间，将会发生异常问题出在将一个字符串复制进一个字符变量指针所地址。虽然可以正确输出结果，但因为越界进行内在读写而导致程序崩溃。

- 7、请问以下代码有什么问题：

```
char* s="AAA";  
printf("%s",s);  
s[0]='B';  
printf("%s",s);  
有什么错？
```

【标准答案】"AAA"是字符串常量。s是指针，指向这个字符串常量，所以声明s的时候就有问题。

```
const char* s="AAA";
```

然后又因为是常量，所以对s[0]的赋值操作是不合法的。

8、 `int (*s[10])(int)` 表示的是啥啊

【标准答案】 `int (*s[10])(int)` 函数指针数组，每个指针指向一个 `int func(int param)` 的函数。

9、c和c++中的struct有什么不同？

【标准答案】 c和c++中struct的主要区别是c中的struct不可以含有成员函数，而c++中的struct可以。c++中struct和class的主要区别在于默认的存取权限不同，struct默认为public，而class默认为private

10、 void getmemory(char *p)

```
{  
    p=(char *) malloc(100);  
    strcpy(p,"hello world");  
}
```

```
int main( )  
{  
    char *str=NULL;  
    getmemory(str);  
    printf("%s/n",str);  
    free(str);  
    return 0;
```

}会出现什么问题？

【标准答案】 程序崩溃，getmemory中的malloc 不能返回动态内存， free（）对str操作很危险。

- 11、 `char szstr[10];`
`strcpy(szstr,"0123456789");`
产生什么结果？为什么？

【标准答案】 长度不一样，出现段错误。

● 12、数组和链表的区别？

【标准答案】数组：数据顺序存储，固定大小；
链表：数据可以随机存储，大小可动态改变

```
13、 void main()  
{  
    char aa[10];  
    printf("%d",strlen(aa));  
}
```

会出现什么问题？打印结果是是是多少？

【标准答案】 sizeof()和初不初始化，没有关系，
strlen()和初始化有关，打印结果值未知。

- 14、给定结构struct A
{
 char t:4;
 char k:4;
 unsigned short i:8;
 unsigned long m;
};问sizeof(A) = ?

【标准答案】 8

- 15、 struct name1{
char str;
short x;
int num;
}; 求sizeof(name1)?

【标准答案】 8

- 16、 struct name2{
 char str;
 int num;
 short x;
};求sizeof(name2)?
【标准答案】 12

● 17、程序哪里有错误

```
wap( int* p1,int* p2 )  
{  
    int *p;  
    *p = *p1;  
    *p1 = *p2;  
    *p2 = *p;  
}
```

【标准答案】p为野指针

- 19、`(void *)ptr` 和 `*(void**)ptr`的结果是否相同？其中ptr为同一个指针。

【标准答案】`(void *)ptr` 和 `*(void**)ptr`值是相同的

- 20、要对绝对地址0x100000赋值，我们可以用
(unsigned int*)0x100000 = 1234;
那么要是想让程序跳转到绝对地址是0x100000去执行，应该怎么做？

【标准答案】*((void (*)())0x100000) ();

首先要将0x100000强制转换成函数指针,即:

(void (*)())0x100000

然后再调用它:

((void ()())0x100000)();

100条经典C语言笔试题目

22、关于内存的思考题（1）你能看出有什么问题？

```
void GetMemory(char *p)
{
    p = (char *)malloc(100);
}

void Test(void)
{
    char *str = NULL;
    GetMemory(str);
    strcpy(str, "hello world");
    printf(str);
}
```

答：程序崩溃。

因为 `GetMemory` 并不能传递动态内存，

`Test` 函数中的 `str` 一直都是 `NULL`。

`strcpy(str, "hello world");` 将使程序崩溃。

● 23、关于内存的思考题（2）你能看出有什么问题？

```
char *GetMemory(void)
{
    char p[] = "hello world";
    return p;
}

void Test(void)
{
    char *str = NULL;
    str = GetMemory();
    printf(str);
}
```

答：可能是乱码。

因为 GetMemory 返回的是指向“栈内存”的指针，该指针的地址不是 NULL，但其原现的内容已经被清除，新内容不可知。

请问运行 Test 函数会有什么样的结果？

● 25、关于内存的思考题（3）你能看出有什么问题？

```
void GetMemory2(char **p, int num)
{
    *p = (char *)malloc(num);
}

void Test(void)
{
    char *str = NULL;
    GetMemory(&str, 100);
    strcpy(str, "hello");
    printf(str);
}
```

答：

（1）能够输出 hello

（2）内存泄漏

请问运行 Test 函数会有什么样的结果？

● 26、关于内存的思考题（4）你能看出有什么问题？

```
void Test(void)
{
    char *str = (char *) malloc(100);
    strcpy(str, "hello");
    free(str);
    if(str != NULL)
    {
        strcpy(str, "world");
        printf(str);
    }
}
```

答：篡改动态内存区的内容，后果难以预料，非常危险。

因为 free(str);之后，str 成为野指针，if(str != NULL)语句不起作用。

请问运行 Test 函数会有什么样的结果？

- 27、关键字volatile有什么含意？并给出三个不同的例子。

【参考答案】一个定义为volatile的变量是说这变量可能会被子意外地改变，这样，编译器就不会去假设这个变量的值了。精确地说就是，优化器在用到这个变量时必须每次都小心地重新读取这个变量的值，而不是使用保存在寄存器里的备份。下面是volatile变量的几个例子：

- 1). 并行设备的硬件寄存器（如：状态寄存器）
- 2). 一个中断服务子程序中会访问到的非自动变量 (Non-automatic variables)
- 3). 多线程应用中被几个任务共享的变量

- 28、嵌入式系统经常具有要求程序员去访问某特定的内存位置的特点。在某工程中，要求设置一绝对地址为0x67a9的整型变量的值为0xaa66。编译器是一个纯粹的ANSI编译器。写代码去完成这一任务。

【参考答案】这一问题测试你是否知道为了访问一绝对地址把一个整型数强制转换（typecast）为一指针是合法的。这一问题的实现方式随着个人风格不同而不同。典型的类似代码如下：

```
int *ptr;  
ptr = (int *)0x67a9;  
*ptr = 0xaa55;
```

- 29、头文件中的 `ifndef/define/endif` 干什么用？
【标准答案】防止该头文件被重复引用。

- 30、`#include <filename.h>` 和 `#include "filename.h"` 有什么区别？

【标准答案】 对于 `#include <filename.h>` ，编译器从标准库路径开始搜索 `filename.h` ；
对于 `#include "filename.h"` ，编译器从用户的工作路径开始搜索 `filename.h` 。

- 31、const 有什么用途？（请至少说明两种）

【标准答案】： （1）可以定义 const 常量

（2）const 可以修饰函数的参数、返回值，甚至函数的定义体。被 const 修饰的东西都受到强制保护，可以预防意外的变动，能提高程序的健壮性。

- 32、static有什么用途？（请至少说明两种）

【标准答案】

- 1.限制变量的作用域（static全局变量）；
- 2.设置变量的存储域（static局部变量）。

● 33、堆栈溢出一般是由什么原因导致的？

【标准答案】 没有回收垃圾资源。

● 34、如何引用一个已经定义过的全局变量？

【标准答案】可以用引用头文件的方式，也可以用extern关键字，如果用引用头文件方式来引用某个在头文件中声明的全局变理，假定你将那个变量写错了，那么在编译期间会报错，如果你用extern方式引用时，假定你犯了同样的错误，那么在编译期间不会报错，而在连接期间报错。

- 35、全局变量可不可以定义在可被多个.C文件包含的头文件中？为什么？

【标准答案】可以，在不同的C文件中以static形式来声明同名全局变量。可以在不同的C文件中声明同名的全局变量，前提是其中只能有一个C文件中对此变量赋初值，此时连接不会出错。

● 36、队列和栈有什么区别？

【标准答案】队列先进先出，栈后进先出。

● 37、Heap与stack的差别。

【标准答案】Heap是堆，stack是栈。

Stack的空间由操作系统自动分配/释放，Heap上的空间手动分配/释放。

Stack空间有限，Heap是很大的自由存储区

C中的malloc函数分配的内存空间即在堆上,C++中对应的是new操作符。

程序在编译期对变量和函数分配内存都在栈上进行,且程序运行过程中函数调用时参数的传递也在栈上进行。

。

- 38、用宏定义写出swap (x, y) , 即交换两数。

【标准答案】

```
#define swap(x, y) (x)=(x)+(y);(y)=(x)-(y);(x)=(x)-(y);
```

- 39、写一个“标准”宏，这个宏输入两个参数并返回较小的一个。

【标准答案】`#define Min(X, Y) ((X)>(Y)?(Y):(X))`//结尾没有;

40、带参宏与带参函数的区别(至少说出5点)?

【标准答案】

	带参宏	带参函数
处理时间	编译时	运行时
参数类型	无	需定义
程序长度	变长	不变
占用存储空间	否	是
运行时间	不占运行时间	调用和返回时占

● 41、关键字volatile有什么含意？

【标准答案】提示编译器对象的值可能在编译器未监测到的情况下改变。

● 42、 int main()

{

int x=3;

printf("%d",x);

return 1;

}

问函数既然不会被其它函数调用，为什么要返回1？

【标准答案】main中，c标准认为0表示成功，非0表示错误。具体的值是某中具体出错信息。

- 43、已知一个数组table，用一个宏定义，求出数据的元素个数。

【标准答案】

```
#define NTBL(table) (sizeof(table)/sizeof(table[0]))
```

- 44、 A.c 和B.c两个c文件中使用了两个相同名字的static变量,编译的时候会不会有问题?这两个static变量会保存到哪里（栈还是堆或者其他的）？

【标准答案】static的全局变量，表明这个变量仅在本模块中有意义，不会影响其他模块。

他们都放在静态数据区，但是编译器对他们的命名是不同的。

如果要使变量在其他模块也有意义的话，需要使用extern关键字。

- 45、static全局变量与普通的全局变量有什么区别？

【标准答案】 static全局变量只初使化一次，防止在其他文件单元中被引用；

- 46、static局部变量和普通局部变量有什么区别

【标准答案】static局部变量只被初始化一次，下一次依据上一次结果值；

- 47、static函数与普通函数有什么区别？

【标准答案】static函数在内存中只有一份，普通函数在每个被调用中维持一份拷贝

● 关于45-47的参考文章

全局变量(外部变量)的说明之前再冠以static 就构成了静态的全局变量。全局变量本身就是静态存储方式，静态全局变量当然也是静态存储方式。这两者在存储方式上并无不同。这两者的区别虽在于非静态全局变量的作用域是整个源程序，当一个源程序由多个源文件组成时，非静态的全局变量在各个源文件中都是有效的。而静态全局变量则限制了其作用域，即只在定义该变量的源文件内有效，在同一源程序的其它源文件中不能使用它。由于静态全局变量的作用域局限于一个源文件内，只能为该源文件内的函数公用，因此可以避免在其它源文件中引起错误。从以上分析可以看出，把局部变量改变为静态变量后是改变了它的存储方式即改变了它的生存期。把全局变量改变为静态变量后是改变了它的作用域，限制了它的使用范围。

static函数与普通函数作用域不同。仅在本文件。只在当前源文件中使用的函数应该说明为内部函数(static)，内部函数应该在当前源文件中说明和定义。对于可在当前源文件以外使用的函数，应该在一个头文件中说明，要使用这些函数的源文件要包含这个头文件。

- 48、程序的局部变量存在于____中，全局变量存在于____中，动态申请数据存在于____中。

【标准答案】程序的局部变量存在于栈(stack)中，全局变量存在于静态数据区中，动态申请数据存在于堆(heap)中。

● 49、什么是预编译，何时需要预编译：

【标准答案】 1、总是使用不经常改动的大型代码体

。

2、程序由多个模块组成，所有模块都使用一组标准的包含文件和相同的编译选项。在这种情况下，可以将所有包含文件预编译为一个预编译头。

- 50、用两个栈实现一个队列的功能？要求给出算法和思路！

【参考答案】 设2个栈为A,B, 一开始均为空.

入队:

将新元素push入栈A;

出队:

(1)判断栈B是否为空;

(2)如果不为空, 则将栈A中所有元素依次pop出并push到栈B;

(3)将栈B的栈顶元素pop出;

- 【标准答案】** c用宏定义， c++用inline

- 52、 1. 用预处理指令#define 声明一个常数，用以表明1年中有多少秒（忽略闰年问题）

【参考答案】 #define SECONDS_PER_YEAR (60 * 60 * 24 * 365)UL

- 53、 Typedef 在C语言中频繁用以声明一个已经存在的数据类型的同义字。也可以用预处理器做类似的事。例如，思考一下下面的例子：

```
#define dPS struct s *
```

```
typedef struct s * tPS;
```

以上两种情况的意图都是要定义dPS 和 tPS 作为一个指向结构s指针。哪种方法更好呢？（如果有的话）为什么？

【参考答案】这是一个非常微妙的问题，任何人答对这个问题（正当的原因）是应当被恭喜的。答案是：typedef更好。思考下面的例子：

```
dPS p1,p2;
```

```
tPS p3,p4;
```

第一个扩展为

```
struct s * p1, p2;
```

上面的代码定义p1为一个指向结构的指，p2为一个实际的结构，这也许不是你想要的。第二个例子正确地定义了p3 和p4 两个指针。

- 54、在 C++ 程序中调用被 C 编译器编译后的函数，为什么要加 extern “C”？

【标准答案】C++语言支持函数重载，C 语言不支持函数重载。函数被 C++编译后在库中的名字与 C 语言的不同。假设某个函数的原型为： `void foo(int x, int y);` 该函数被 C 编译器编译后在库中的名字为 `_foo`，而 C++编译器则会产生像 `_foo_int_int` 之类的名字。C++ 提供了 C 连接交换指定符号 `extern“C”` 来解决名字匹配问题。

- 55、请简述以下两个 for 循环的优缺点。

```
for (i=0; i<N; i++)  
{  
    if (condition)  
        DoSomething();  
    else  
        DoOtherthing();  
}
```

```
if (condition)  
{  
    for (i=0; i<N; i++)  
        DoSomething();  
}  
else  
{  
    for (i=0; i<N; i++)  
        DoOtherthing();  
}
```

【标准答案】

```
for (i=0; i<N; i++)
{
    if (condition)
        DoSomething();
    else
        DoOtherthing();
}
```

优点：程序简洁

缺点：多执行了 N-1 次逻辑判断，并且打断了循环“流水线”作业，使得编译器不能对循环进行优化处理，降低了效率。

```
if (condition)
{
    for (i=0; i<N; i++)
        DoSomething();
}
else
{
    for (i=0; i<N; i++)
        DoOtherthing();
}
```

优点：循环的效率 high

缺点：程序不简洁

- 56、语句for(; 1 ;)有什么问题？它是什么意思？

【标准答案】死循环，和while(1)相同。

- 57、 do.....while和while.....do有什么区别？

【标准答案】 前一个循环一遍再判断，后一个判断以后再循环。

- 58、请写出下列代码的输出内容

```
#include <stdio.h>
int main()
{
    int a,b,c,d;
    a=10;
    b=a++;
    c=++a;
    d=10*a++;
    printf("b, c, d: %d, %d, %d", b, c, d) ;
    return 0;
}
```

【标准答案】 10, 12, 120

- 59、 unsigned char *p1;
unsigned long *p2;
p1=(unsigned char *)0x801000;
p2=(unsigned long *)0x810000;
请问p1+5= ;
p2+5= ;

【标准答案】 0x801005、 0x810020

```
60、 main()
{
    int a[5]={1,2,3,4,5};
    int *ptr=(int *)(&a+1);
    printf("%d, %d",*(a+1),*(ptr-1));
}
```

请问输出：

【标准答案】 2,5

- 61、请问下面程序有什么错误？

```
int a[60][250][1000],i,j,k;  
for(k=0;k<=1000;k++)  
    for(j=0;j<250;j++)  
        for(i=0;i<60;i++)  
            a[i][j][k]=0;
```

【标准答案】 把循环语句内外换一下。

- 62、以下是求一个数的平方的程序,请找出错误:

```
#define SQUARE(a)((a)*(a))
```

```
int a=5;
```

```
int b;
```

```
b=SQUARE(a++);
```

【标准答案】宏在预编译时会以替换的形式展开，仅仅会替换。涉及到宏的地方，不要用++ --，标准中对此没有规定，因此最终结果将会依赖于不同的编译器。执行程序的答案可能是25、也有可能是36。

```
63、 #define Max_CB 500
void LmiQueryCSmd(Struct MSgCB * pmsg)
{
    unsigned char ucCmdNum;
    .....
    for(ucCmdNum=0;ucCmdNum<Max_CB;ucCmdNum++)
    {
        .....;
    }
```

}这段代码执行有什么问题？

【标准答案】死循环

unsigned char //无符号字符型 表示范围0~255

char //有符号字符型 表示范围-128~127

- 64、嵌入式系统中经常要用到无限循环，你怎么用C编写死循环。

【标准答案】 `while(1){}`或者`for(;;)`

65、程序输出结果 是？

```
int sum(int a)
{
    auto int c=0;
    static int b=3;
    c += 1;
    b += 2;
    return(a+b+c);
}

void main()
{
    int I;
    int a = 2;
    for(I = 0; I < 5; I++)
    {
        printf("%d,", sum(a));
    }
}
```

【标准答案】 8, 10, 12, 14, 16

66、

```
int func(int a)
{
    int b;
    switch(a)
    {
        case 1:
            b = 30;
        case 2:
            b = 20;
        case 3:
            b = 16;
        default: 0;
    }
    return b;
}

int main()
{
    printf("%d",func(1));
}
```

【标准答案】 16

```
67、 int modifyvalue()
{
    return(x+=10);
}
int changevalue(int x)
{
    return(x+=1);
}
void main()
{
    int x=10;
    x++;
    changevalue(x);
    x++;
    modifyvalue();
    printf("First output:%dn",x);
    x++;
    changevalue(x);
    printf("Second output:%dn",x);
    modifyvalue();
    printf("Third output:%dn",x);
}输出?
```

【标准答案】 12、 13、 13

- 68、不能做switch()的参数类型是：
【标准答案】 switch的参数不能为实型。

- 69、请写出下列代码的输出内容

```
#include<stdio.h>
main()
{
int a,b,c,d;
a=10;
b=a++;
c=++a;
d=10*a++;
printf("b, c, d: %d, %d, %d", b, c, d) ;
return 0;
}
```

【标准答案】 10, 12, 120

70、找出程序的错误。

```
main()
{
    char*src="hello, world";
    char* dest=NULL;
    int len=strlen(src);
    dest=(char*)malloc(len);
    char* d=dest;
    char* s=src[len];
    while(len--!=0)
        d++=s--;
    printf("%s", dest);
    return 0;
}
```

【标准答案】

```
int main()
{
    char* src = "hello, world";
    int len = strlen(src);
    char* dest = (char*)malloc(len+1); //
    要为\0分配一个空间
    char* d = dest;
    char* s = &src[len-1]; //指向最后一个
    字符
    while( len-- != 0 )
        *d++=*s--;
    *d = 0; //尾部要加\0
    printf("%s\n", dest);
    free(dest); // 使用完，应当释放空间，
    以免造成内存泄露
    return 0;
}
```

- 71、一语句实现x是否为2的若干次幂的判断。

【参考答案】 void main()

```
{  
    int a;  
    scanf("%d",&a);  
    printf("%c",(a)&(a-1)?'n':'y'); // 若是打印y，否则n  
}
```

- 72、中断是嵌入式系统中重要的组成部分，这导致了很多编译开发商提供一种扩展—让标准C支持中断。具代表事实是，产生了一个新的关键字__interrupt。下面的代码就使用了__interrupt关键字去定义了一个中断服务子程序(ISR)，请评论一下这段代码的。

```
__interrupt double compute_area (double radius)
{
    double area = PI * radius * radius;
    printf(" Area = %f", area);
    return area;
}
```

【参考答案】这个函数有太多的错误了，以至让人不知从何说起了：

- 1). ISR 不能返回一个值。如果你不懂这个，那么你不会被雇用的。
- 2). ISR 不能传递参数。如果你没有看到这一点，你被雇用的机会等同第一项。
- 3). 在许多的处理器/编译器中，浮点一般都是不可重入的。有些处理器/编译器需要让额处的寄存器入栈，有些处理器/编译器就是不允许在ISR中做浮点运算。此外，ISR应该是短而有效率的，在ISR中做浮点运算是不明智的。
- 4). 与第三点一脉相承，printf()经常有重入和性能上的问题。如果你丢掉了第三和第四点，我不会太为难你的。不用说，如果你能得到后两点，那

- 73、下面的代码输出是什么，为什么？

```
void foo(void)
{
    unsigned int a = 6;
    int b = -20;
    (a+b > 6)? puts("> 6") : puts("<= 6");
}
```

【参考答案】 这个问题测试你是否懂得C语言中的整数自动转换原则，我发现有些开发者懂得极少这些东西。不管怎样，这无符号整型问题的答案是输出是“>6”。原因是当表达式中存在有符号类型和无符号类型时所有的数都自动转换为无符号类型。因此-20变成了一个非常大的正整数，所以该表达式计算出的结果大于6。这一点对于应当频繁用到无符号数据类型的嵌入式系统来说是非常重要的。如果你答错了这个问题，你也就到了得不到这份工作的边缘。

● 74、评价下面的代码片断：

```
unsigned int zero = 0;  
unsigned int compzero = 0xFFFF;  
/*1's complement of zero */
```

【参考答案】 对于一个int型不是16位的处理器为说，上面的代码是不正确的。应编写如下：

```
unsigned int compzero = ~0;
```

这一问题真正能揭露出应试者是否懂得处理器字长的重要性。在我的经验里，好的嵌入式程序员非常准确地明白硬件的细节和它的局限，然而PC机程序往往把硬件作为一个无法避免的烦恼。

到了这个阶段，应试者或者完全垂头丧气了或者信心满满志在必得。如果显然应试者不是很好，那么这个测试就在这里结束了。但如果显然应试者做得不错，那么我就扔出下面的追加问题，这些问题是比较难的，我想仅仅非常优秀的应试者能做得不错。提出这些问题，我希望更多看到应试者应付问题的方法，而不是答案。不管如何，你就当是这个娱乐吧...

- 75、下面的代码片段的输出是什么，为什么？

```
char *ptr;  
if ((ptr = (char *)malloc(0)) == NULL)  
    puts("Got a null pointer");  
else  
    puts("Got a valid pointer");
```

【参考答案】这个你可以先大胆猜测下，然后再用你的编译器尝试着编译下~~

欢迎进入C语言程序笔试面试，编写程序代码单元。

```
char *strcpy(char *strDest, const char *strSrc);
```

76、已知 const strSrc (1) strcpy

```
{
    assert((strDest!=NULL) && (strSrc !=NULL));
    char *address = strDest;
    while( (*strDest++ = *strSrc++) != '\0' )
        NULL ;
    return address ;
}
```

- (2) strcpy 能把 strSrc 的内容复制到 strDest, 为什么还要 char * 类型的返回值?

答：为了实现链式表达式。

// 2分

例如 `int length = strlen( strcpy( strDest, "hello world" ) );`

77、写出二分查找的代码。

【参考答案】

```
int binary_search(int* arr, int key, int n)
{
    int low = 0;
    int high = n - 1;
    int mid;
    while (low <= high)
    {
        mid = (high + low) / 2;
        if (arr[mid] > key)
            high = mid - 1;
        else if (arr[mid] < key)
            low = mid + 1;
        else
            return mid;
    }
    return -1;
}
```

- 78、请编写一个 C 函数，该函数给出一个字节中被置 1 的位的个数。

【参考答案】

```
unsigned int TestAsOne0(char log)
{
    int i;
    unsigned int num=0, val;
    for(i=0; i<8; i++)
    {
        val = log >> i; //移位
        val &= 0x01; //与1相与
        if(val)
            num++;
    }
    return num;
}
```

- 79、请编写一个 C 函数，该函数将给定的一个字符串转换成整数。

【参考答案】

```
int Invert(char *str)
{
    int num=0;
    while(*str!='\0')
    {
        int digital=*str-48;
        num=num*10+digital;
        str=str+1;
    }
    return num;
}
```

- 80、请编写一个C函数，该函数将给定的一个整数转换成字符串。

【参考答案】

```
void IntroCharChange(int num, char* pval)
{
    char strval[100];
    int i, j;
    int val0 = 0;
    int val1 = 0;
    val0 = num;
    for(i=0; i<100; i++)
    {
        val1 = val0 % 10; //取余
        val0 = val0 / 10; //取整
        strval[i] = val1 + 48; //数字—字符
        if(val0 < 10)
        {
            i++;
            strval[i] = val0 + 48;
            break;
        }
    }
    for(j=0; j<=i; j++) //倒置
        pval[j] = strval[i-j];
    pval[j] = '\0';
}
```

81、实现strcmp函数。

【参考答案】

```
int mystrcmp(const char* str1, const char* str2)
{
    assert((str1 != NULL) && (str2 != NULL));
    int ret = 0;
    while (!(ret = *(unsigned char*)str1 - *(unsigned char*)str2) && *str2)
    {
        str1++;
        str2++;
    }
    if (ret > 0)
        ret = 1;
    else if (ret < 0)
        ret = -1;
    return ret;
}
```

- 82、请编写一个 C 函数，该函数将一个字符串逆序。

【参考答案】

```
void AntitoneValue(char* father, char* child)
{
    int i;
    char source[100];
    int j = 0;
    while(father[j]) //放入source, [j]为长度
    {
        source[j] = father[j];
        j++;
        if(j > 99)
            return;
    }
    source[j] = '\0';
    for(i=0; i<j; i++)
        child[i] = source[j-i-1]; //反序
    child[i] = '\0';
}
```

- 83、请编写一个 C 函数，该函数在给定的内存区域搜索给定的字符，并返回该字符所在位置索引值。

【参考答案】

```
int search(char *cpSource, int n, char ch) //起始地址，搜索长度，目标字符
{
    int i;
    for(i=0; i<n && *(cpSource+i) != ch; ++i);
    return i;
}
```

- 84、请编写一个 C 函数，该函数在一个字符串中找到可能的最长的子字符串，该字符串是由同一字符组成的。

【参考答案】

```
int ChildString(char*p) //自己写
{
    char *q=p;
    int stringlen=0, i=0, j=1, len=0, maxlen=1;
    while(*q!='\0') //不能用strlen,求得长度stringlen
    {
        Stringlen++;
        q++;
    }
    while( i< Stringlen )
    {
        if(*(p+i)==*(p+j)&&j< Stringlen)
        {
            len++; //统计子串长度
            i++;
            j++;
        }
        else
        {
            if(len>maxlen) //统计最大子串长度
            {
                maxlen=len+1;
                len=0;
            }
            else
            {
                len=0;
            }
            i++;
            j++;
        }
    }
    return maxlen;
}
```

● 85、华为面试题：怎么判断链表中是否有环？

【参考答案】答：用两个指针来遍历这个单向链表，第一个指针p1，每次走一步；第二个指针p2，每次走两步；当p2 指针追上 p1的时候，就表明链表当中有环路了。

```
int    testLinkRing(Link *head)
{
    Link *t1=head,*t2=head;
    while( t1->next && t2->next)
    {
        t1 = t1->next;
        if (NULL == (t2 = t2->next->next))
            return 0; //无环
        if (t1 == t2)
            return 1;
    }
    return 0;
```

- 86、有一浮点型数组A,用C语言写一函数实现对浮点数组A进行降序排序,并输出结果,要求要以数组A作为函数的入口.(建议用冒泡排序法)

【参考答案】

```
void BubbleSort(double arr[], int n)
```

```
{
```

```
    int i, j;  
    int exchange = 1;    //交换标志  
    for(i=1;i<n;i++)
```

```
    { //最多做n-1趟排序
```

```
        exchange=0;    //本趟排序开始前,交换标志应为假  
        for(j=n-1;j>=i; j--) //对当前无序区R[i..n]自下向上扫描  
            if(arr[j+1] > arr[j])
```

```
            { //交换记录
```

```
                arr[0]=arr[j+1];    //R[0]不是哨兵,仅做暂存单元
```

```
                arr[j+1]=arr[j];
```

```
                arr[j]=arr[0];
```

```
                exchange=1;    //发生了交换,故将交换标志置为真
```

```
            }
```

```
        } if(!exchange) //本趟排序未发生交换,提前终止算法
```

```
            return;
```

```
    } //endfor(外循环)
```

```
}
```

- 87、实现双向链表删除一个节点P，在节点P后插入一个节点，写出这两个函数。

【参考答案】

//删除操作

```
Status ListDelete_DuL(DuLinkList &L,int i,ElemType &e)
{
    if(!(p=GetElemP_DuL(L,i))) return ERROR;
    e=p->data;
    p->prior->next=p->next;
    p->next->prior=p->prior;
    free(p);
    return OK;
}
```

//插入操作

```
Status ListInsert_DuL(DuLinkList &L,int i,ElemType &e)
{
    if(!(p=GetElemP_DuL(L,i)))
        return ERROR;
    if(!(s=(DuLinkList)malloc(sizeof(DuLNode))))
        return ERROR;
    s->data=e;
    s->prior=p;
    p->next->prior=s;
    p->next=s;

    s->next=p->next->next;
    return OK;
}
```

88、把一个链表反向。

【参考答案】从第一个元素开始，ps指向他，将他（ps）指向头节点（ps->next = head），将ps设为头节点（head = ps;）操作下一个元素（ps = ps->next;）等于是依次将每个元素翻到原头节点前面。

```
void reverse(test* head)
{
    test* pe = head;
    test* ps = head->next;
    while(ps)
    {
        pe->next = ps->next;
        ps->next = head;
        head = ps;
        ps = ps->next;
    }
}
```

- 89、将二维数组行列元素互换，存到另一个数组中。

【参考答案】

```
#include <stdio.h>
```

```
main()
```

```
{ int a[2][3]={1,2,3},{4,5,6}};
```

```
    int b[3][2],i,j;
```

```
    printf("array a:\n");
```

```
    for(i=0;i<=1;i++)
```

```
    { for(j=0;j<=2;j++)
```

```
        { printf("%5d",a[i][j]);
```

```
          b[j][i]=a[i][j];
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
    printf("array b:\n");
```

```
    for(i=0;i<=2;i++)
```

```
    { for(j=0;j<=1;j++)
```

```
        printf("%5d",b[i][j]);
```

```
        printf("\n");
```

```
    }
```

```
}
```

- 90、输入一行字符，统计其中有多少个单词。

【参考答案】 `#include <stdio.h>`

`main()`

`{`

`char string[81];`

`int i,num=0,word=0;`

`char c;`

`gets(string);`

`for(i=0;(c=string[i])!='\0';i++)`

`if(c==' ')`

`word=0;`

`else if(word==0)`

`{`

`word=1; num++; }`

`printf("There are %d words in the line\n",num);`

- 91、写一个内存拷贝函数,不用任何库函数.就是前些时候本版讨论的那个问题。

【参考答案】

```
void* memcpy(void* pvTo, const void* pvFrom, size_t size)
{
    assert((pvTo != NULL) && (pvFrom != NULL));
    byte* pbTo = pvTo;
    byte* pbFrom = pvFrom;
    while (size-- > 0)
    {
        *pbTo++ = *pbFrom++;
    }
    return pvTo;
}
```

- 92、有1、2、3、4个数字，能组成多少个互不相同且无重复数字的三位数？都是多少？

【参考答案】

```
#include "stdio.h"
#include "conio.h"
main()
{
    int i,j,k;
    printf("\n");
    for(i=1;i<5;i++) /*以下为三重循环*/
        for(j=1;j<5;j++)
            for (k=1;k<5;k++)
                {
                    if (i!=k&&i!=j&&j!=k) /*确保i、j、k三位互不相同*/
                        printf("%d,%d,%d\n",i,j,k);
                }
    getch();
}
```

- 93、取一个整数a从右端开始的4~7位。

【参考答案】

```
main()
{
    unsigned a,b,c,d;
    scanf("%o",&a);
    b=a>>4;
    c=~(~0<<4);
    d=b&c;
    printf("%o\n%o\n",a,d);
}
```

- 94、打印出杨辉三角形（要求打印出10行如下图）。

【参考答案】

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
```

```
main()
{
    int i,j;
    int a[10][10];
    printf("\n");
    for(i=0;i<10;i++)
    {
        a[i][0]=1;
        a[i][i]=1;
    }
    for(i=2;i<10;i++)
    for(j=1;j<i;j++)
    a[i][j]=a[i-1][j-1]+a[i-1][j];
    for(i=0;i<10;i++)
    {
        for(j=0;j<=i;j++)
        printf("%5d",a[i][j]);
        printf("\n");
    }
    getch();
}
```

95、实现strcmp函数。

```
int mystrcmp(const char* str1, const char* str2)
{
    assert((str1 != NULL) && (str2 != NULL));
    int ret = 0;
    while (!(ret = *(unsigned char*)str1 - *(unsigned char*)str2) && *str2)
    {
        str1++;
        str2++;
    }
    if (ret > 0)
        ret = 1;
    else if (ret < 0)
        ret = -1;
    return ret;
}
```

- 96、写一个函数，求一个字符串的长度，在main函数中输入字符串，并输出其长度。

【参考答案】

```
main()
{
    int len;
    char *str[20];
    printf("please input a string:\n");
    scanf("%s",str);
    len=length(str);
    printf("the string has %d characters.",len);
    getch();
}

length(p)
char *p;
{
    int n;
    n=0;
    while(*p!='\0')
    {
        n++;
        p++;
    }
    return n;
}
```

- 97、 $809 * ?? = 800 * ?? + 9 * ?? + 1$ 其中??代表的两位数, $8 * ??$ 的结果为两位数, $9 * ??$ 的结果为3位数。求??代表的两位数, 及 $809 * ??$ 后的结果。

【参考答案】

```
output(long b,long i)
{
    printf("\n%d/%d=809*%d+%d",b,i,i,b%i);
}
main()
{
    long int a,b,i;
    a=809;
    for(i=10;i<100;i++)
    {
        b=i*a+1;
        if(b>=1000&&b<=10000&&8*i<100&&9*i>=100)
            output(b,i);
    }
    getch();
}
```

- 98、某个公司采用公用电话传递数据，数据是四位的整数，在传递过程中是加密的，加密规则如下：每位数字都加上5,然后用和除以10的余数代替该数字，再将第一位和第四位交换，第二位和第三位交换。

【参考答案】

```
main()
{
    int a,i,aa[4],t;
    scanf("%d",&a);
    aa[0]=a%10;
    aa[1]=a%100/10;
    aa[2]=a%1000/100;
    aa[3]=a/1000;
    for(i=0;i<=3;i++)
    {
        aa[i]+=5;
        aa[i]%=10;
    }
    for(i=0;i<=3/2;i++)
    {
        t=aa[i];
        aa[i]=aa[3-i];
        aa[3-i]=t;
    }
    for(i=3;i>=0;i--)
        printf("%d",aa[i]);
}
```

getch();

99、计算字符串中子串出现的次数。

【参考答案】

```
main()
{
```

```
    char str1[20],str2[20],*p1,*p2;
    int sum=0;
    printf("please input two strings\n");
    scanf("%s%s",str1,str2);
    p1=str1;p2=str2;
    while(*p1!='\0')
    {
        if(*p1==*p2)
        {
            while(*p1==*p2&&*p2!='\0')
            {
                p1++;
                p2++;
            }
        }
        else
        {
            p1++;
            if(*p2=='\0')
            {
                sum++;
                p2=str2;
            }
        }
    }
    printf("%d",sum);
    getch();
```

- 100、有两个磁盘文件A和B,各存放一行字母,要求把这两个文件中的信息合并（按字母顺序排列），输出到一个新文件C中。

【参考答案】

100条经典C语言笔试题目

```
main()
{
    FILE *fp;
    int i,j,n,ni;
    char c[160],t,ch;
    if((fp=fopen("A","r"))==NULL)
    {
        printf("file A cannot be opened\n");exit(0);
    }
    printf("\n A contents are :\n");
    for(i=0;(ch=fgetc(fp))!=EOF;i++)
    {
        c[i]=ch;putchar(c[i]);
    }
    fclose(fp);
    ni=i;
    if((fp=fopen("B","r"))==NULL)
    {
        printf("file B cannot be opened\n");exit(0);
    }
    printf("\n B contents are :\n");
    for(i=0;(ch=fgetc(fp))!=EOF;i++)
    {
        c[i]=ch;putchar(c[i]);
    }
    fclose(fp);
    n=i;
    for(i=0;i<n;i++)
        for(j=i+1;j<n;j++)
            if(c[i]>c[j]){t=c[i];c[i]=c[j];c[j]=t;}
    printf("\n C file is:\n");
    fp=fopen("C","w");
    for(i=0;i<n;i++){putc(c[i],fp);putchar(c[i]);}
    fclose(fp);
    getch();
}
```