

C 语言编程笔试题-C 卷

一、判断题（对的写 T，错的写 F 并说明原因，每小题 3 分，共 15 分）

1、 #define TRUE 1

#define FALSE 0

int x = 2;

int y = 0;

if(x)

{

y++; F,y自增后变成了1

}。则 y 等于 0。（ ）

原因:

2、 void main()

{

unsigned char i = 0;

int sum = 0;

for(i=0; i < 256;i++)

{

sum += 1;

}

printf("sum = %d ", sum);

} 执行此函数将会打印 sum = 256。（ F, i的最大值才是255，这会是个无限循环

原因:

3、 switch(c) 语句中 c 可以是 int, long, char, double, unsigned int 类型。（ ）

原因: F, switch 只能判断整型变量。

4、 void GetMemory(char **p, int num)

{

*p = (char *) malloc(num);

}

void main()

{

char *str = NULL;

GetMemory(&str, 100);

strcpy(str, " hello ");

printf(str);

}

任何时候运行 main 函数将会打印"hello"字符。（ ）

原因: T,正确，因为*p 指向的是str,正确给str分配了内存空间。

5、 void main()

```

{
    char temp[20000000];
    int i = 0;
    int sum = 0;

    memset(temp, 0x01, sizeof(temp));
    for(i=0; < sizeof(temp)/sizeof(int); i++)
    {
        sum += temp[i];
    }

    printf("sum=%d",sum);
}

```

F,因为代码写错了，sizeof(temp)/sizeof(temp[0])就正确了

任何时候运行 main()函数将会打印"20000000"字符。()

原因:

二、填空题（每空 2 分，共 40 分）

1、int a()

```

{
    static int i = 0;
    if(i >= 1)
    {
        return --i;
    }
    return i++;
}

void main()
{
    int A1 = 0;
    int A2 = 0;
    int A3 = 0;
    A1 = a();
    A2 = a();
    A3 = a();
    printf("A1 = %d, A2 = %d, A3 = %d ", A1, A2,A3);
}

```

0, 0, 0 , 这题需要注意，return i++,是先return i,再++的
程序打印的 A1,A3,A3 的值分别为 (), (), ()。

2、int add(int a, int b)

```

{
    return ((a++) + (++b));
}

void main()
{
    int a = 1;
}

```

```

int b = 2;
int c = 0;
c = add(a,b);
printf("a = %d, b = %d, c = %d ", a, b, c);
}

```

a,b传入的是副本，本身的值不会发生变化，c=4,因为是++b。

程序打印的 a, b, c 的值分别为 (), (), ()。

3、在 Windows 64 位系统下，写出运行结果：

```

char str[ ] = " Hello " ;
char *p = str;
int n = 10;
struct AAA
{
    int a;
    char b;
};

```

sizeof(str) = (8) str是数组，应该是数组大小6，5个字符加一个字符结束标志

sizeof(p) = (8)

sizeof(n) = (4)

sizeof(struct AAA) = (8) int大小是4，内存对齐的情况下，char 也占用4个字节

```

void func(char str[100])
{
    sizeof(str) = (100)
}

```

64位系统指针大小是8，32地址，指针大小是4

4、int arg[] = {0, 1, 2, 3};

```

int * p = &arg[1];
int v = 0;
int w = 0;
*(p++) += 5;
v = *p;
*p = *p + 5;
w = *p;

```

这里 v 的值等于 (2)，w 的值等于 (7)，数组 arg[0]、arg[1]、arg[2]、arg[3] 的值分别等于 (0)、(6)、(7)、(3)。

具体解析在「嵌入式Linux」微信公众号

5、int a[] = {1,2,3};

```

void test(int ** v)
{
    *v = a + 1;
    return;
}

void main()
{

```

```

int * p;
test(&p);
memset(p, 0, sizeof(int)*2);
printf("a[0] %d, a[1] %d, a[2] %d", a[0], a[1], a[2]);
}

```

程序运行 main 函数后，打印的 a[0]、a[1]、a[2] 分别等于 (1)、(0)、(0)。

这题可以引申修改成 $*v = \&a+1$;

三、编程题（共 45 分）

单向、非循环的链表节点存储的学生信息包括 “IDx” 和 “Name” 两个属性，如下“学生信息 链表结点定义”：

```

//学生信息 链表结点定义
typedef struct _STUDENT_INFO{
{
    int IDx;
    char Name[32]
    struct _STUDENT_INFO * Next;
}STUDENT_INFO_DEF;

```

请完成两个函数：

1)查找指定的学生序号 IDx 的链表节点，把此节点后面的所有节点都删除掉，并返回新的链表。例如指定的学生序号 IDx 是 20，链表有 5 个节点，第 3 个节点的 IDx 是 20，那么删除第 4、5 节点，返回包含前面 3 个节点的新链表。

2)将学生链表 A 按序号 IDx 的奇数、偶数分解成 2 个新的链表 A1、A2，其中 A1 是 IDx 为奇数的链表，A2 是 IDx 为偶数的链表。例如链表 A 有 4 个节点，IDx 分别为 1、2、5、6，那么分解后的链表 A1 有 2 个节点、IDx 是 1 和 5，分解后的链表 A2 有 2 个节点、IDx 是 2 和 6。

（注：不能使用 STL 链表库，可以使用 malloc(), free(),memset(),memcpy() 标准库）

// 函数说明：查找学生 IDx 的链表节点，并把此节点后面的所有节点都删除掉，并返回新的链表。

// 函数输入：

STUDENT_INFO_DEF * Head 输入的链表头指针，int IDx 指定的学生序号

// 返回值：新的链表头指针

STUDENT_INFO_DEF * DeleteList(STUDENT_INFO_DEF * Head, int IDx)

{

/*请完成此函数*/

```

    }
    // 函数说明：将学生链表 A 按序号 IDx 的奇数、偶数分解成 2 个新的链表 A1、A2，其中
    A1 是 IDx 为奇数的链表，A2 是 IDx 为偶数的链表。
    // 函数输入：
    STUDENT_INFO_DEF * A --- 链表 A 的头指针
    STUDENT_INFO_DEF ** A1 --- 生成的链表 A1 的头指针
    STUDENT_INFO_DEF ** A2 --- 生成的链表 A2 的头指针
    // 返回值：成功返回 0，失败返回-1
    int SplitList(STUDENT_INFO_DEF * A,
                  STUDENT_INFO_DEF **A1,
                  STUDENT_INFO_DEF **A2,
                  )
    {

        /*请完成此函数*/

```

```

    }

```