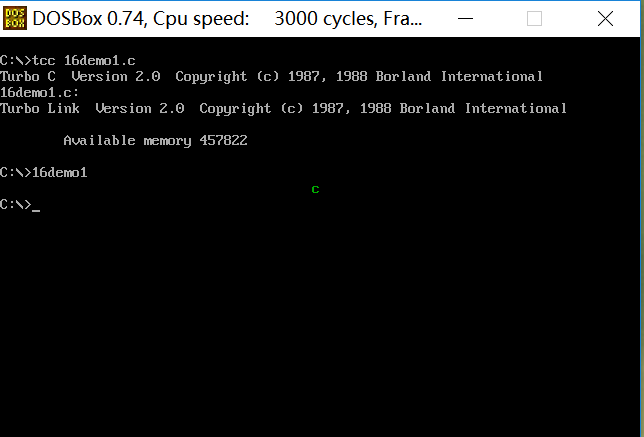
**c语言综合研究十六报告V1**

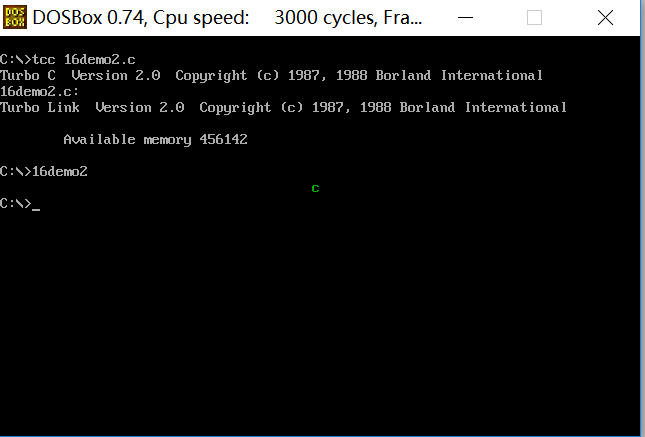
1. **研究过程展示**

**1>.程序1部分：**

运行程序16demo1：



运行程序16demo2：

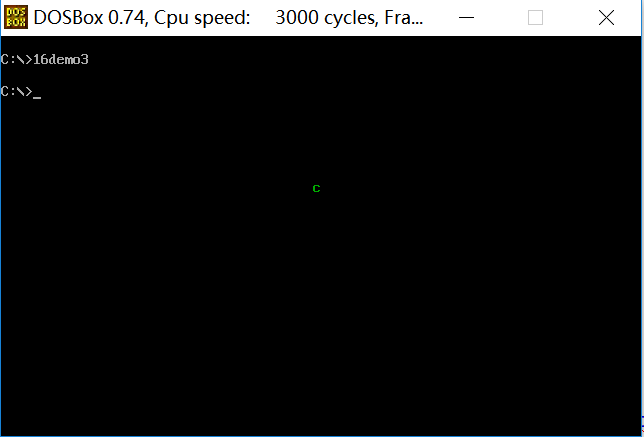


这两个程序的数据组织方式是一样的，都是使用结构体。但是a.c和b.c封装和分离的角度是不一样的。16demo1.c没有将字符和颜色的属性分离出来，而只是将显示功能分离出来，16demo2.c将字符、颜色的赋值和显示功能都分离了出来，用三个子函数实现，并将相对应的函数指针封装到结构体里去。

在面向对象的编程中，总是避免不了类的概念，而类里的封装还有一点就是比如“private int a;”等这样把数据“保护”，这也是一种封装，c++一般就是这样。“Public int setavalue(int);”等来调用，不仅是实现功能上的共性和个性分层，可扩展性的提升。在这里，我感觉16demo2.c中的程序有点向数据保护的方向靠拢了，而16demo1.c在mian()函数里面是直接对结构体中的属性值进行修改的做法。相较而言，第二个程序更有面向对象编程的封装思想，起码是表现出朝那个方向靠拢的意思了。

到了现在，一定要明白一个基本原则，我要实现的封装应该是除了避免程序代码重复之外，还要防止出现不必要的改变。把变化的东西封装起来，如何让我编的程序还能有很强的灵活性又是目前我要面对的问题。

运行程序16demo3：



填空处：#define new(x) (x\*)malloc(sizeof(x))

这里的宏名是new，学过java我们会发现java里初始化对象也是使用new，这里的new其实也是实现一个相似的功能。我们可以把结构体ch理解成一个类，用new对它进行实例化，这样就可以实现面向对象的程序设计思想。其实java里实例化对象也是开辟一个内存空间并给这个空间取一个名字即对象名。结构体为什么可以实现类的功能呢？我们知道，类里面可以定义变量、数组、函数，并进行一些操作如赋值、调用函数之类的，只是在java中类里面程序员不能定义和使用指针进行操作。而结构体里面也可以进行定义变量、数组、函数指针等的操作，所以如果我们要用c语言编写具有面向对象思想的程序，我们可以用结构体来实现类似“类”的功能，并用带参宏定义来实现实例化的功能，或者可以直接用malloc函数来实现实例化，只不过这样语句比较重复。

现实中，编写单片机程序的时候也会用到这种宏函数还有宏定义，因为效率会比一般的函数高，提高可移植性、可读性、方便性。不分配内存，避免了函数调用的开销，不过也有人说，现在机器速度快，宏函数的缺点比起这个更明显。这个应该也是有道理的，不然干嘛不都弄成宏函数呢？

1. **已思考研究并已解决问题汇总**
2. **已思考研究并未解决问题汇总**
3. 可以仅用c实现c++中真正的类的全部功能吗？没见过。
4. 现在有过程化的，面向对象的，还有宏，选择多了，有时候还要会进行合适的选择。
5. **研究感想（心得体会）**

这一次研究，自己暂时没有下手拓展程序观察想象，还是再多看看其他书籍，下一版会完善。