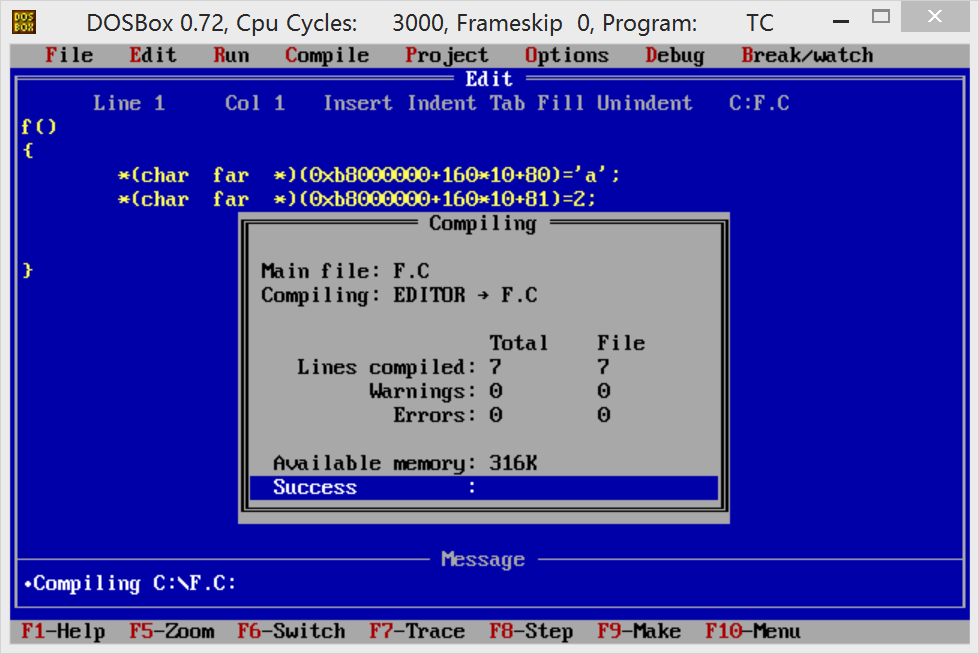
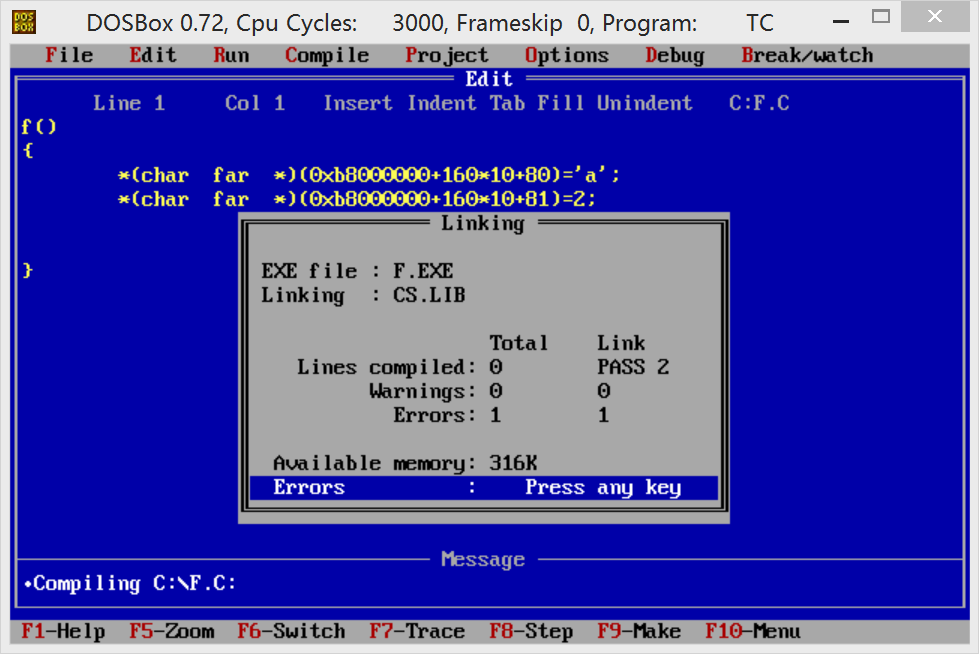
**综合研究四研究报告**

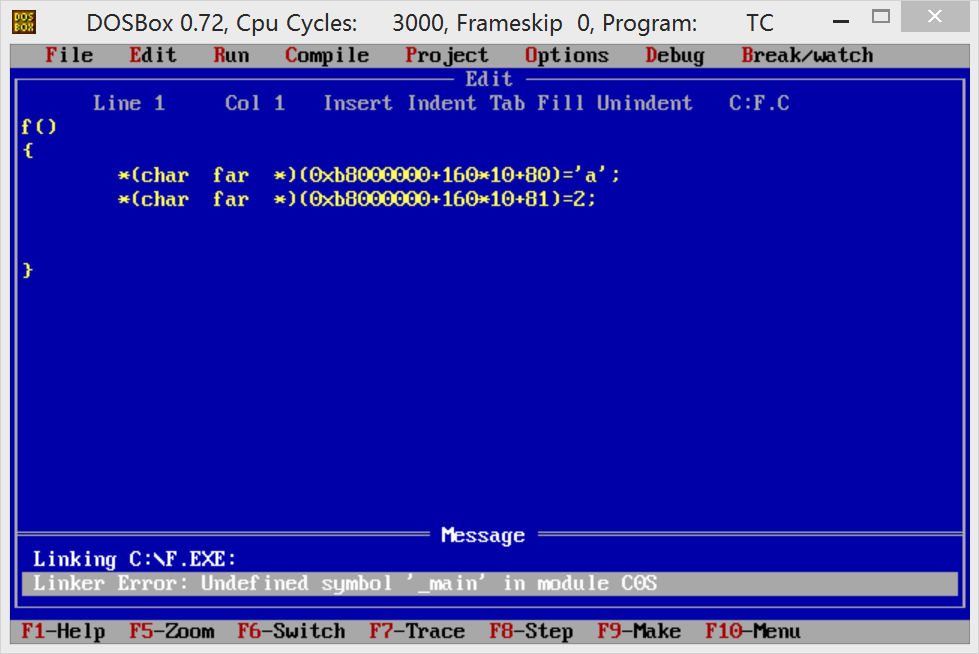
1. **研究过程展示**

1>.编写的f.c程序编译可以通过，但是连接的时候出现问题。如图：

****

****

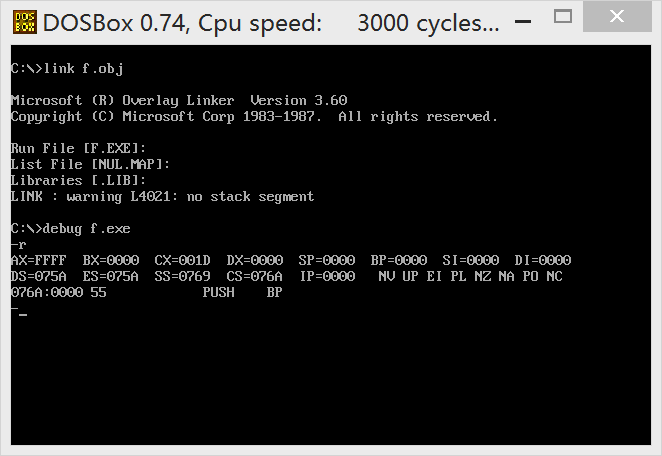
显示出的错误信息如图：

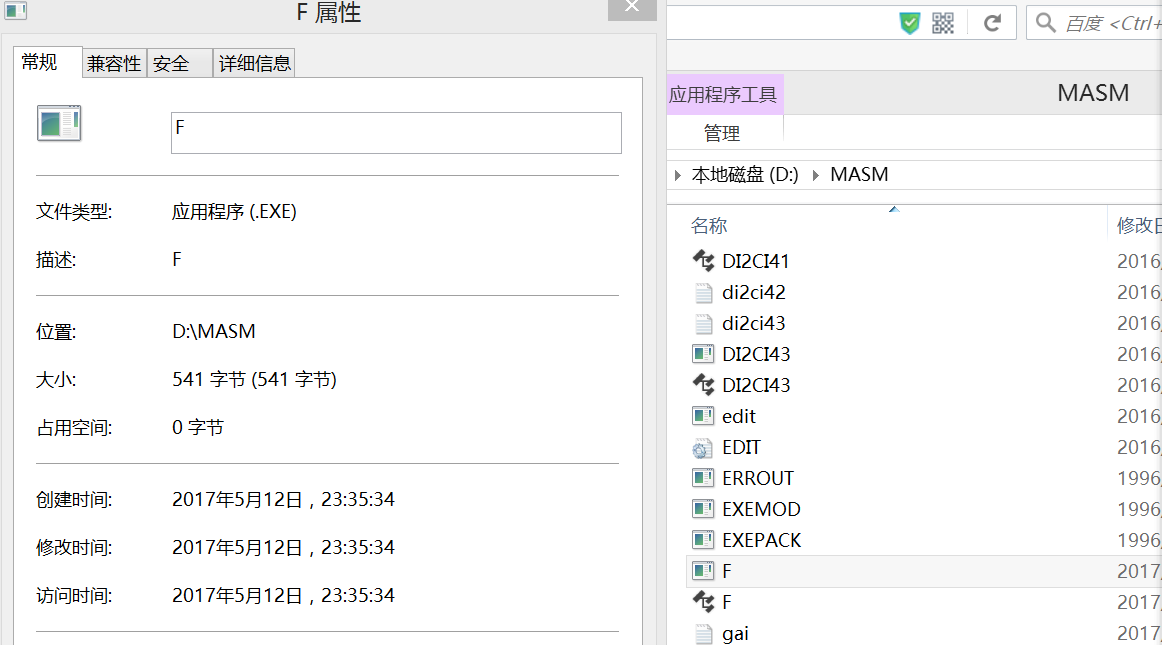


翻译成中文是：在c0s模块没有定义符号’\_main’。

这个错误信息应该与存放在**C:/minic**目录下的 **c0S.OBJ**文件有关。

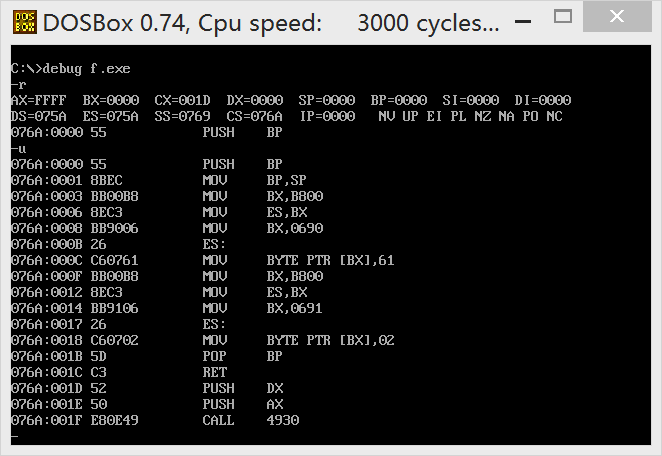
2>.此处cx=0x1d，即29字节。代码长度29字节。



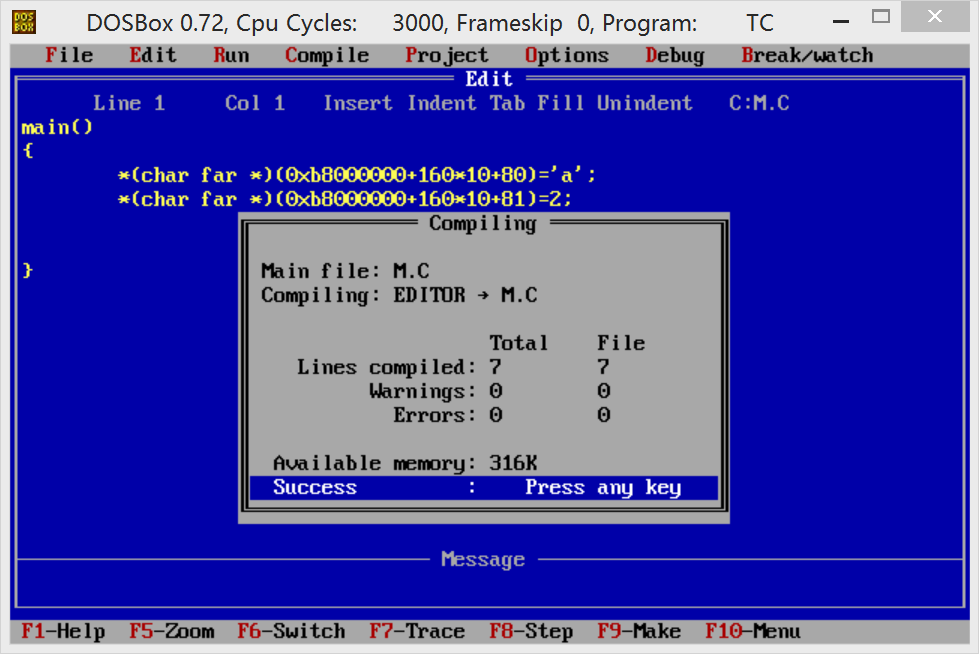


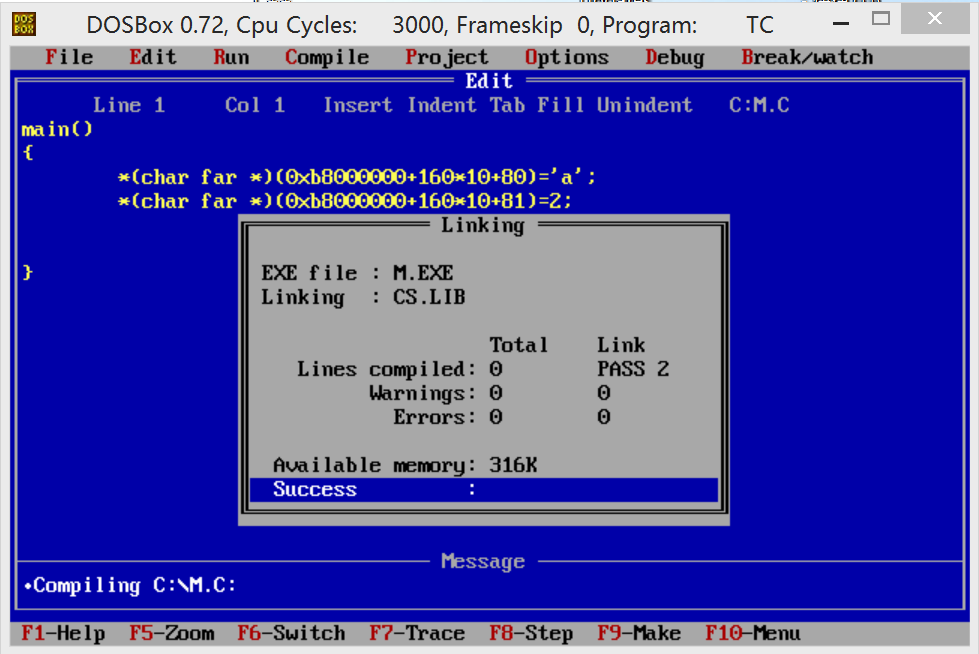
但是查看f.exe的文件属性时，又显示一个比29字节要大得多的541字节。

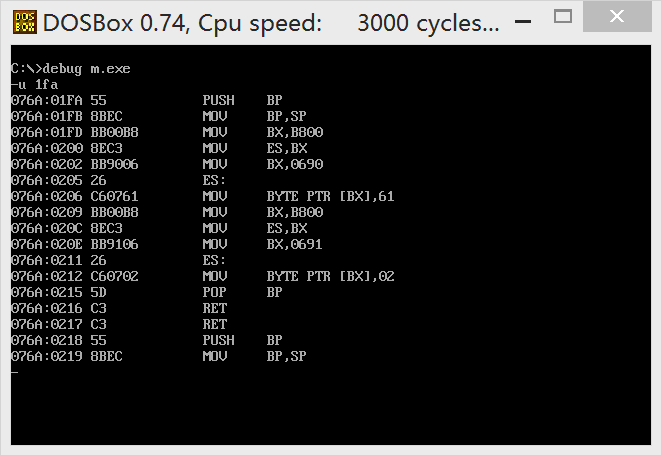
当直接用“u”命令反汇编的时候，正好出现了f.c程序的反汇编代码。可得，f函数的偏移地址是0。

****

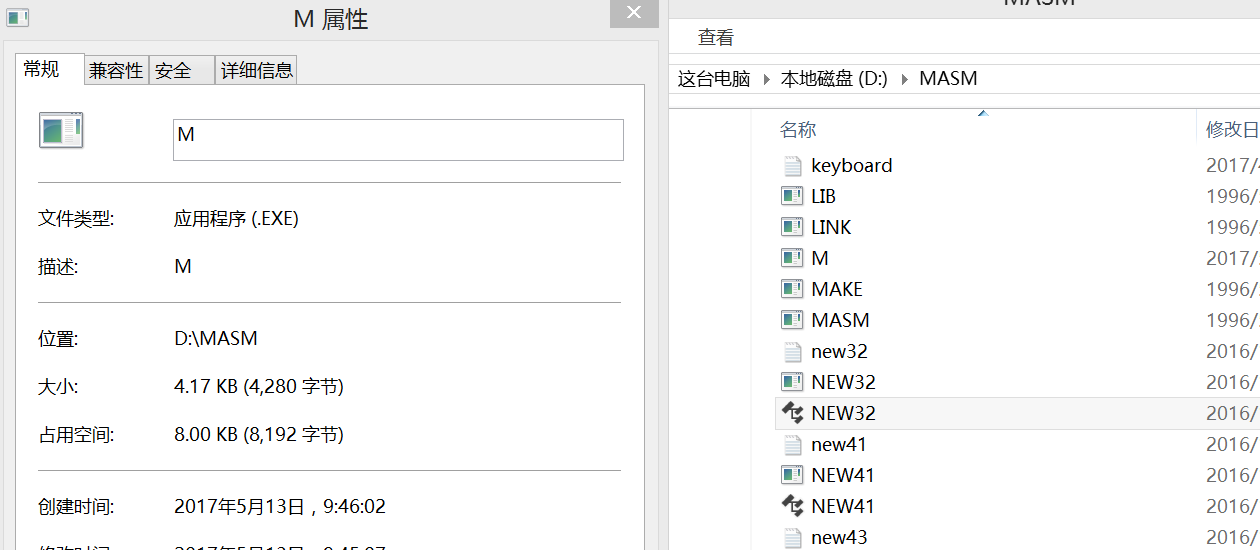
3>.用tc对m.c进行编译、连接，并用的debug查看m.exe整个程序的汇编代码。

****

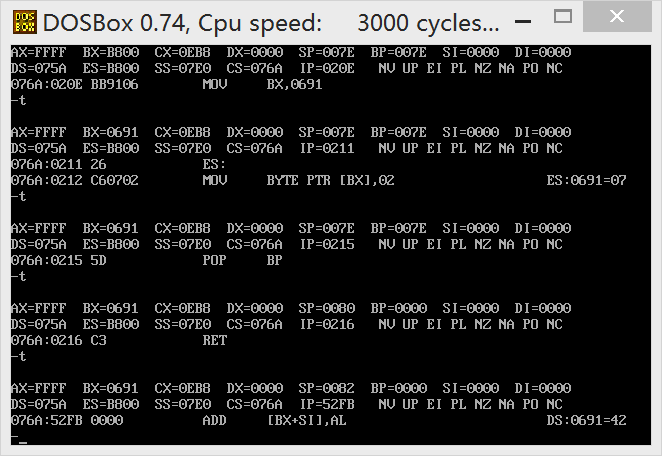
****

****

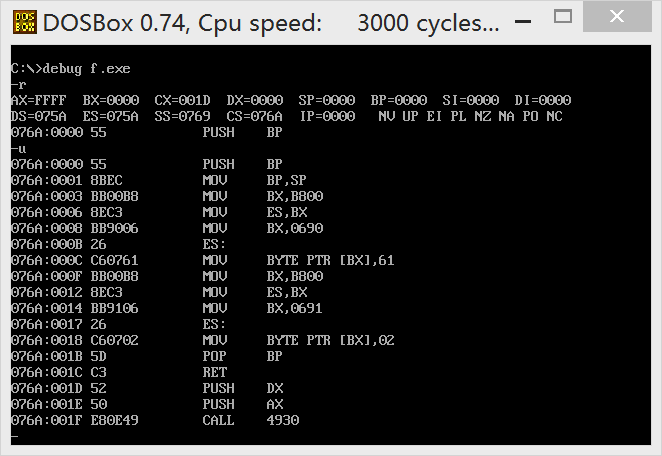
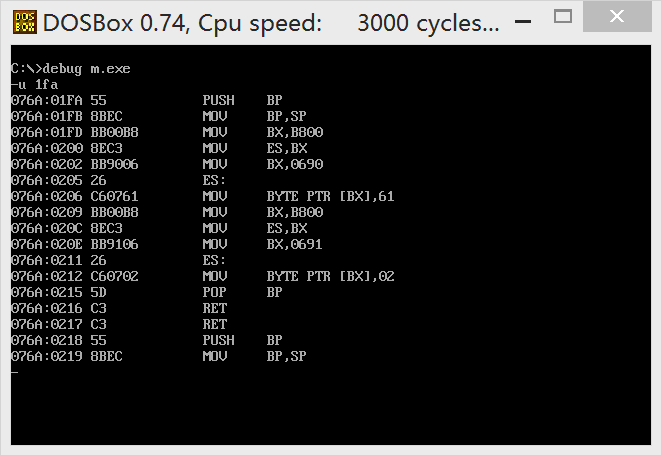
程序是从076a：01fa到076a：0216，一共29个字节。

****

属性查看的话，又是4280字节的大容量。

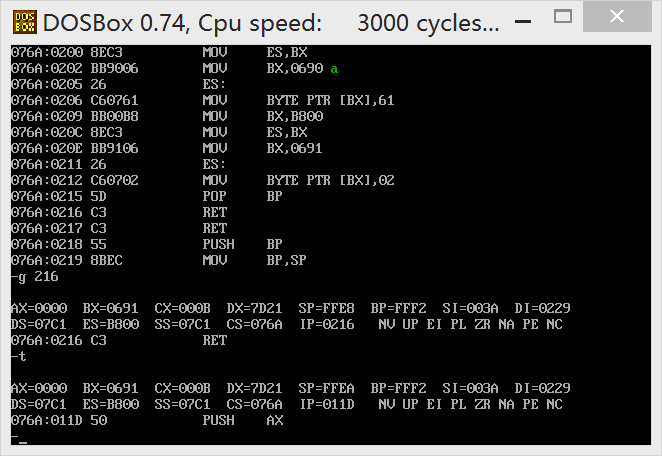
****

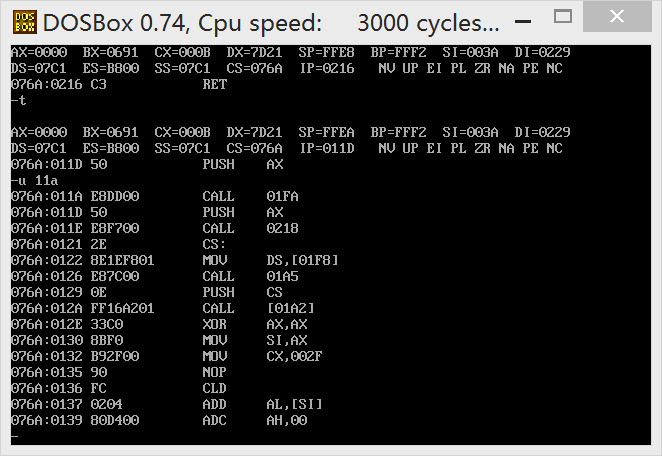
**m.exe的汇编代码与f.exe的汇编代码对比：**

****

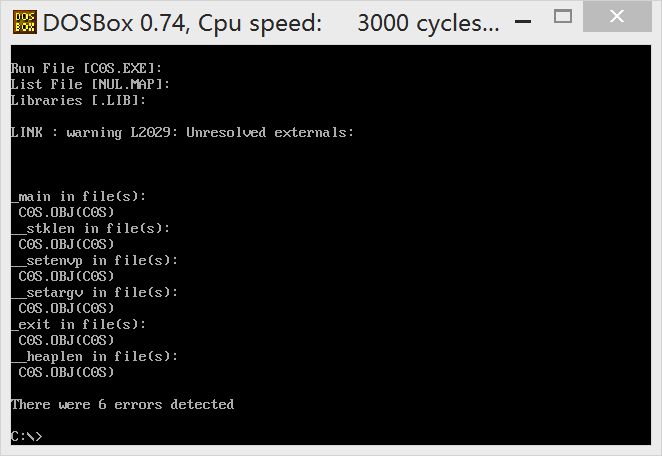
可见两者所在位置不同，还有就是m.exe可以正确返回，但是f.exe不行。

4>.

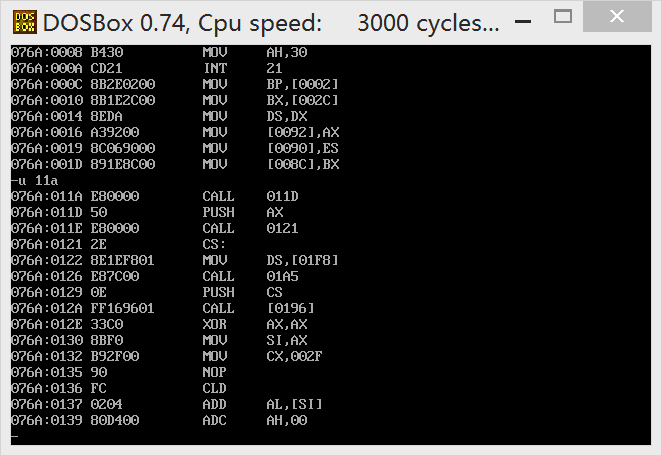
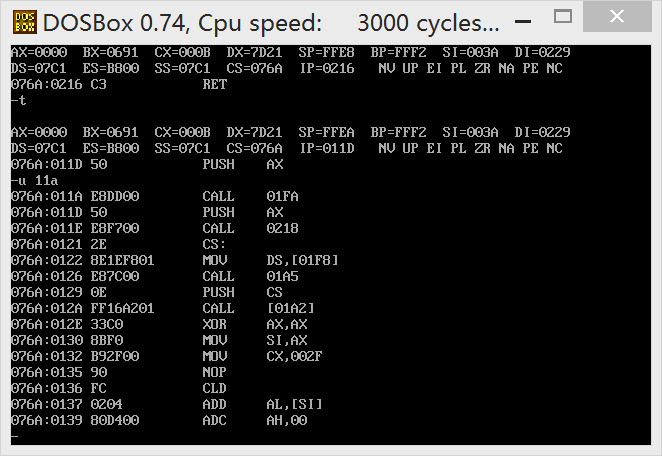




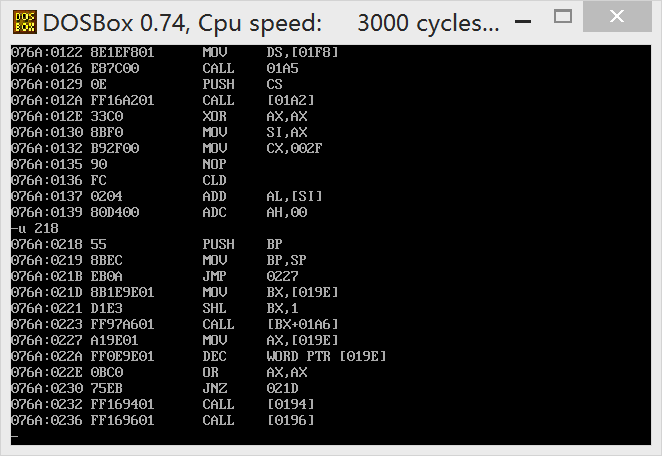
6>.要对c0s.obj进行研究，可以用link.exe对它进行连接，再用debug查看汇编代码。发现虽然link提示多个错误，但是还是生成了c0s.exe文件：



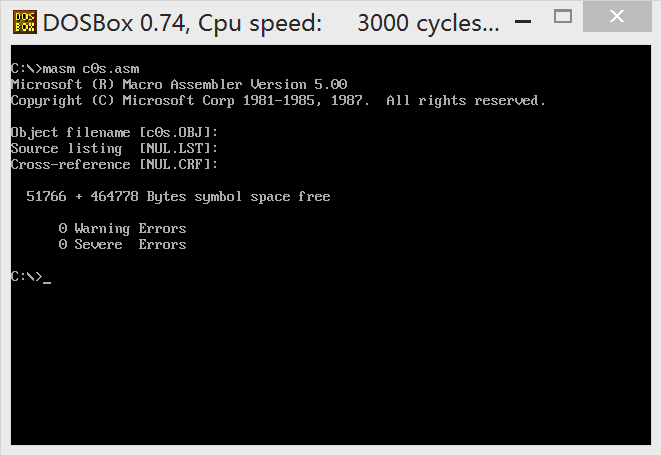
7>.

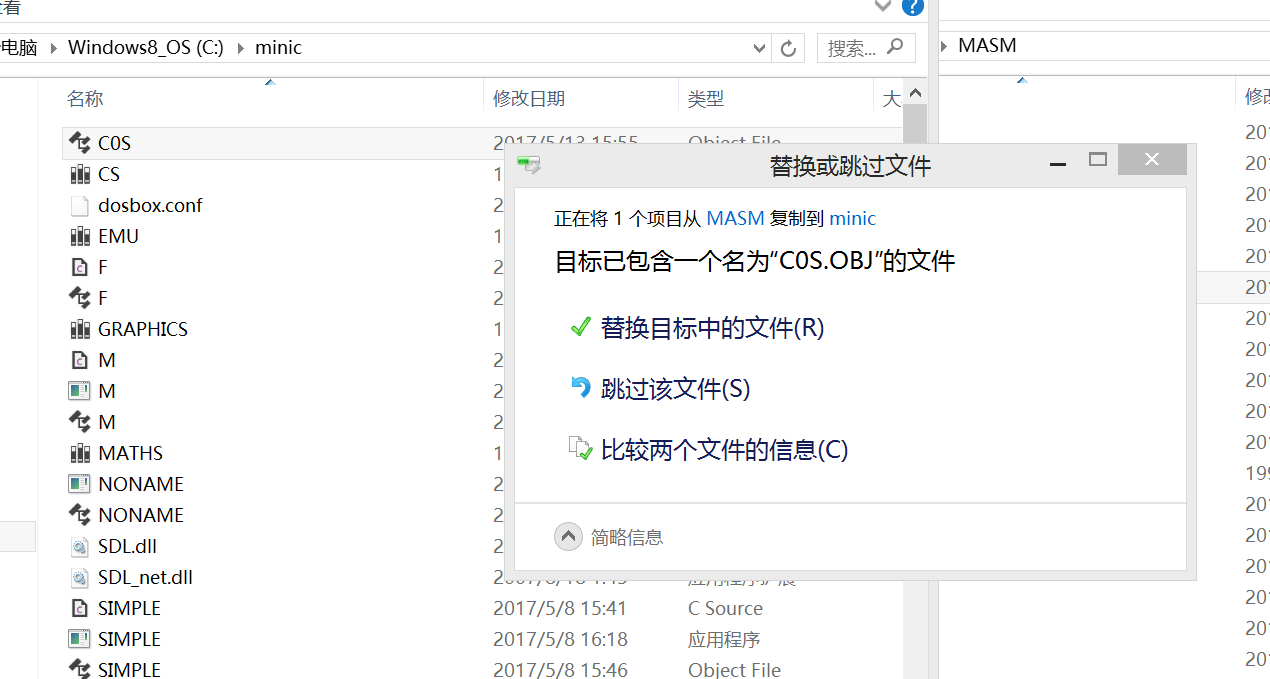


由图可以看出两者的主要区别体现在，在m.exe中第一条call指令，调用自己编写好的m.c中的程序，而c0s.exe中的第一条call指令，则是调用下一条语句，不过这样可以用吗？（没有配套的ret，存疑）；同样的，在m.exe中第二条call指令，调用地址为“076a:0218”中的程序，而c0s.exe中的第二条call指令，则是调用本身所在行的下一条语句，

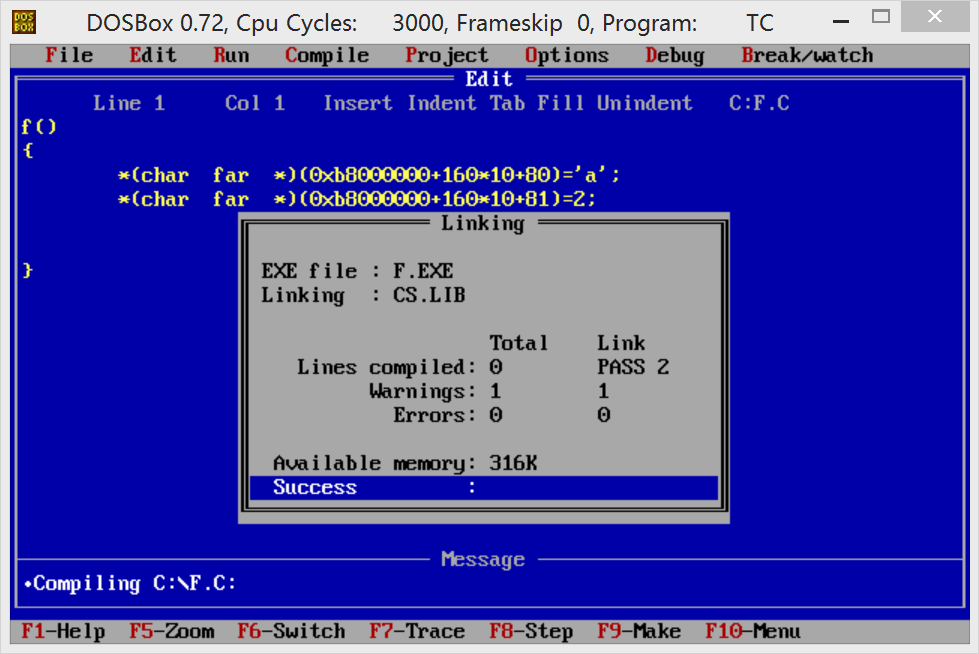


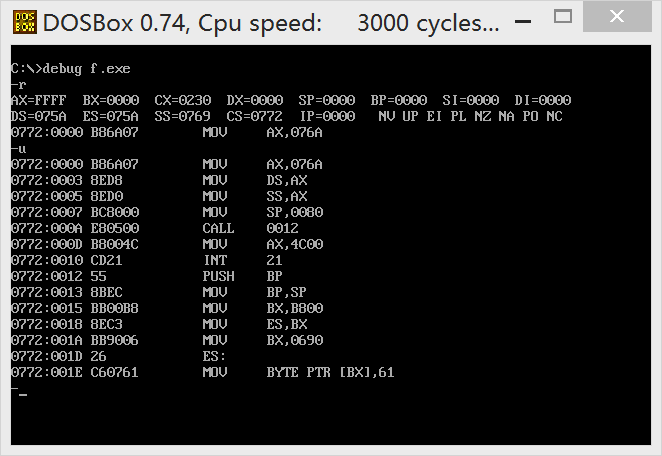
8>.



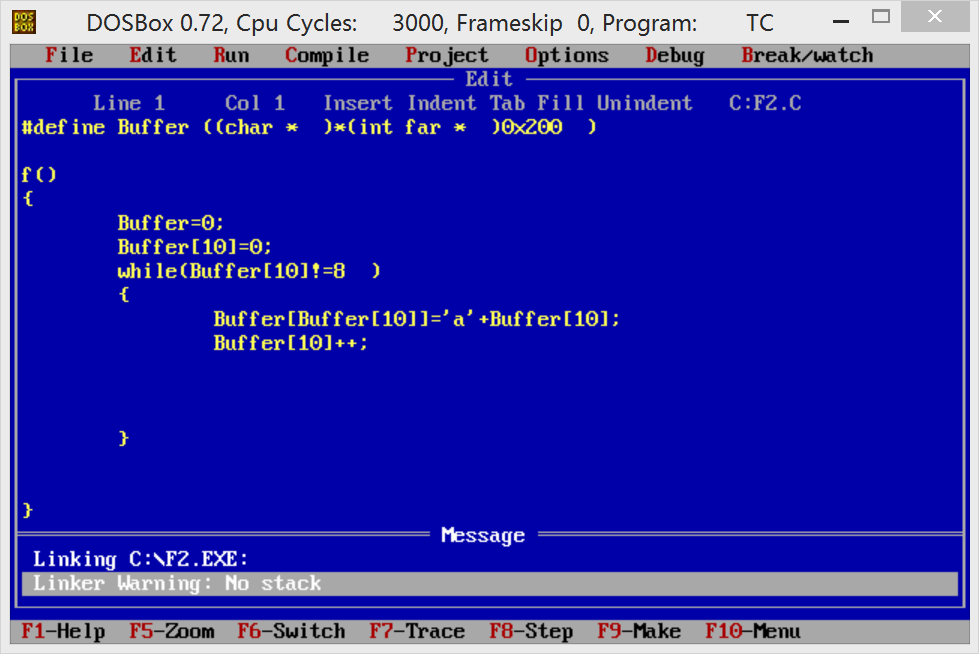


9>.用tc将f.c重新进行编译连接，生成f.exe。这次可以通过连接，可以正确运行。





10>.



1. **已思考研究并已解决问题汇总**

**1>.c0s.obj文件的作用**

c0s.obj文件的作用：在程序开始运行，进行相关初始化，再调用main函数，返回后进行相关的资源释放，环境恢复等工作，再将程序返回。

**2>.可以不用main函数编程吗？**

可以。只不过需要修改相应的原来在程序开始运行时，负责进行相关初始化操作、调用main函数、返回后进行相关的资源释放、环境恢复等工作的文件。在本次研究中用到的是c0s.obj文件，同样的道理，以后的话要学会举一反三。

**3>.文件的大小为何会显示两种结果，大小相差会如此巨大？**

在这里也查看了一下c0s.obj文件的属性：



在连接的过程中是将程序用到的目标文件都连接形成一个可执行文件的过程，大小自然就会变大了。

1. **已思考研究并未解决问题汇总**

**1>.为什么tc不能将原本的f.c连接成功，而link.exe却可以？Link.exe就这么“不挑食”吗？只要是个obj文件就可以生成相应的\*.exe文件吗？为什么tc要想成功连接f.c还需要修改相应的c0s.obj文件，而在command下，直接link就可以了？**

1. **研究感想（心得体会）**

这次的研究实验做的感受就是“又从空中飘回地面”。这回王爽老师让我们一步步朝着“陷阱”走下去。走到底，再想一下。对啊，我们修改的可是c0s.obj文件，这个文件可是tc自带的，算一个编译器商品中比较重要的一小部分了。可是现在呢，我们修改了它，的确是用的我们学习的汇编知识，当然很多是书上的引导，也就是所谓的“陷阱”了。这就有起码好几个启发点：1.这是个编译器，现在你也算对它稍有了解了，继续下去，自己DIY一个编译器，哇，成就感。2.这次我们是修改了c0s.obj文件，因为他让我们不爽了，它竟然必须让我们用main函数编程，我们就不用，偏要做叛逆的小孩。人总要经历叛逆期的，这样看来，程序员当然也要，不然还怎么学会“成长”呢？