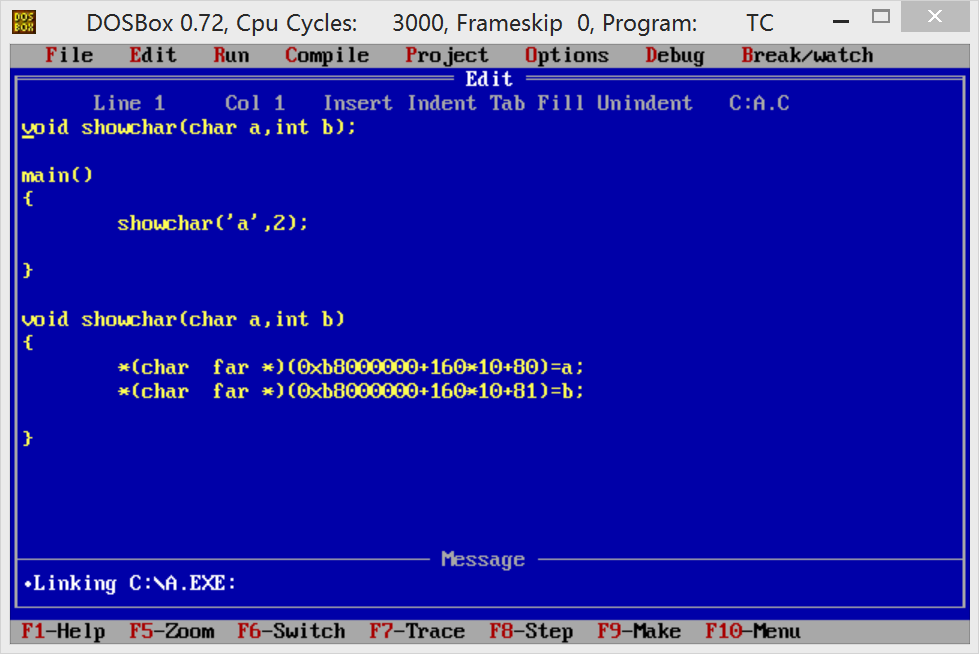
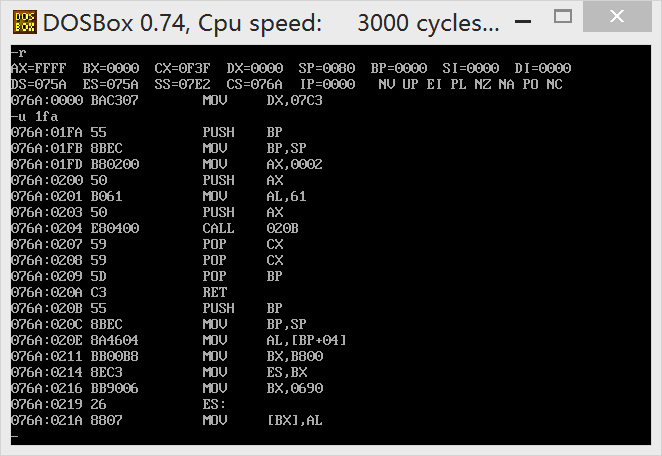
**综合研究五研究报告**

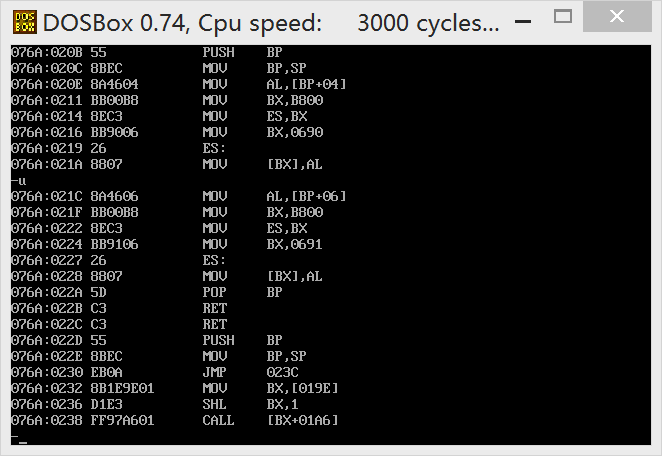
1. **研究过程展示**

1>.

****

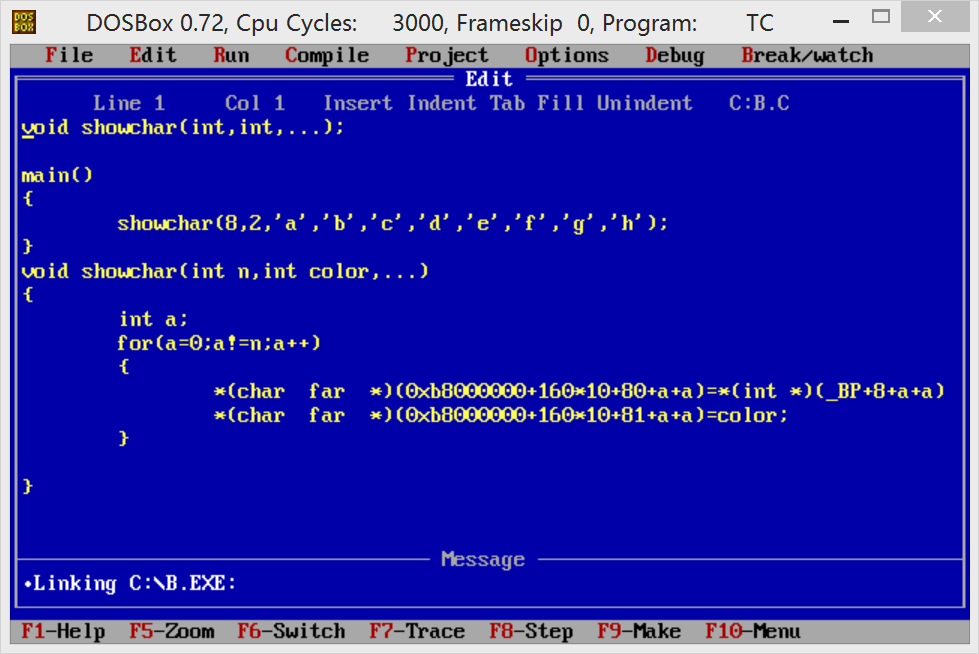
****

如图，main函数要传递两个参数‘a’和2，在汇编代码中是先将2赋给ax，再将ax入栈，然后将a赋给ax,将ax入栈。可见main()给showchar()传递参数是把要传递的值赋给ax，再将ax入栈，且如果有多个要传递的值，是由后往前将参数入栈。

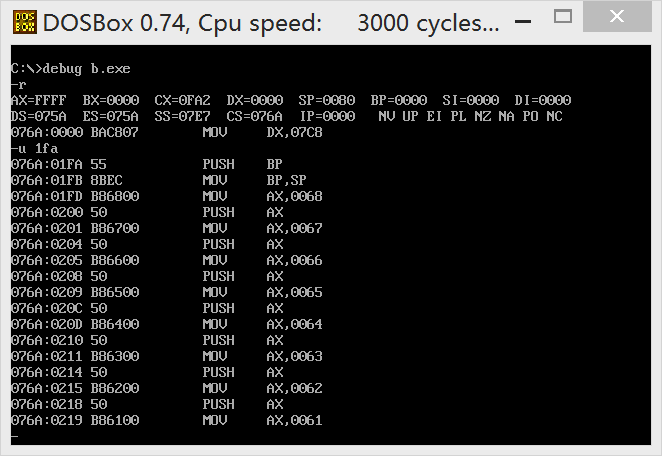
****

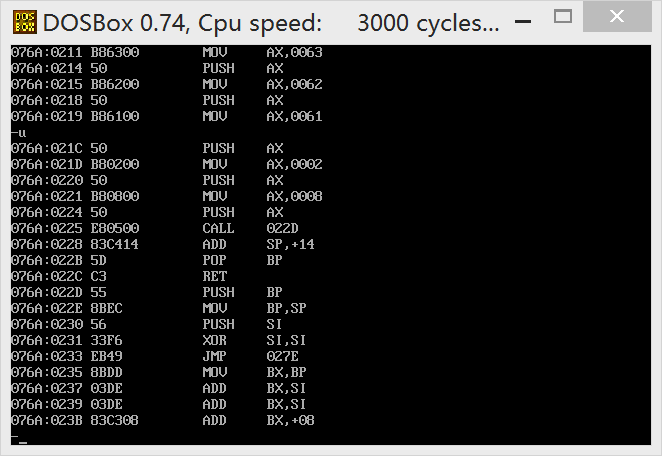
Showchar函数接收参数是将sp赋给bp，然后由bp+4找到栈中存储的参数a，由bp+6找到栈中存储的的参数b，为什么是bp+4和bp+6呢？因为程序在将两个参数入栈后，call指令将showchar的地址入栈占2个字节，在showchar中将bp入栈又占2个字节，所以要由bp+4找到第一个参数的地址。

2>.

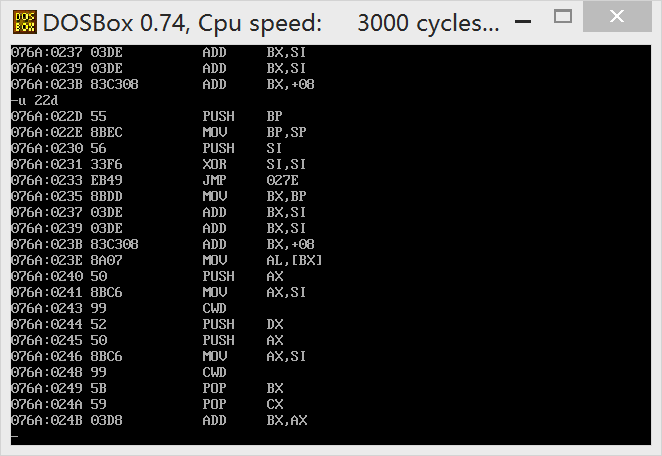
****

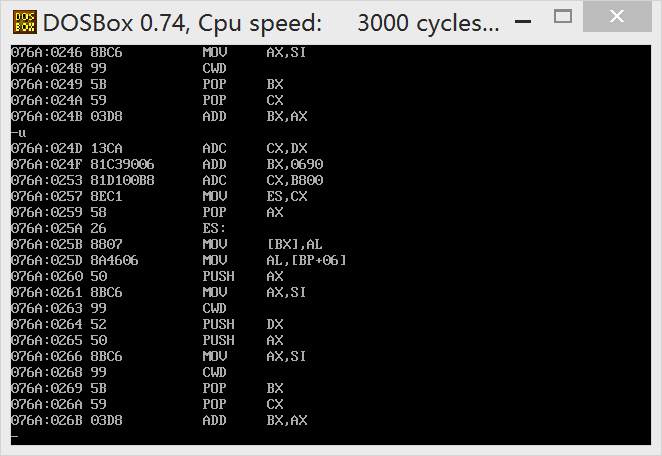
Main()部分：

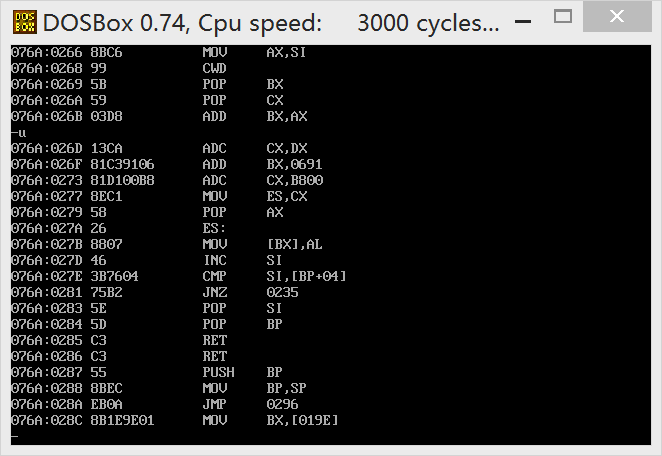
****

****

Showchar函数为：

****

****

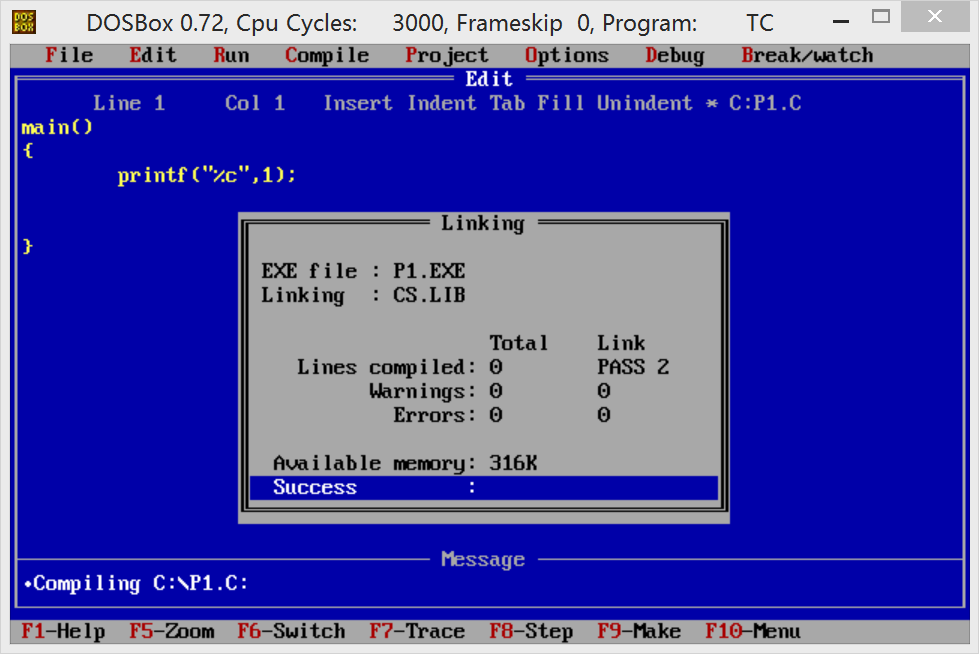
****

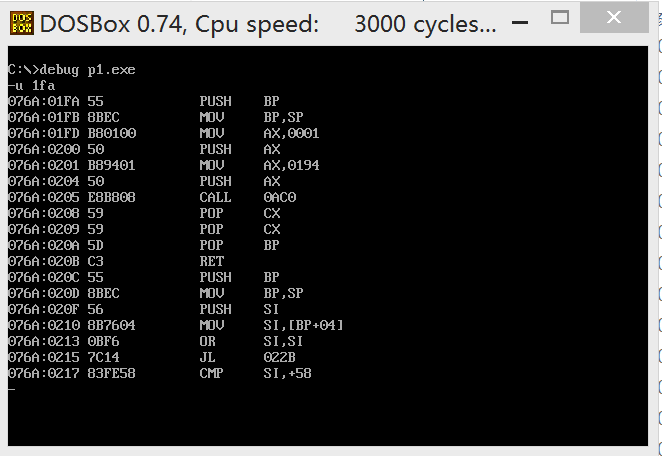
Showchar函数通过参数n来知道要显示多少个字符。然后通过循环来调用寄存器从栈中提取参数。

Printf()部分：

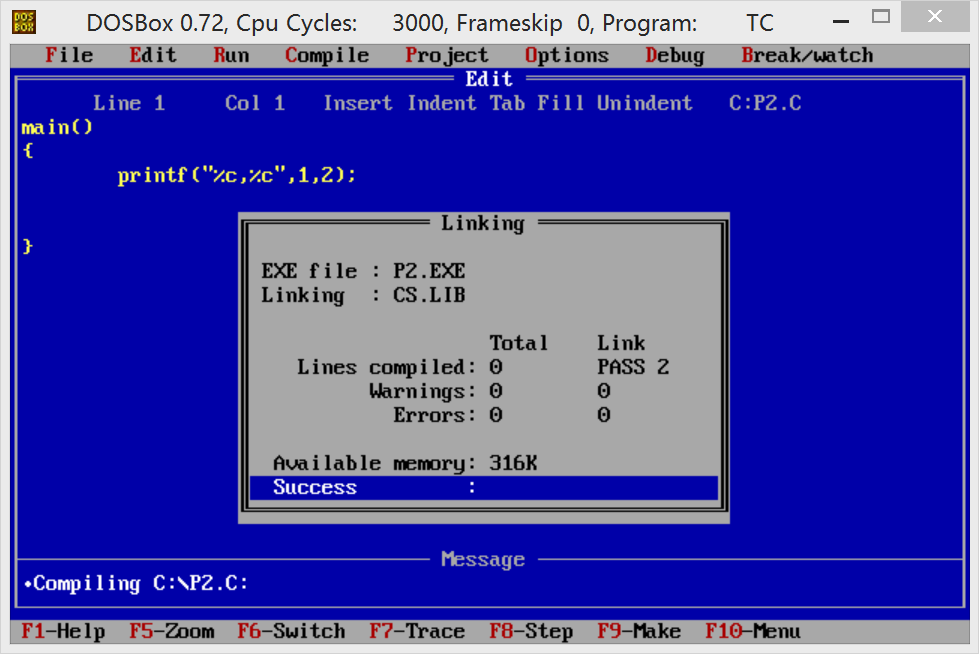
分别打印一个、两个、三个字符，分别反汇编看效果，观察规律。

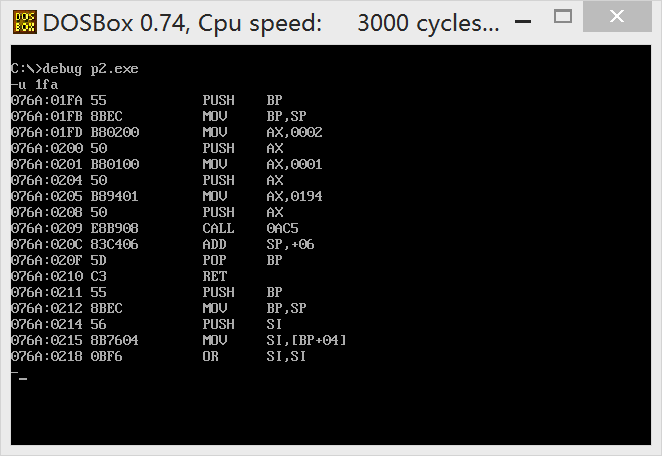
一个字符：



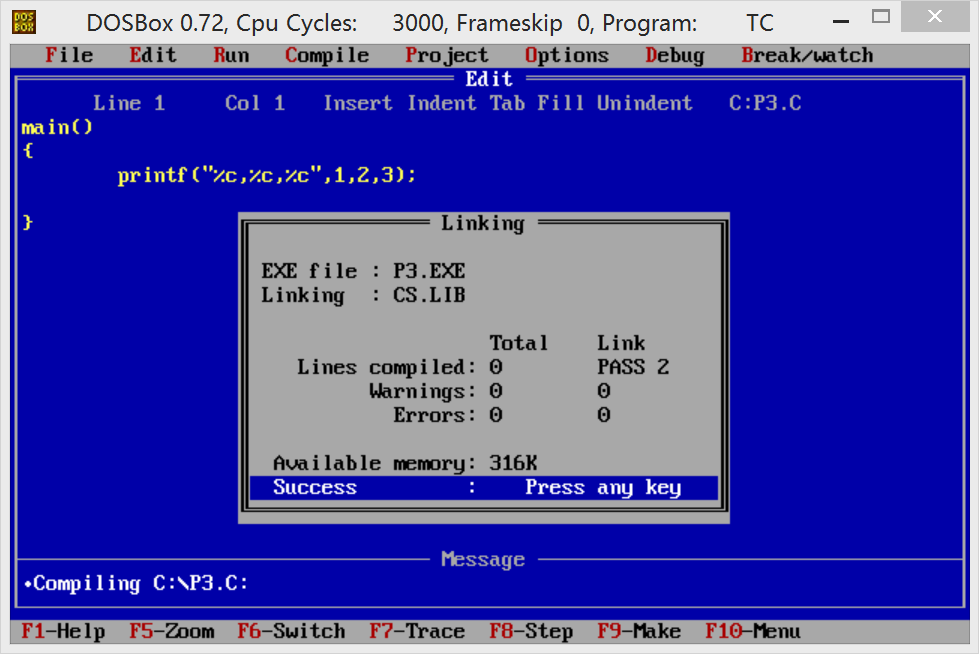
****

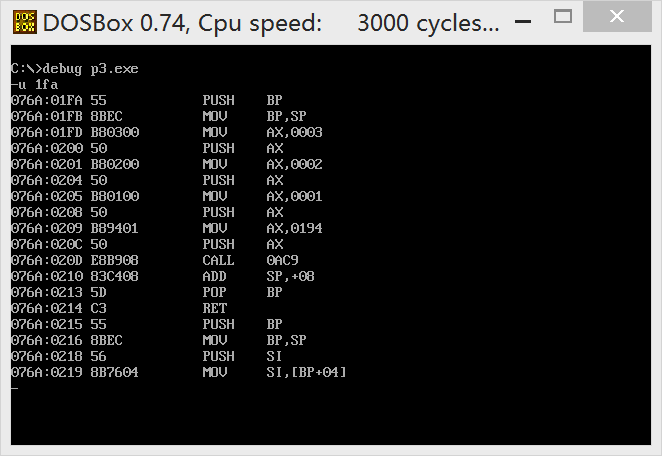
两个字符：





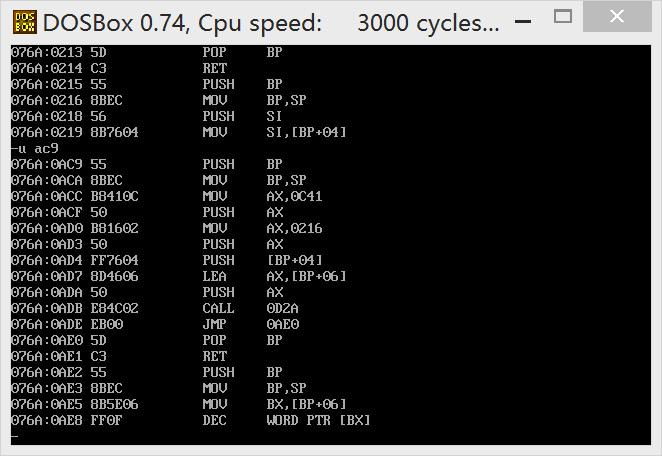
三个字符：

****

****

到此，规律已经比较明显了。每个程序中除了动态的参数入栈规律之外，还有不变的“0194H”和动态的“call 0ac0(0ac5/0ac9)”。

选择其中一个0ac9处代码进行查看，如下：



结合上网搜寻得到的答案，程序将%c和%d等符号放在偏移地址0194处，结尾加0，通过统计该地址处的%个数来确定要输出的字符数量。

由此，可以尝试在showchar()的基础上结合找到的规律进行编写新的printf()。

3>.实现一个简单的printf函数，只需要支持“%c 、%d”即可。

#include<stdio.h>

#include<stdarg.h>

int my\_printf(char \*str,...)

{

int n; //记录返回值

va\_list list;

va\_start(list,str);

n=vprintf(str,list);

va\_end(list);

return n;

}

void main()

{

my\_printf("%s,%d","hello world",10);

}

1. **已思考研究并已解决问题汇总**

**1>.Main函数是如何给showchar传递参数的？showchar是如何接受参数的？**

main函数要传递两个参数‘a’和2，在汇编代码中是先将2赋给ax，再将ax入栈，然后将a赋给ax,将ax入栈。可见main()给showchar()传递参数是把要传递的值赋给ax，再将ax入栈，且如果有多个要传递的值，是由后往前将参数入栈。

Showchar函数接收参数是将sp赋给bp，然后由bp+4找到栈中存储的参数a，由bp+6找到栈中存储的的参数b，为什么是bp+4和bp+6呢？因为程序在将两个参数入栈后，call指令将showchar的地址入栈占2个字节，在showchar中将bp入栈又占2个字节，所以要由bp+4找到第一个参数的地址。

**2>.printf()的参数个数的规律。**

由上述过程可以得出。

1. **已思考研究并未解决问题汇总**

**1>.printf()的规律中，参数个数规律有所发现了，但是参数类型呢？Double？Float？...**

1. **研究感想（心得体会）**

找规律，一件很重要的事情。而善于使用变量，并将变量灵活运用于建造规律公式的过程是十分重要的。以上两者，都是加强编程的必由之路。