**c语言综合研究九报告**

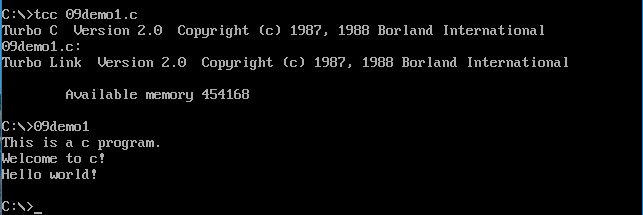
1. **研究过程展示**

在本次研究中，可以看到就是让完善showstr()函数的内容，以使得整个程序的功能可以实现“打印多个字符串，接收不定数量的char\*型参数，最后一个参数可传入0，showstr()通过此可以识别何时停止打印，返回”。

那在这个过程中，参数的数量是不定的，像这样的话，最好找一下规律，看一下参数的值之间有没有什么规律，参数的地址之间有没有什么规律。。。。。。

首先，参数的值，除了最后一个是“0”之外，其余的全部是非常随机的字符串，没有什么规律；那么，先看一下参数的地址之间呢？函数的参数从右往左是顺序入栈的。既然给出了第一个参数，那么就可以得到这个参数的地址，如果将这个地址依次+2，就可以同样依次得到第二个参数的地址，第三个参数的地址。。。。。。而得到了参数的地址，相应的参数的值同样也能得到。最后的“0”作为结束的标志，当然可以随意编写，“1”、“2”。。。。。。“end”等，不过字符串比较用一下strcmp()。

这样就解决了识别接收到的不定数量的char\*型参数，这个不定数量是多少，而且还可以返回了。那么，对于单个char\*型参数，如何将之完整地打印出来，就是最后一个问题了。字符串，都是“\0”结尾的，当然可以把这个作为判断条件。这里我用的是strlen()，嗯，感觉效果还不错。



程序09demo1.c：

void showstr(char \*,...);

char \*p1 = "This is a c program.";

char \*p2 = "Welcome to c!";

char \*p3 = "Hello world!";

main()

{

showstr(p1,p2,p3,0);

}

void showstr(char\* p,...)

{

int\* ptemp;

unsigned int i;

ptemp = (int\*)(&p);

while(1)

{

if( 0 == \*ptemp )

{

break;

}

else

{

for(i = 0; i < strlen(\*ptemp) ; i++)

{

printf("%c",\*( (char\*)(\*ptemp) + i) );

}

printf("\n");

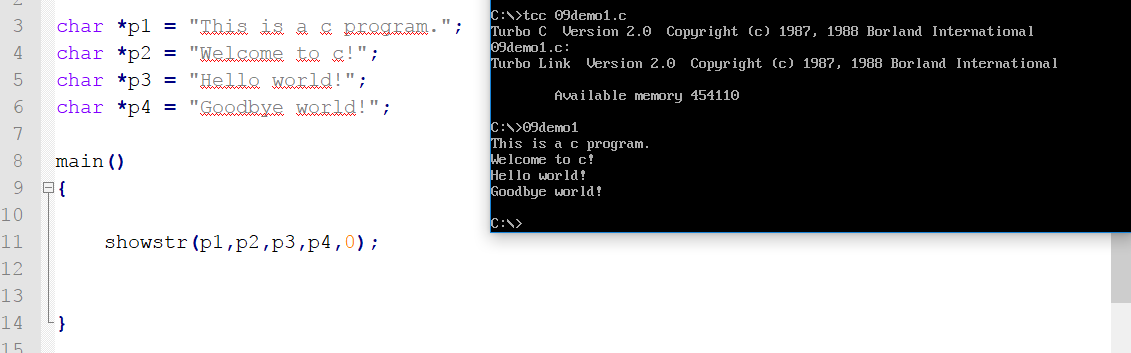
ptemp++;

}

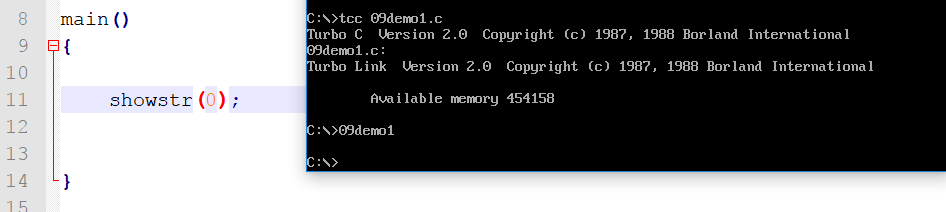
}

}

加一条语句看一下：



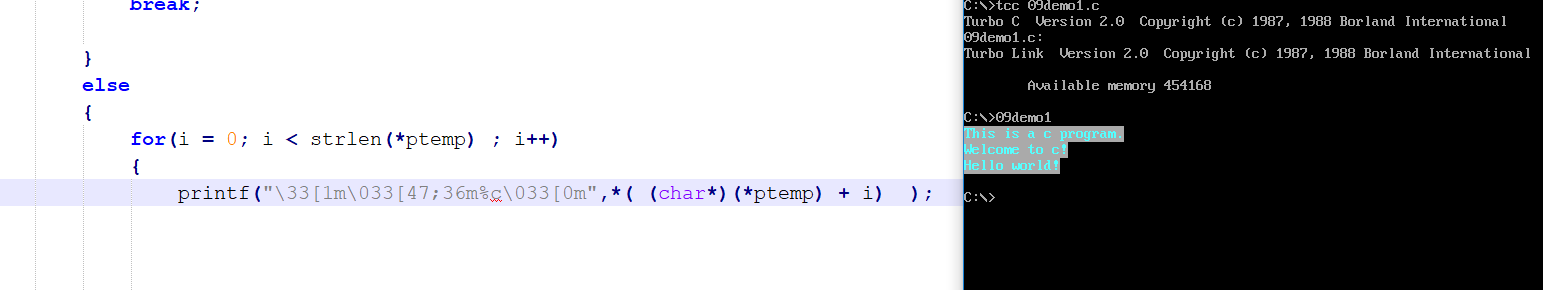
直接调用“showstr(0);”看一下：



1. **已思考研究并已解决问题汇总**
2. 自定义的不定参数函数如何编写？

可以参考本次研究的内容。

1. 本次实验还可以直接将字符串等显示内容通过拷贝到0b800段内存中实现，从这里还可以思考一下，平常也不咋用printf（）函数打印彩色，也就是一般不修改字的背景色，字的颜色，等，这回可以试一下。因为在汇编阶段，这是我们经常玩的。



1. **已思考研究并未解决问题汇总**

有多个地方有多种方法可以去实现。

1. **研究感想（心得体会）**

最近在弄stm32的时候可以再看一下数据结构，脑子有点不想动，唉，太懒了。