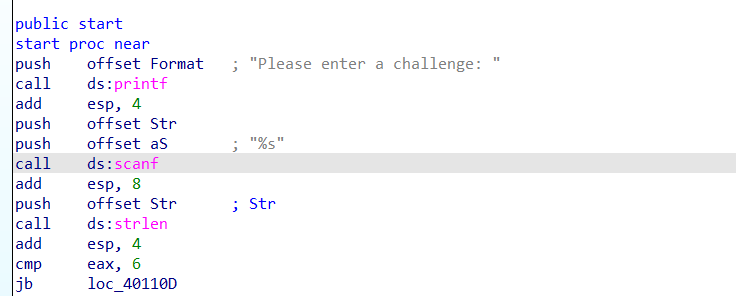
1. 逆向分析二进制代码的计算过程、数据结构、条件判断、分支结构等信息，在实验报告中记录逆向分析的详细过程。

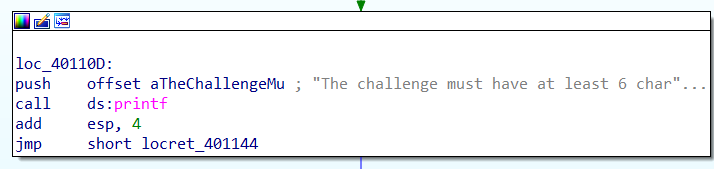
1.



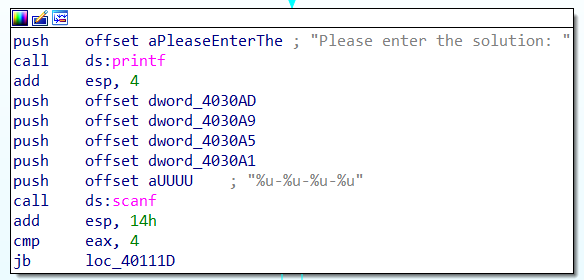
这段代码表示先输入一段字符串“please enter a challenge”，然后输入后面的要输入的字符串， “s格式符，用来输出一个字符串。



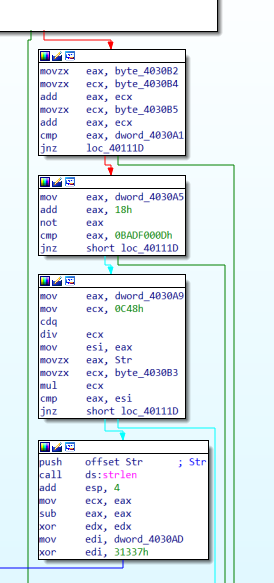
输入str，然后用strlen函数令eax返回字符串的长度



从这段可以看出，输入的字符串长度需要大于等于6



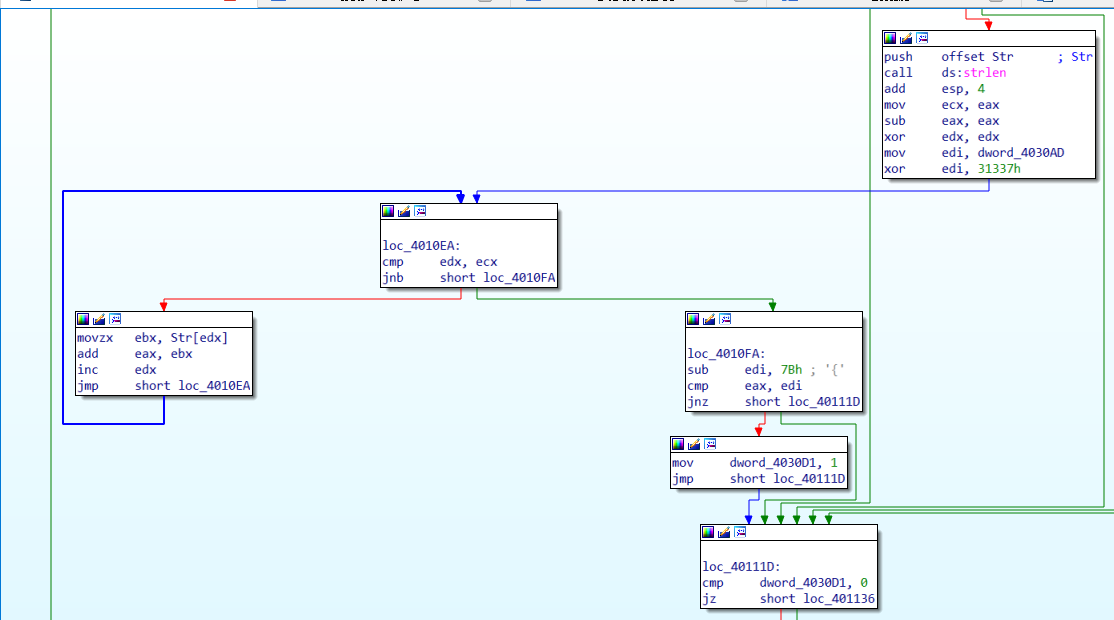
然后输出‘Please enter the solution’,输出结果，输入四个整数，中间以-间隔，%u表示无符号十进制整数，地址分别是4030A1-AD。

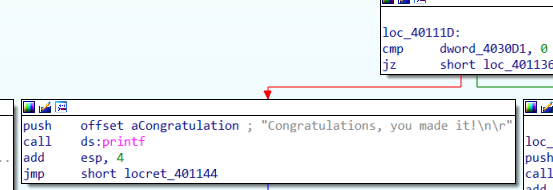


一.接下来是输入四个十进制整数的条件，第一个将一开始输入的字符串的第二个byte位赋给eax，将第四个byte位赋给ecx，然后相加，再加上第五个byte位，最后和我们输入的第一个十进制整数比较，因为我们要走红线才能成功，所以要相等才可以，假设我一开始输入的是000000，则他们ASCII码均为48，然后加起来是144，所以我们第一个数输入144。

二.第二个输入十进制整数的条件是它加上18h后取反等于BADF000Dh即可，我们反向计算，先让BADF000Dh取反，是4520FFF2h，然后再减去18h，得到4520FFDAh，然后转为十进制，即为1159790554，这个是固定的，与我们输入的字符串无关。

三．最后比较的是eax和ecx，我们需要让他们两个相等，他最后将str，即我们输入的第一个字符串的字符给eax，然后把输入的第三个字符给ecx，让他们两个相乘，得到eax，假定我们输入000000，所以得到ASCII码即是48\*48，得到2304，然后esi是我们输入的第三个十进制整数除以C48h（十进制3144）得到的商，所以我们输入的第三个十进制整数应该是2304\*3144=7243776.

四. 



如果想要成功，需要走那条红线，然后我们的dword\_4030D1就要置为1，然后就需要过上面的mov dword\_4030D1 , 1，然后就需要实现上面的那条红线，然后我们再从第四个十进制整数的输入开始看起，将我们一开始输入的字符串长度给ecx，eax置为0，然后将edx与edx进行异或，就会全置为0，然后把我们输入的第四个十进制整数赋给edi，让edi与31337h进行异或，然后让edx与ecx进行比较，当edx的值从0加到ecx的值（即一开始输入的字符串长度）一样时，实现跳转，此前执行的操作是将输入的字符串每个字符的ASCII码进行求和存在eax里，假定我们输入的字符串是000000，则此时eax保存的就是48\*6=288，然后edi（此时存的是第四个输入的十进制整数与31337h的异或结果）减去78h，再让他与eax相等，则288对应十六进制120，先加上7Bh，为19Bh，即0011—0001—0011—0011—0111与第四个输入的数异或结果为0001—1001—1011，所以可得出第四个输入的数是0011—0001—0010—1010—1111，转化成十进制就是201388.

五.最后我们可以得出我们想要的一组结果，challenge：000000

Solution：144-1159790554-7243776-201388.

1. 运行程序，根据提示输入字符串和逆向挑战的结果，获得“Congratulations，you made it！”输出，将成功的截图复制到实验报告中。

