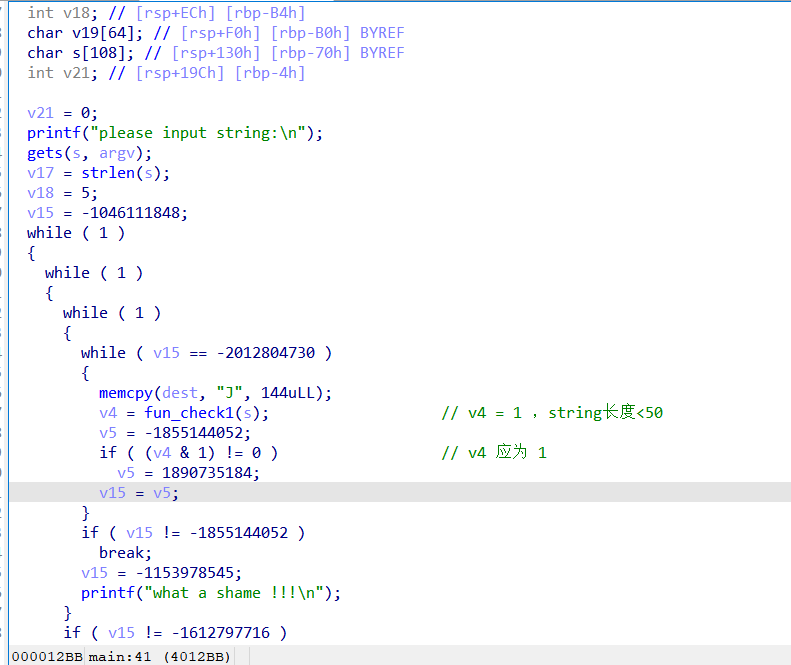
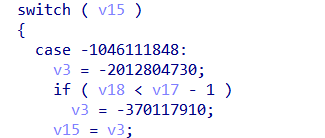
Die查出为64位elf，丢到ida：

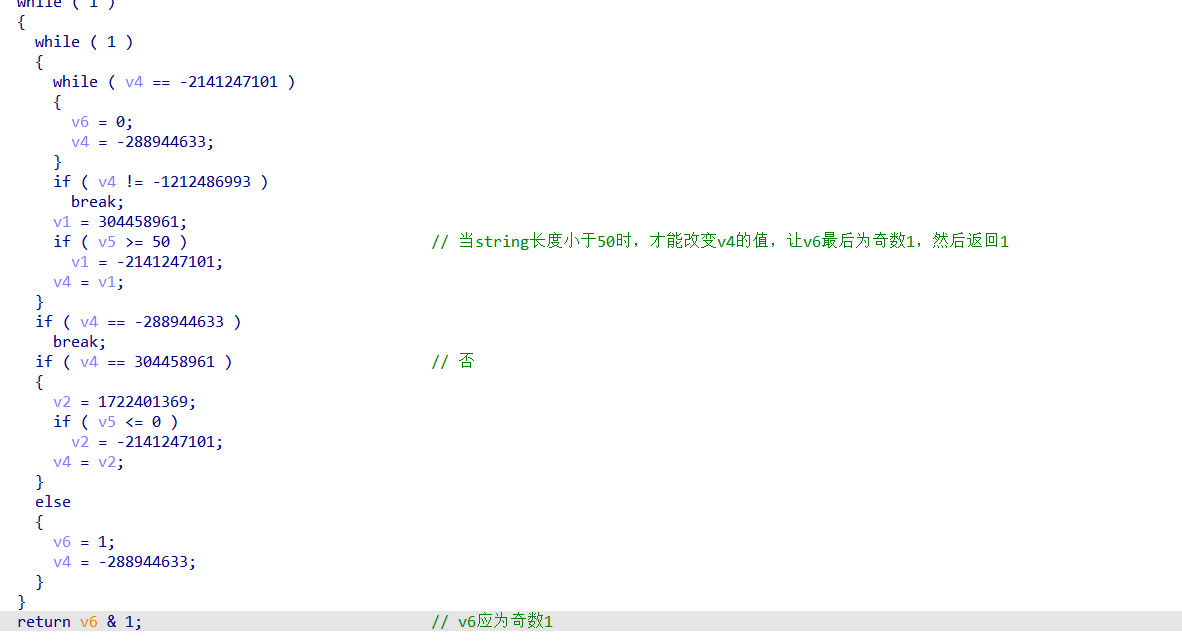
很长一段代码，慢慢看



先输入以几个数字试试，发现每次都会调到what a shame !!!，但是v15并不等于-2012804730

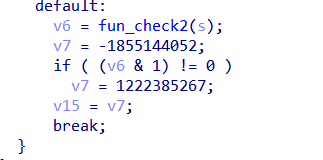


后来在后面switch语句发现一开始时的v15=-1046111848刚好可以进行进入第一种情况，并为v15赋值为-2012804730

然后我们必须跳过what a shame !!!，所以第一个break必须成立，v15必须改变，因此第一个if语句必须进入，也就是说v4为奇数，所以我们必须让fun\_check1的返回值为奇数，进入这个看看：  


理解后发现，他是在限定我们的string的长度小于50

记住现在的v15 = 1890735184，然后继续往后面看，发现，每一个可以与他对应起来，最后只能进入default的情况

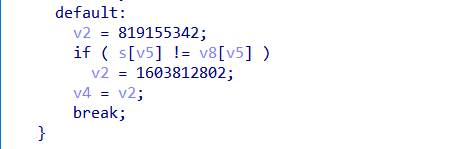


这个时候发现又有check2，还是2，刚刚是check1，所以现在一个可以大致猜测，这个string经过多次check的限制就可以得出

继续看这个default，发现和刚刚差不多，v我们需要让这个check2函数返回1，进去看看：

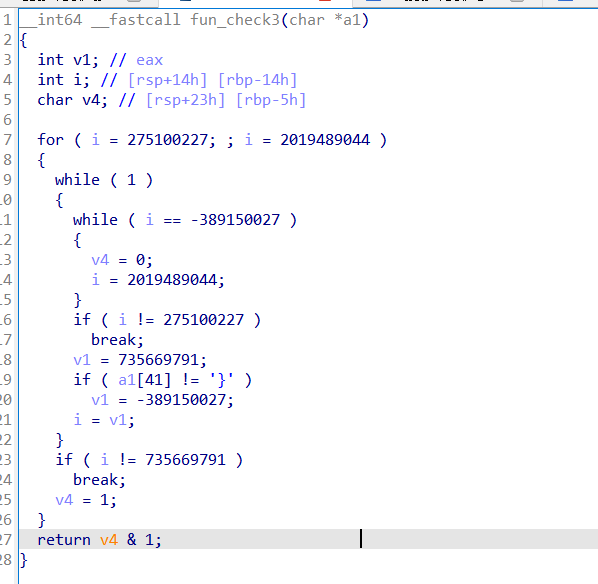


进去就把string的开头告诉我们了，和一般的flag答案一样



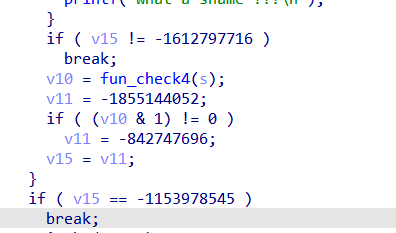
再结合上面的一些代码，可以发现，这是在限制输入的前五个字符为flag{

然后按照一个一个的规律check，直接去看check3

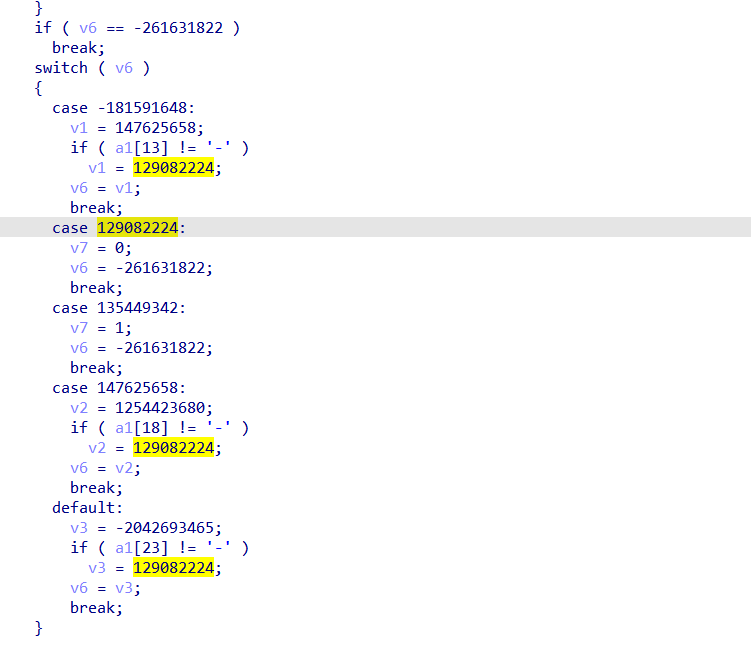


这个函数就是限制下标为41的字符为}，string的长度为42，且是flag{}的形式，记住现在的v15=-1612797716

然后再继续看代码，不出意外地来到了check4：

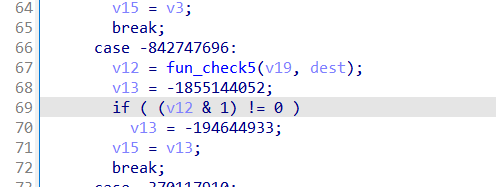


进去看看



还是和之前一样需要返回1，在这里就要让v7=1，v2就不能等于129082224，所以这三个if语句就不能为逻辑真，所以这个其实是在限制string字符串下标为13,18,23的字符为-

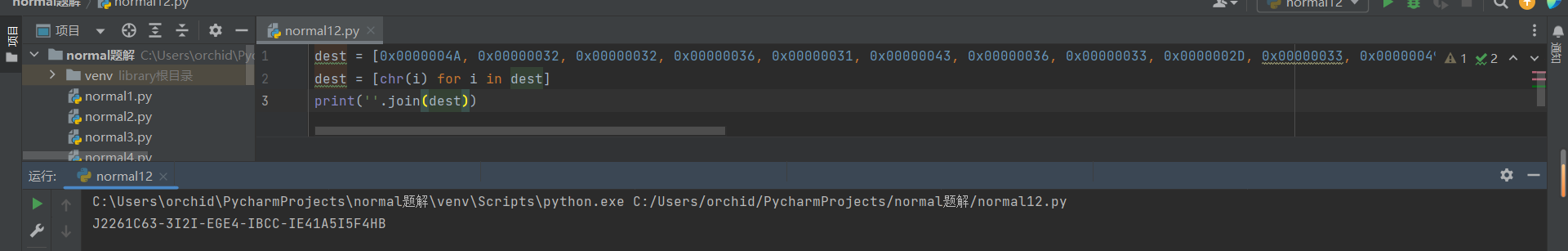
然后返回main函数，记住现在v15=-842747696



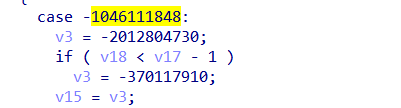
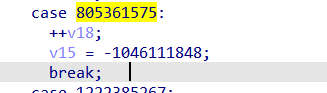
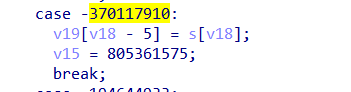
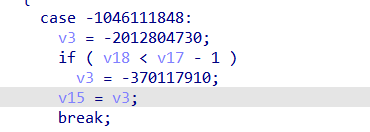
最后到了check5，明确目标，是为了让check返回1，进去看看：

Wc，好复杂！

首先先从上面的代码获得dest = J2261C63-3I2I-EGE4-IBCC-IE41A5I5F4HB，再去找v19



然后再看看main函数，发现条件1还有一些联动：



这其实就是把v19变成我们的输入的flag{}中{}里面的字符串

然后就可以分析check5了：

（截图不放，太长了）

虽然是大致按照数据特点看的，有些猜测，然后看了一下wp，确定这个其实是表示：

当输入字符为小写字母a-z（97-122）时，将其的ascii码转换成的十进制数字减48变成数字0-9（48-57）和A-J（65-74）（这个A-J不会产生），输入的为数字0-9（48-57）时加17变成大写字母A-J（65-74），输入为-时不变

然后就可以写脚本了：

