## 题目分析

程序首先对输入的奇偶位进行了不同的异或运算，然后再进行编码表变换的BASE64加密，这个BASE64加密比较有意思。

一进入sub\_A84发现就很少的代码，然后里面的sub\_7FA有64个case分支，这64个case中的字符依次拼接起来就是变形的编码表了。

当发现是程序是变形的base64加密后就好解决了，我们将用变形的base64编码表加密得到的密文转换为正常编码表得到的base64密文，然后就行base64解密，再对奇偶位进行异或运算即可。

脚本：

base = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/"  
diy\_base = "nopqrstuvwxyzabcdefghijklm0123456789ABCDEFGHIJKL+/MNOPQRSTUVWXYZ"  
diy\_cipher = "OBufaa21Td86rWS8Wob8iGhZYocbr5vxZfcCoWv3"  
  
#将通过变形的base64编码表加密得到的密文转换为base64编码表加密的密文，再进行base64解密  
cipher = ""  
for i in range(len(diy\_cipher)):  
 cipher += base[diy\_base.find(diy\_cipher[i])]  
cipher = cipher.decode("base64")  
  
#进行奇偶位异或运算  
flag = ""  
for i in range(len(cipher)):  
 if(i & 1):  
 flag += chr(ord(cipher[i]) ^ 0x60)  
 else:  
 flag += chr(ord(cipher[i]) ^ 0x91)  
print(flag)