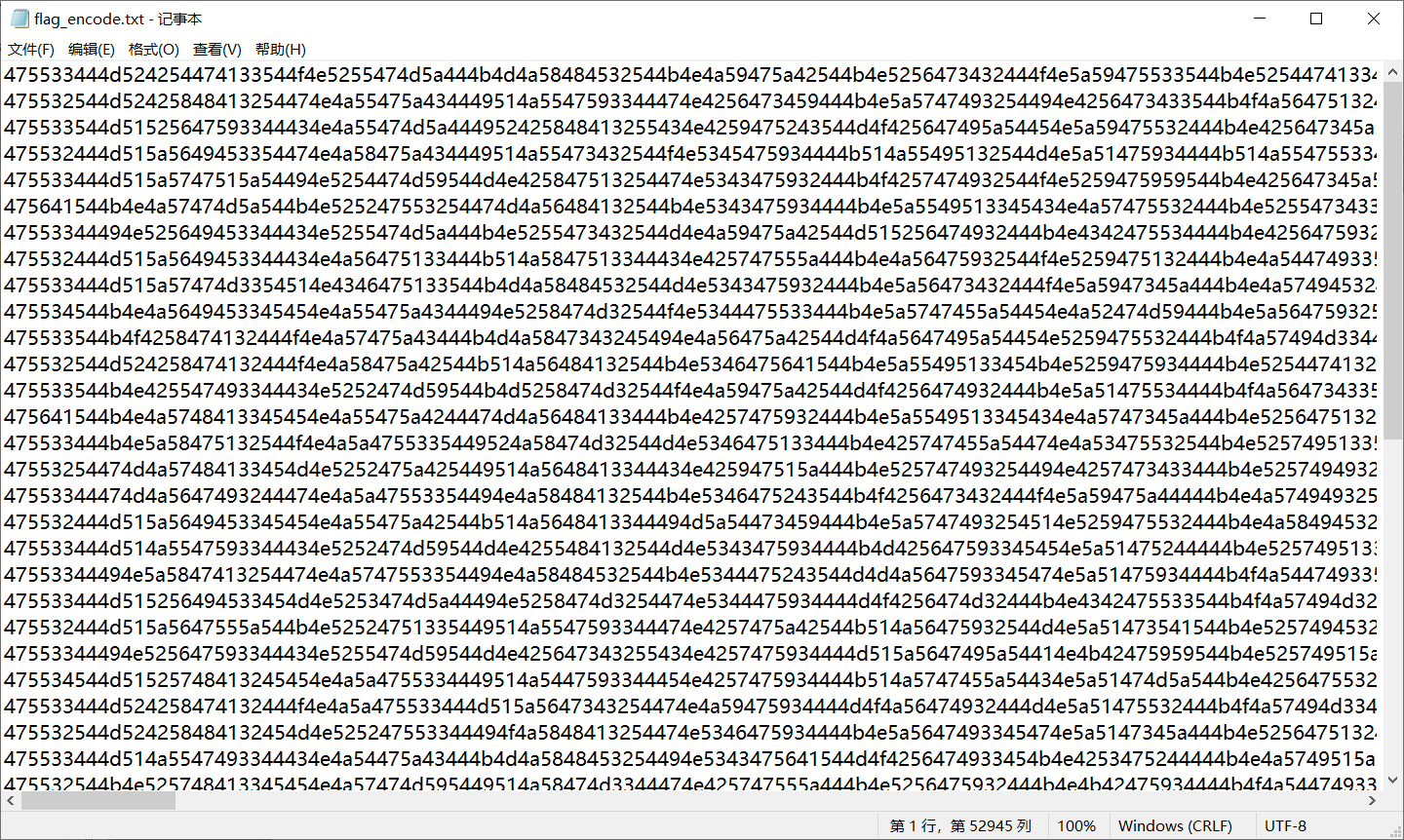
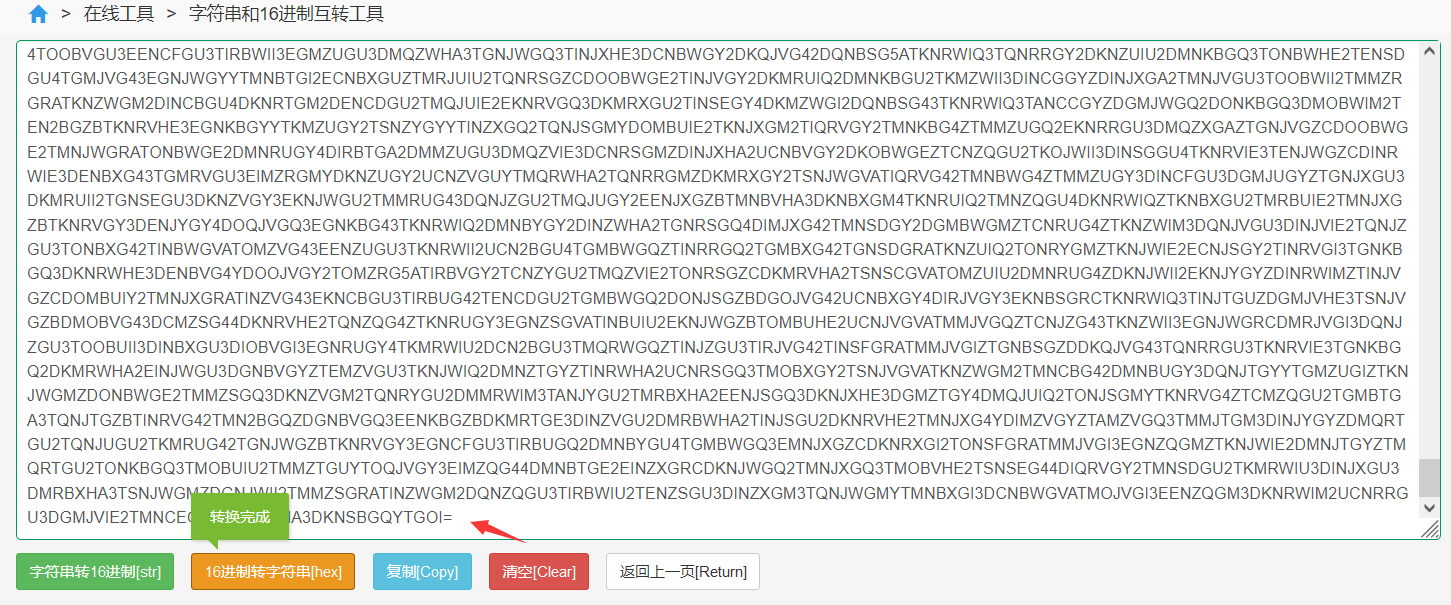
# Base

下载附件，是一个文本，判断可能是16进制



尝试解码以后，观察到有=，考虑可能是base的解码



尝试用base64解码不了，考虑可能是base32，base16或者混合多次解码

写脚本：

import base64

with open(r"flag\_encode.txt", 'r', encoding='utf-8') as f:

for a in f:

while 1:

try:

a = base64.b64decode(a).decode("utf-8")

except:

pass

try:

a = base64.b32decode(a).decode("utf-8")

except:

pass

try:

a = base64.b16decode(a).decode('utf-8')

except:

pass

if "{" in a:

print(a)

break

得到flag

flag{U\_5h0u1d\_Us3\_T00l5}

tip:这道题的flag经过20次base64加密，一次base16加密，一次base32加密

# ks

下载附件，一个文档，密文为synt{21np-13579o-24h-6810t-llqf12-3nop}

看起来应该是凯撒类型的密码

写脚本：

flag = 'synt{21np-13579o-24h-6810t-llqf12-3nop}'

table = ''

def rot(table):

for i in flag:

if(i>='a' and i<='z'):

a = chr(ord(i) + 13)

if (a > 'z'):

a = chr(ord(a) - 26)

table+=a

else:

table+=a

elif(i>='A' and i<='Z'):

b = chr(ord(i) + 13)

table+=b

if(b>'Z'):

b = chr(ord(b) - 26)

table+=b

else:

table+=b

else:

table+=i

print(table)

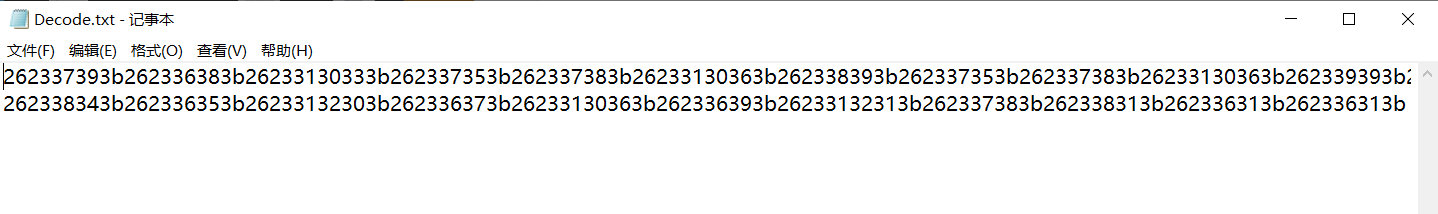
rot(table)

直接暴力破解得到flag

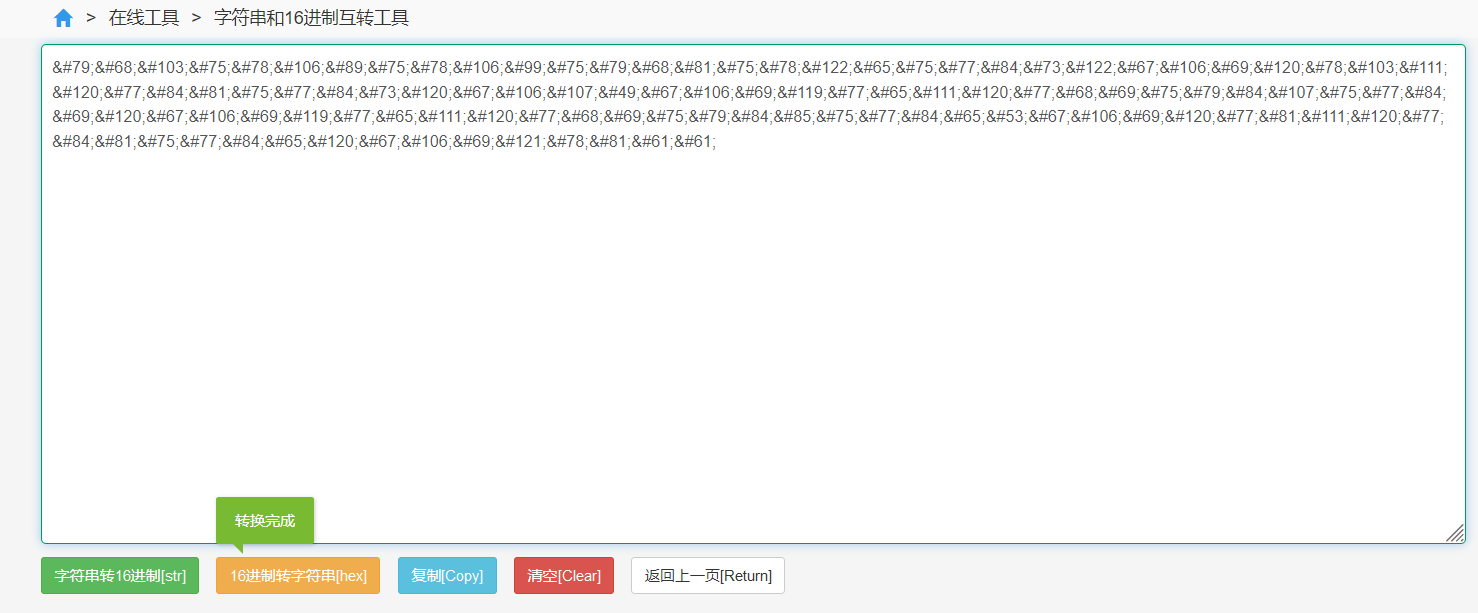
flag{21ac-13579b-24u-6810g-yyds12-3abc}

# Decode

下载附件，一个文档，密文看上去像16进制



经过解码，判断是Unicode编码



Unicode解码，得到的密码判断是base类编码



经过一次base64解码，密码变成ascii码的形式

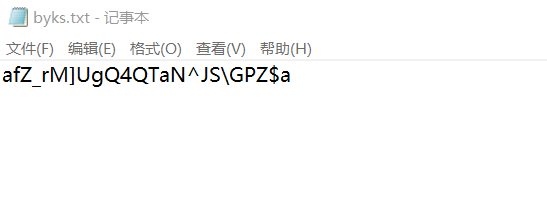


经过ascii码解码得到flag

XBCTF{try\_decode\_more}

# byks

下载附件，文档里的密文



发现密文前五位ascii与格式相差为5，6，7，8，9。猜测应该是变异凯撒

写脚本：

m='afZ\_rM]UgQ4QTaN^JS\GPZ$a'

a=5

for i in range(0,len(m)):

print(chr(ord(m[i])+a),end='')

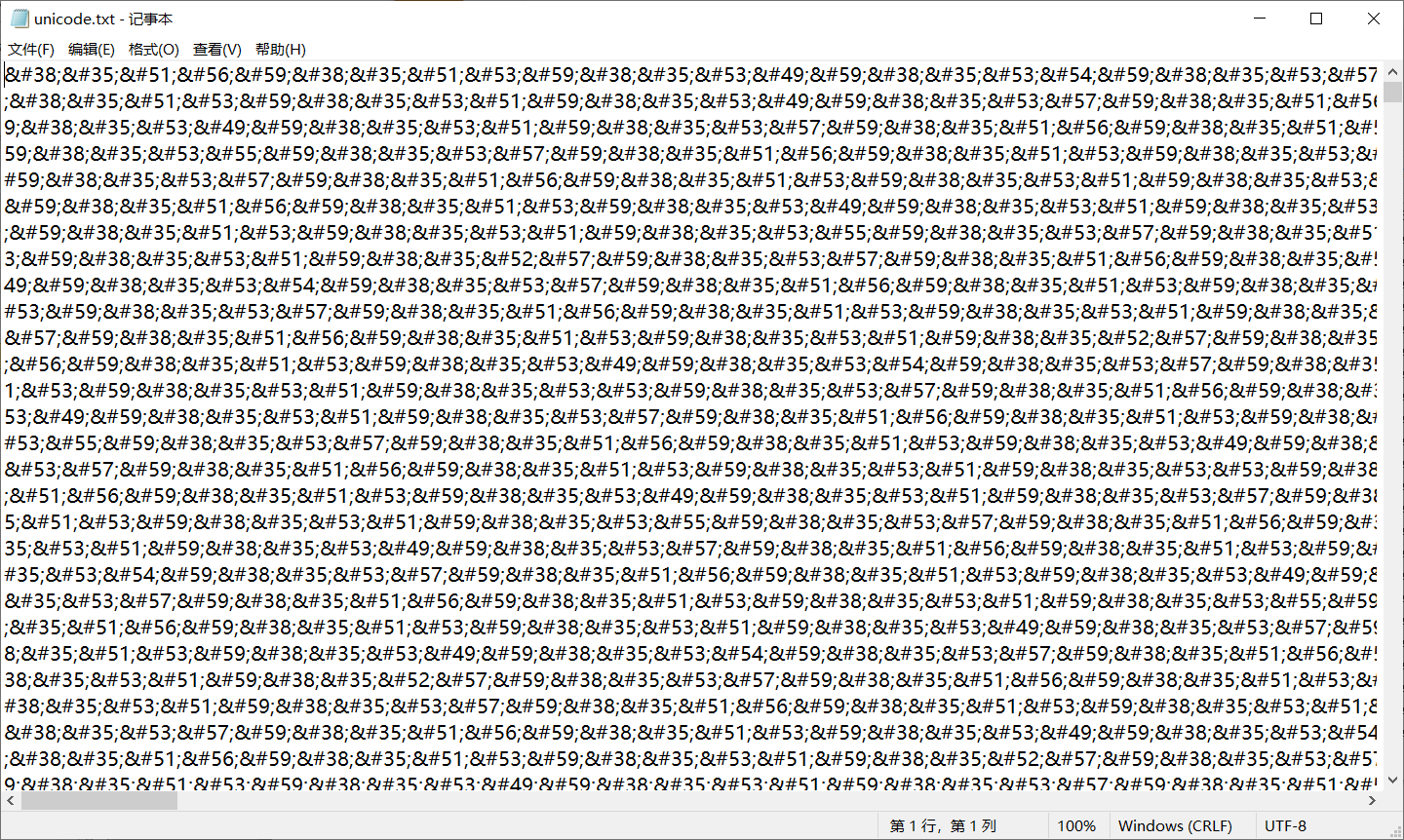
a+=1

得到flag

flag{What\_Caesar\_is\_it?}

# Unicode

下载附件，查看文档是很长的密码，目测是Unicode编码



写脚本：

import html

with open(r"123.txt", 'r', encoding='utf-8') as f:

for a in f:

t=10

while t:

print(html.unescape(a))

t=t-1

a=html.unescape(a)

得到flag

flag{1m\_t0o\_t1r3d}