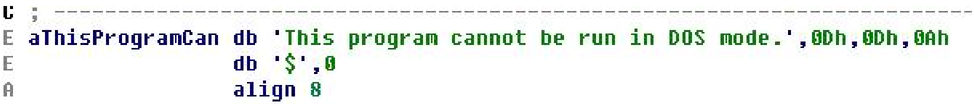
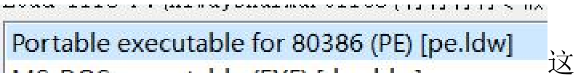
---------------------------------你看看，逆向多简单！ POINT: 10--

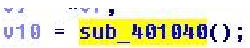
丢进 32 位的 ida 用 DOS 模式打开发现这样一句话



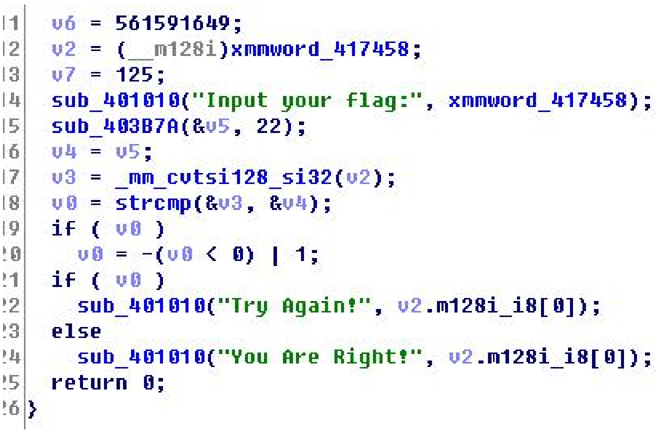
```
C ; ---------------------------------------------------------------------------
E aThisProgramCan db 'This program cannot be run in DOS mode.',0Dh,0Dh,0Ah
E                 db '$',0
A                 align 8
```



所以换 Portable executable for 80386 (PE) [pe.ldw] 这个打开就正常了

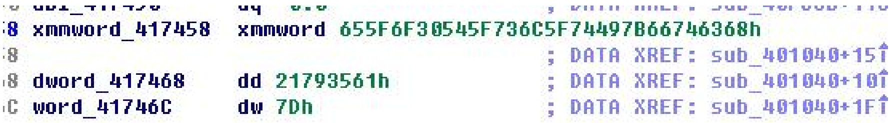


F5 后挨个函数点进去[恩我就是这么蠢]发现这个函数 `v10 = sub_401040();` 里面点开是这样的



```
  v6 = 561591649;
  v2 = (__m128i)xmmword_417458;
  v7 = 125;
  sub_401010("Input your flag:", xmmword_417458);
  sub_403B7A(&v5, 22);
  v4 = v5;
  v3 = _mm_cvtsi128_si32(v2);
  v0 = strcmp(&v3, &v4);
  if ( v0 )
    v0 = -(v0 < 0) | 1;
  if ( v0 )
    sub_401010("Try Again!", v2.m128i_i8[0]);
  else
    sub_401010("You Are Right!", v2.m128i_i8[0]);
  return 0;
}
```

于是返回查看汇编代码找到了



```
8 xmmword_417458   xmmword 655F6F30545F736C5F74497B66746368h
8                                          ; DATA XREF: sub_401040+15↑
8 dword_417468     dd 21793561h            ; DATA XREF: sub_401040+10↑
C word_41746C      dw 7Dh                  ; DATA XREF: sub_401040+1F↑
```

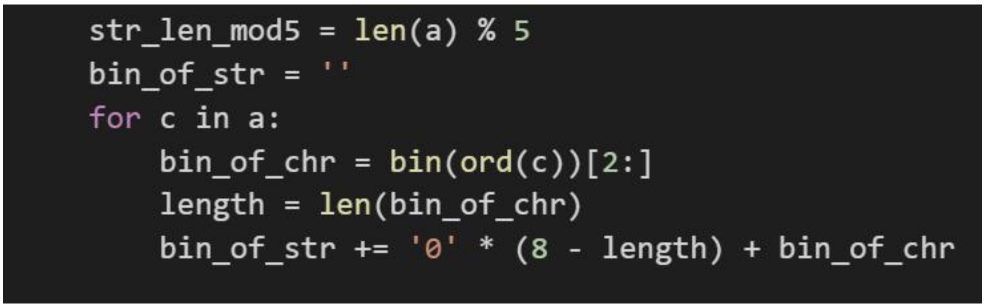
得到

7D 21 79 35 61 65 5F 6F 30 54 5F 73 6C 5F 74 49 7B 66 74 63 68

转换成十进制按照 ascii 码比对出}!y5ae_o0T_sl_tI{ftch，倒序就是 hctf{It_ls_T0o_ea5y!}

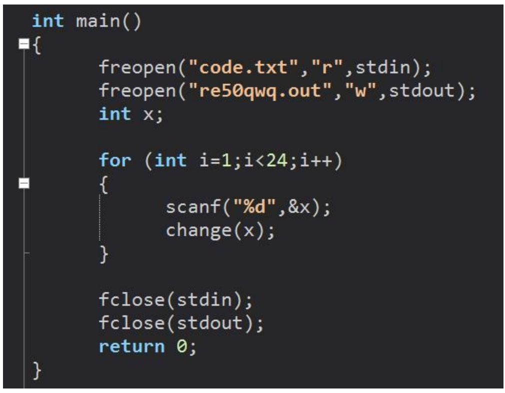
--------------------------------------蛤，这是啥？ POINT: 50---

没后缀，改 txt 也打不开，改成.pyc 后丢进反编译工具里面打开发现是 python 的代码。
并不会 py，GG，一条一条语句地查语法

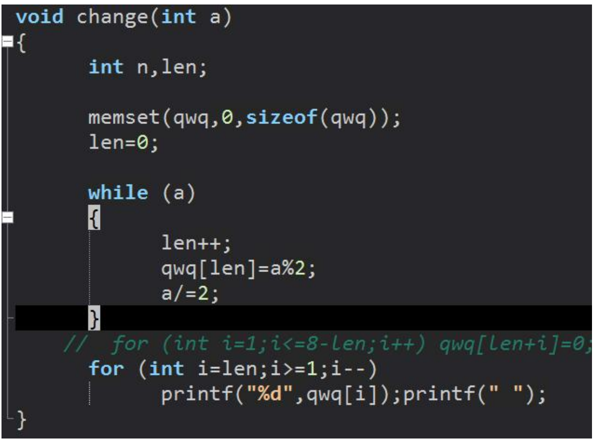


```
    str_len_mod5 = len(a) % 5
    bin_of_str = ''
    for c in a:
        bin_of_chr = bin(ord(c))[2:]
        length = len(bin_of_chr)
        bin_of_str += '0' * (8 - length) + bin_of_chr
```

于是大概搞懂是个啥了，就是把 flag 按字符转换成二进制的，每个二进制串不足 8 位补齐八位，根据字符串长度在后面补’0’和’=’，再每五位分组转成字符。所以就很好办了



```
int main()
{
        freopen("code.txt","r",stdin);
        freopen("re50qwq.out","w",stdout);
        int x;

        for (int i=1;i<24;i++)
        {
                scanf("%d",&x);
                change(x);
        }

        fclose(stdin);
        fclose(stdout);
        return 0;
}
```



```
void change(int a)
{
        int n,len;

        memset(qwq,0,sizeof(qwq));
        len=0;

        while (a)
        {
                len++;
                qwq[len]=a%2;
                a/=2;
        }
    //  for (int i=1;i<=8-len;i++) qwq[len+i]=0;
        for (int i=len;i>=1;i--)
                printf("%d",qwq[i]);printf(" ");
}
```

01101 00001 10001 10111 01000 11001 10011 11011 01100 01001 10000 10111 00110 11001 01010 11111 00110 01100 11001 00010 00010 11111 01000

然后

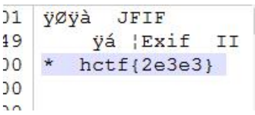
01101000 01100011 01110100 01100110 01111011 01100010 01100001 01110011 01100101 01011111 00110011 00110010 00100001 01111101

得到 flag:

hctf{base_32!}

------------------------------Explorer 的图库之一   POINT: 10------------------------------------

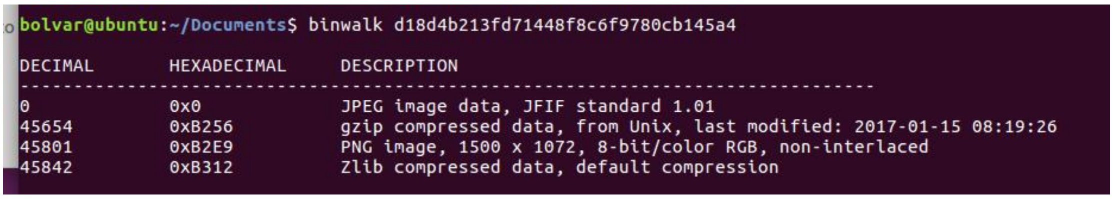
改后缀了是张图，丢进 winhex 打开发现 flag ： hctf{2e3e3}



```
01  ÿØÿà  JFIF
49      ÿá ¦Exif  II
00  *   hctf{2e3e3}
00
```

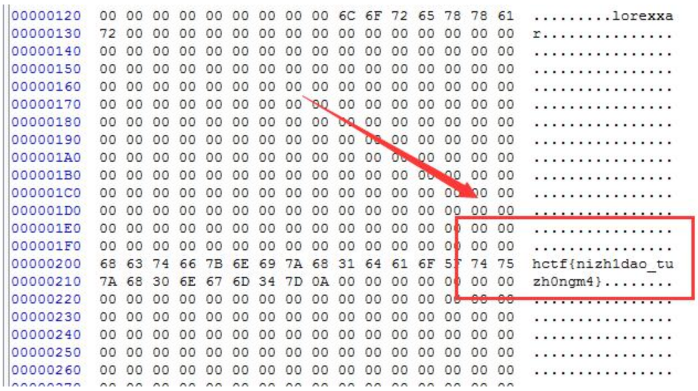
------------------------------Explorer 的图库之二　　POINT: 30-------------------------------------

图片有 1 个多 MB，很大，所以用 binwalk 看了下发现里面有东西



```
bolvar@ubuntu:~/Documents$ binwalk d18d4b213fd71448f8c6f9780cb145a4

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
45654         0xB256          gzip compressed data, from Unix, last modified: 2017-01-15 08:19:26
45801         0xB2E9          PNG image, 1500 x 1072, 8-bit/color RGB, non-interlaced
45842         0xB312          Zlib compressed data, default compression
```

然后把 B256 到 B2E8 之间的东西用 TThexEdit 提出来是个 jar 文件，解压了打开，用 TThex 查看得到 flag: hctf{nizh1dao_tuzh0ngm4}

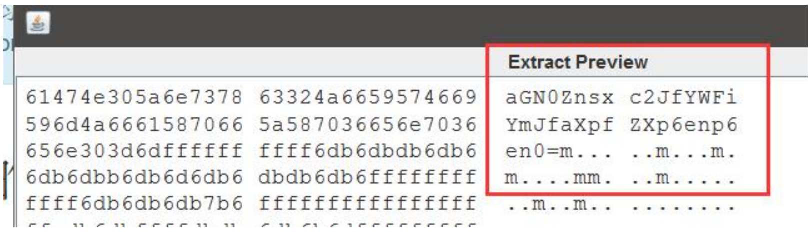


```
00000120  00 00 00 00 00 00 00 00 00 6C 6F 72 65 78 78 61  .........lorexxa
00000130  72 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  r...............
00000140  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000150  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000160  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000170  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000180  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000190  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001A0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001B0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001D0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001E0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001F0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000200  68 63 74 66 7B 6E 69 7A 68 31 64 61 6F 5F 74 75  hctf{nizh1dao_tu
00000210  7A 68 30 6E 67 6D 34 7D 0A 00 00 00 00 00 00 00  zh0ngm4}........
00000220  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000230  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000240  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000250  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000260  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```

-----------------------------Explorer 的图库之三　　POINT: 100------------------------------------

和 MISC10 30 都是同一张图片，用 TThex 把 B2E9 之后的东西提出来是张图片

放进 Stegsolve 里面，打开 analyse--->Data Extract，选中 RGB 的最低有效位后发现了一串 base64 码，解得 flag:　　hctf{1sb_aabbb_iz_ezzzzzz}



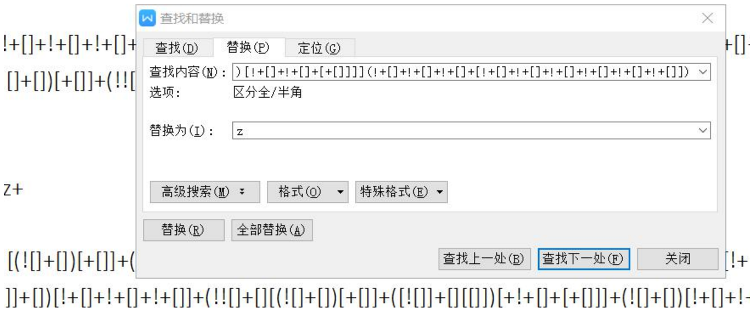
```
61474e305a6e7378 63324a6659574669  aGN0Znsx c2JfYWFi
596d4a6661587066 5a587036656e7036  YmJfaXpf ZXp6enp6
656e303d6dffffff ffff6db6dbdb6db6  en0=m... ..m...m.
6db6dbb6db6d6db6 dbdb6db6ffffffff  m....mm. ..m.....
ffff6db6db6db7b6 ffffffffffffffff  ..m..m.. ........
```

-------------------------------------这 TM 是啥　　POINT: 20--

打开网址，查看源码发现鬼畜的 jsfuck，然而并不会解。。

找到个可以把代码转换成 jsfuck 的在线工具，把”hctf{”和”}”分别转换成 jsfuck,在那些字符中查找，找到了，然后把这一段字符抠出来...

word 文档的查找替换真是个好东西...得到 flag:　hctf{j5fuck_1z_m1233}



查找内容(N)：)[!+[]+!+[]+[+[]]](!+[]+!+[]+!+[]+[!+[]+!+[]+!+[]+!+[]+!+[]+!+[]])

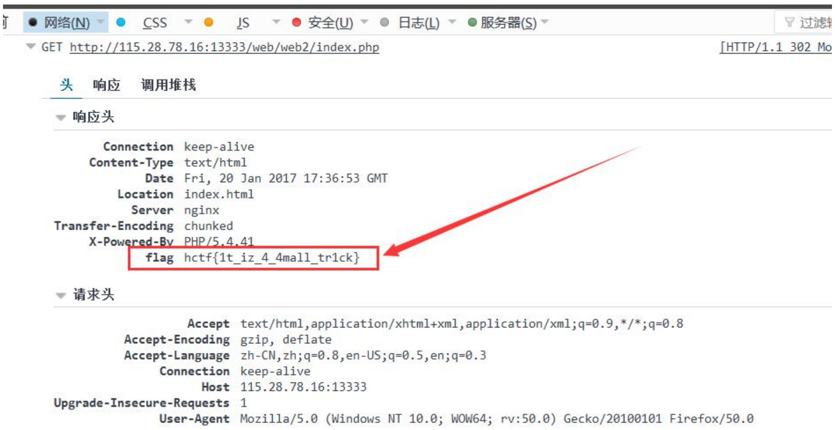
选项：区分全/半角

替换为(I)：z

```
!+[]+!+[]+!+[]+                                                  +[]
[]+[])[+[]]+(!![
z+
[(![]+[])[+[]]+(                                                 [!+
]]+[])[!+[]+!+[]+!+[]]+(!![]+[][(![]+[])[+[]]+([![]]+[][[]])[+!+[]+[+[]]]+(![]+[])[!+[]+!-
```

---------------------------------我是谁我在哪？？？　　POINT: 20---------------------------------

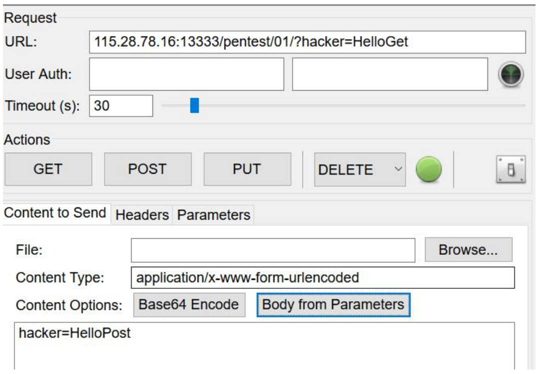
页面从 php 跳转成了 html，用 firefox 的 F12 打开看见 flag:　hctf{1t_iz_4_4mall_tr1ck}



```
GET http://115.28.78.16:13333/web/web2/index.php        [HTTP/1.1 302 Mo

响应头
        Connection  keep-alive
      Content-Type  text/html
              Date  Fri, 20 Jan 2017 17:36:53 GMT
          Location  index.html
            Server  nginx
 Transfer-Encoding  chunked
      X-Powered-By  PHP/5.4.41
              flag  hctf{1t_iz_4_4mall_tr1ck}

请求头
                   Accept  text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
          Accept-Encoding  gzip, deflate
          Accept-Language  zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
               Connection  keep-alive
                     Host  115.28.78.16:13333
Upgrade-Insecure-Requests  1
               User-Agent  Mozilla/5.0 (Windows NT 10.0; WOW64; rv:50.0) Gecko/20100101 Firefox/50.0
```

-------------------------lightless 的渗透教室入门篇（一）　　POINT: 10-------------------------

用 firefox 的 poster 工具，得到 flag：　hctf{PostAndGetIsSoEasy_comeon!}



URL: 115.28.78.16:13333/pentest/01/?hacker=HelloGet

User Auth:

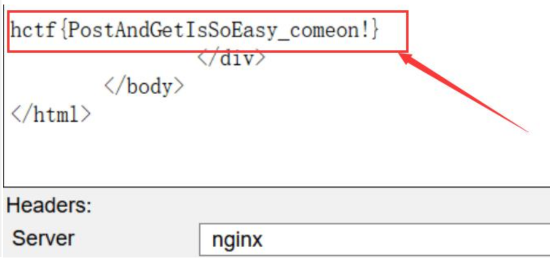
Timeout (s): 30

File:

Content Type: application/x-www-form-urlencoded

Content Options: Base64 Encode　Body from Parameters

hacker=HelloPost



```
hctf{PostAndGetIsSoEasy_comeon!}
        </div>
    </body>
</html>
```

Headers:

Server　nginx

-------------------------------密码学教室入门（一）　　POINT: 20-------------------------------

密文：m,　m^e%N=c,　c 为密文，c^d%N=m

p*q=N,

N:

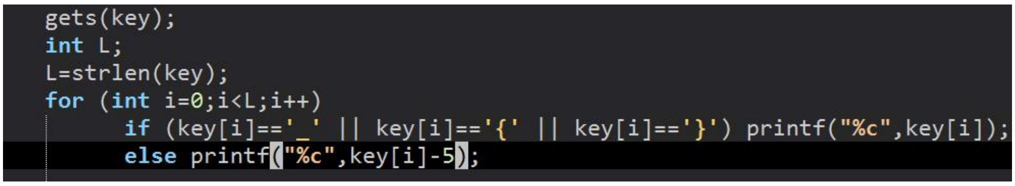
5487B497826693313FECEACA5B5868ADDD959B85A8FD82C62245AC5EF0153A3A7A9550824BE6A117B3B1B549AD3EED37378D13985A6B2CCAA26194000912986009AA12E42568C3F4D390B38D8A3BB583DB27C9FD72AF9C7BE72933C3788AA058A115B140DDFC2067B514A06B4CF27AE1D8AD2A73867614505CB56BC786068547

所以 m 是 6867616D657B7273615F31735F763372795F65347379217D，转成字符串就是

hgame{rsa_1s_v3ry_e4sy!}

--------------------------------密码学教室入门（二）　　POINT: 10------------------------------------

查表发现位移量是 5 密文是 mlfrj{Hfjxfw_hnumjw_8x_ozxy_ktw_kzs}，所以



```
    gets(key);
    int L;
    L=strlen(key);
    for (int i=0;i<L;i++)
        if (key[i]=='_' || key[i]=='{' || key[i]=='}') printf("%c",key[i]);
        else printf("%c",key[i]-5);
```

hgame{Caesar_cipher_3s_just_for_fun}
交上去还是不对啊，然后把数字 3 改成 1，交上去就对了
所以 flag 是 hgame{Caesar_cipher_1s_just_for_fun}

---------------------------------密码学教室入门（三）　　POINT: 50---------------------------------

这个题，我先算了重合指数，很诡异，分析不出密钥长度，GG...
然后发现

ET. PESFVWI, AJGZII BU D LJSQ ISW IKV MRQTLWTOOHRY JP WLJ CCVXNMNH, XJTVLJNFU RR IBTQED'T DHLFMH DX MJU WVNBN. GEWOCB MX SGOIFTGG, SSMA WS GF CUVJTVHH FHCLR QBVHV YICW HFZ. C QIB UTLEQ CGJMST QQ XMF HRPQPYLRL ECB, YSEGU RJX EKEWHGV FWPWJLY CA WLJ EGIEWHGV ESE C WLNSF LRIJXLHZBN ZLT JU VSTO THZJBNHH FT FU. QFOGWXJ. IG KEI XTLXYFP DR FDERYSU QI LNT KPTWJURRRFPW EY UJH LFOFV SK ECURFZ'U IEYIGU ESE JLHIFP LX NO JLW HFNO; HJGCUKJ GQXRI JV ZLNMG VIFSEKMSH VKI HFNO HZSKQK YIG VXTSOLRL PH WLJ CCVXNMNH.

w-w-w:　150 195

c-c : 70

以上取公因数，得到密钥长度是 5，把密文去掉标点符号和空格后整理分组

length_of_key=5:

efjbsiqojcmtfbtmjbostmfthbizuctfpeexhpcehesjbjobtoixfdutjfuoeziefofggjmshfsispcm

tvguqktopcnvutdhuncggacvcvcctgqhycgegwaggcfxnutnfggtpeqkupjfcugjpjncqvgevnkgohcn

pwzdivlhwvhlrqhdwgbogwuhlhwqljqrlbukvjwivwllzvhhuwkldriprwhvuiulllouxzvkkoqvlwvh

eiilsmwrlxxjrelxvemissvhrvhiemxpryreflleelrhlszhqxexryltrelsreehxwhkrlimihkxrlx

Saijwrtyjnjnidfmnwxfsgjfqyfbqsmqlsjwwyjwsnizttjffjiyfsnwryfkfysinhjjinfshzytljn

已知在英文中

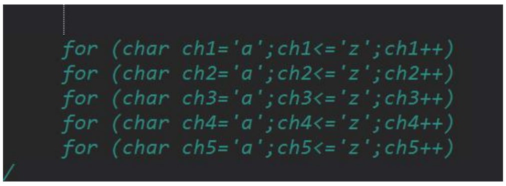
高频字母：E、T、A、O、N、 I、R、S、H

中频字母：D、L、U、C、M

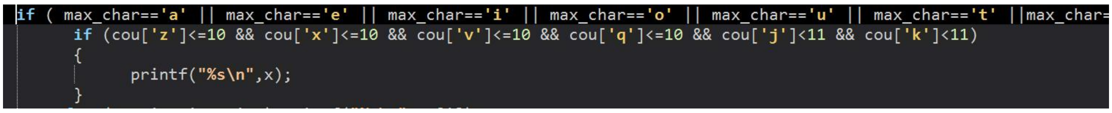
低频字母：P、F、Y、W、G、B、V

稀频字母：J、K、Q、X、Z

剩下的就是暴力了，暴力枚举密钥，解密得到明文，对明文统计字频，基本符合条件的就输出，代码如下

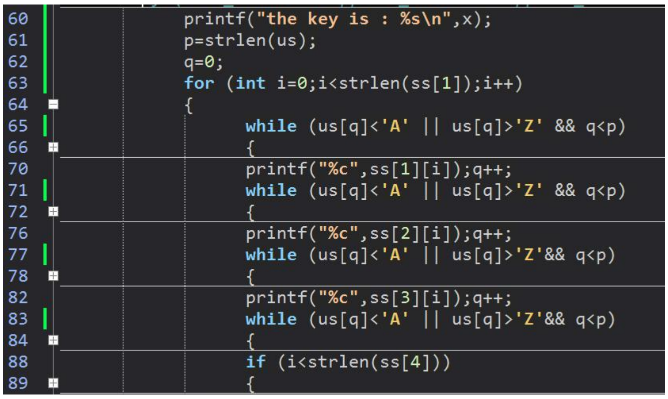


```
for (char ch1='a';ch1<='z';ch1++)
for (char ch2='a';ch2<='z';ch2++)
for (char ch3='a';ch3<='z';ch3++)
for (char ch4='a';ch4<='z';ch4++)
for (char ch5='a';ch5<='z';ch5++)
```



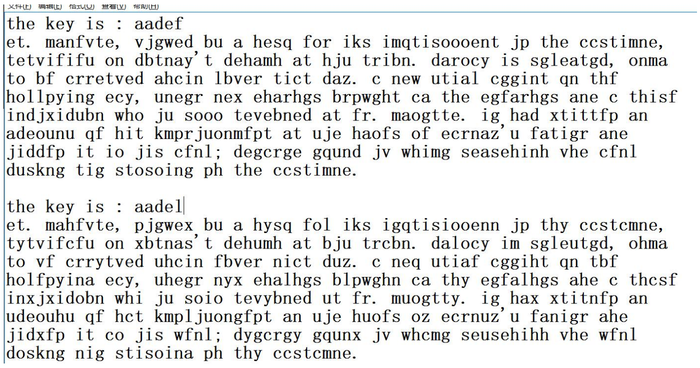
```
if ( max_char=='a' || max_char=='e' || max_char=='i' || max_char=='o' || max_char=='u' || max_char=='t' ||max_char=
    if (cou['z']<=10 && cou['x']<=10 && cou['v']<=10 && cou['q']<=10 && cou['j']<11 && cou['k']<11)
    {
        printf("%s\n",x);
    }
```

非常简单粗暴，这样筛选出了 8579 条密钥，下面就是对着密钥跑明文了



```
        printf("the key is : %s\n",x);
        p=strlen(us);
        q=0;
        for (int i=0;i<strlen(ss[1]);i++)
        {
            while (us[q]<'A' || us[q]>'Z' && q<p)
            {
            printf("%c",ss[1][i]);q++;
            while (us[q]<'A' || us[q]>'Z' && q<p)
            {
            printf("%c",ss[2][i]);q++;
            while (us[q]<'A' || us[q]>'Z'&& q<p)
            {
            printf("%c",ss[3][i]);q++;
            while (us[q]<'A' || us[q]>'Z'&& q<p)
            {
            if (i<strlen(ss[4]))
            {
```

这样得到一堆类似于下面的东西



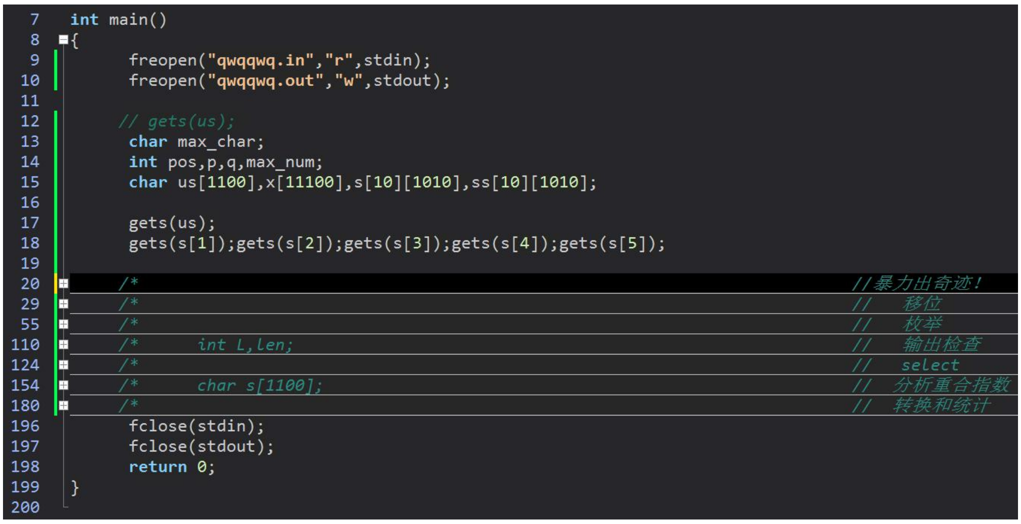
```
the key is : aadef
et. manfvte, vjgwed bu a hesq for iks imqtisoooent jp the ccstimne,
tetvififu on dbtnay't dehamh at hju tribn. darocy is sgleatgd, onma
to bf crretved ahcin lbver tict daz. c new utial cggint qn thf
hollpying ecy, unegr nex eharhgs brpwght ca the egfarhgs ane c thisf
indjxidubn who ju sooo tevebned at fr. maogtte. ig had xtittfp an
adeounu qf hit kmprjuonmfpt at uje haofs of ecrnaz'u fatigr ane
jiddfp it io jis cfnl; degcrge gqund jv whimg seasehinh vhe cfnl
duskng tig stosoing ph the ccstimne.

the key is : aadel
et. mahfvte, pjgwex bu a hysq fol iks igqtisiooenn jp thy ccstcmne,
tytvifcfu on xbtnas't dehumh at bju trcbn. dalocy im sgleutgd, ohma
to vf crrytved uhcin fbver nict duz. c neq utiaf cggiht qn tbf
holfpyina ecy, uhegr nyx ehalhgs blpwghn ca thy egfalhgs ahe c thcsf
inxjxidobn whi ju soio tevybned ut fr. muogtty. ig hax xtitnfp an
udeouhu qf hct kmpljuongfpt an uje huofs oz ecrnuz'u fanigr ahe
jidxfp it co jis wfnl; dygcrgy gqunx jv whcmg seusehihh vhe wfnl
doskng nig stisoina ph thy ccstcmne.
```

大概有 100 多 MB 吧....复制出来丢进 word 文档里面，word 有自动拼写检查，检索字串”the ”，找到了密钥=bcdef，明文如下

dr. manette, viewed as a hero for his imprisonment in the bastille, testifies on darnay's behalf at his trial. darnay is released, only to be arrested again later that day. a new trial begins on the following day, under new charges brought by the defarges and a third individual who is soon revealed as dr. manette. he had written an account of his imprisonment at the hands of darnay's father and hidden it in his cell; defarge found it while searching the cell during the storming of the bastille.

搜索出来得到 flag=hgame{A_Tale_of_Two_Cities}

手撸 200 行代码，GG



```
int main()
{
        freopen("qwqqwq.in","r",stdin);
        freopen("qwqqwq.out","w",stdout);

     // gets(us);
        char max_char;
        int pos,p,q,max_num;
        char us[1100],x[11100],s[10][1010],ss[10][1010];

        gets(us);
        gets(s[1]);gets(s[2]);gets(s[3]);gets(s[4]);gets(s[5]);

     /*                                                                      //暴力出奇迹!
     /*                                                                      //    移位
     /*                                                                      //    枚举
     /*      int L,len;                                                      //    输出检查
     /*                                                                      //    select
     /*      char s[1100];                                                   //   分析重合指数
     /*                                                                      //   转换和统计
        fclose(stdin);
        fclose(stdout);
        return 0;
}
```

------------------------------密码学教室入门（四）　　POINT: 20------------------------------

m^e%N=c，这里 e=1，N 明显大于 c，所以 m=c=:
6867616d657b7273615f31735f737469316c5f653473795f6e6f77217d;
转成字符串：hgame{rsa_1s_sti1l_e4sy_now!}

----------------------------------密码学教室番外篇　　POINT: 20----------------------------------

一开始想复杂了，还以为是不同地方套了几层不同的位移量...然后怎么都试不出来
突然想起来前后 2 个花括号都没有变 是不是符号也没有变呢
偏移量是 17，一试果然出来了,mmp....hgame{dgfdyhcry42287235413//+/%}
[这根本就不像是个 flag...]

That's all above!

By Bolvar