



南开大学  
Nankai University

南 开 大 学

网 络 空 间 安 全 学 院

现代密码学实验报告

---

## 实验一 古典密码算法及攻击方法

---

于文明

年级：2020 级

专业：信息安全

指导教师：古力

2022 年 11 月 16 日

# 摘要

关键字: Classical Cryptography Shift Cipher Substitution Cipher Letter Frequency Attack

# 目录

一、 实验内容	1
二、 移位密码	1
(一) 实验原理 . . . . .	1
(二) 算法流程图 . . . . .	1
(三) 代码实现 . . . . .	1
(四) 测试结果 . . . . .	3
三、 移位密码攻击过程	3
四、 置换密码	4
(一) 实验原理 . . . . .	4
(二) 算法流程图 . . . . .	5
(三) 代码实现 . . . . .	5
(四) 测试结果 . . . . .	7
五、 字母频率攻击	7
(一) 介绍 . . . . .	7
(二) 解题过程 . . . . .	7

## 一、 实验内容

- (1) 根据实验原理部分对移位密码算法的介绍，自己创建明文信息，并选择一个密钥，编写移位密码算法实现程序，实现加密和解密操作。
- (2) 两个同学为一组，互相攻击对方用移位密码加密获得的密文，恢复出其明文和密钥。
- (3) 已创建明文信息，并选择一个密钥，构建置换表。编写置换密码的加解密实现程序，实现加密和解密操作。
- (4) 用频率统计方法，破译用单表置换加密的一段密文

## 二、 移位密码

### (一) 实验原理

移位密码 [1] 将英文字母向前或向后移动一个固定位置，来实现字母表的置换。如果将 26 个英文字母进行编码：A→0, B→1, …, Z→25，则加密过程可简单地写成：明文： $m = m_1m_2 \dots m_i \dots$ ，则有密文： $c = c_1c_2 \dots c_i \dots$ ，其中  $c_i = (m_i + \text{key} \bmod 26)$ ,  $i = 1, 2, \dots$ 。

### (二) 算法流程图

具体的移位密码加密解密算法流程图如下

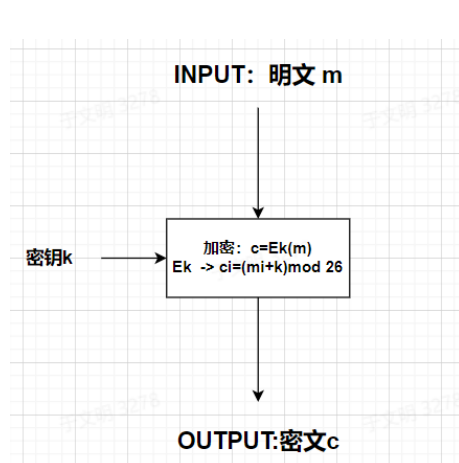


图 1: 移位算法流程图

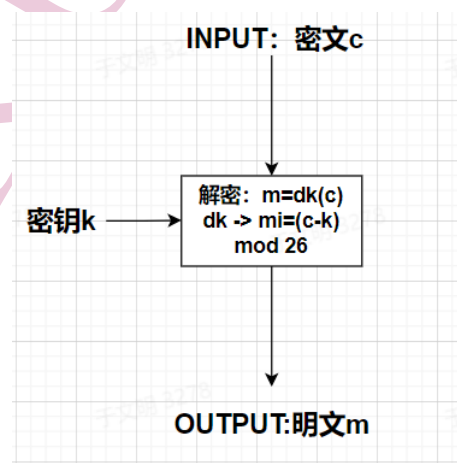


图 2: 移位解密流程图

### (三) 代码实现

加密算法按照实验原理描述的过程即可实现  
加密算法

移位密码加密算法

```

1 string encode() {
2     string message, dm="";
3     getline(cin, message);
4     for (int i = 0; i < message.length(); i++)
5     {
  
```

```

6         int m, c;
7         if (message[i] >= 'a' && message[i] <= 'z')
8         {
9             m = message[i] - 'a';
10            c = (m + key) % 26;
11            dm += char('a' + c);
12        }
13        else if (message[i] >= 'A' && message[i] <= 'Z')
14        {
15            m = message[i] - 'A';
16            c = (m + key) % 26;
17            dm += char('A' + c);
18        }
19        else
20            dm += message[i];
21    }
22    return dm;
23 }

```

解密过程只需要对加密过程逆向即可，需要注意的是防止结果出现负数

解密算法

移位密码解密算法

```

1 void decode() {
2     int guesskey=1;
3     string dm;
4     getline(cin, dm);
5     for (; guesskey <= 25; guesskey++)
6     {
7         cout << "key:" << guesskey << " ";
8         for (int i = 0; i < dm.length(); i++)
9         {
10            int c;
11            if (dm[i] >= 'a' && dm[i] <= 'z')
12            {
13                c = dm[i] - 'a';
14                cout << char('a' + (c - guesskey + 26) % 26);
15            }
16            else if (dm[i] >= 'A' && dm[i] <= 'Z')
17            {
18                c = dm[i] - 'A';
19                cout << char('A' + (c - guesskey + 26) % 26);
20            }
21            else
22                cout << dm[i];
23        }
24        cout << endl;
25    }
26    return ;

```

27 }

#### (四) 测试结果



图 3: 移位算法结果

### 三、 移位密码攻击过程

本节找到同学算法得到的密文 N qtaj SfsPfn Zsna jwxnyd, 采用穷举攻击, 编写如下算法, 对 key(1:25) 遍历解密密文, 取其中有意义的明文即可得到加密密钥 k 和明文 m, 具体实现和结果如下:

穷举攻击算法

移位密码解密算法

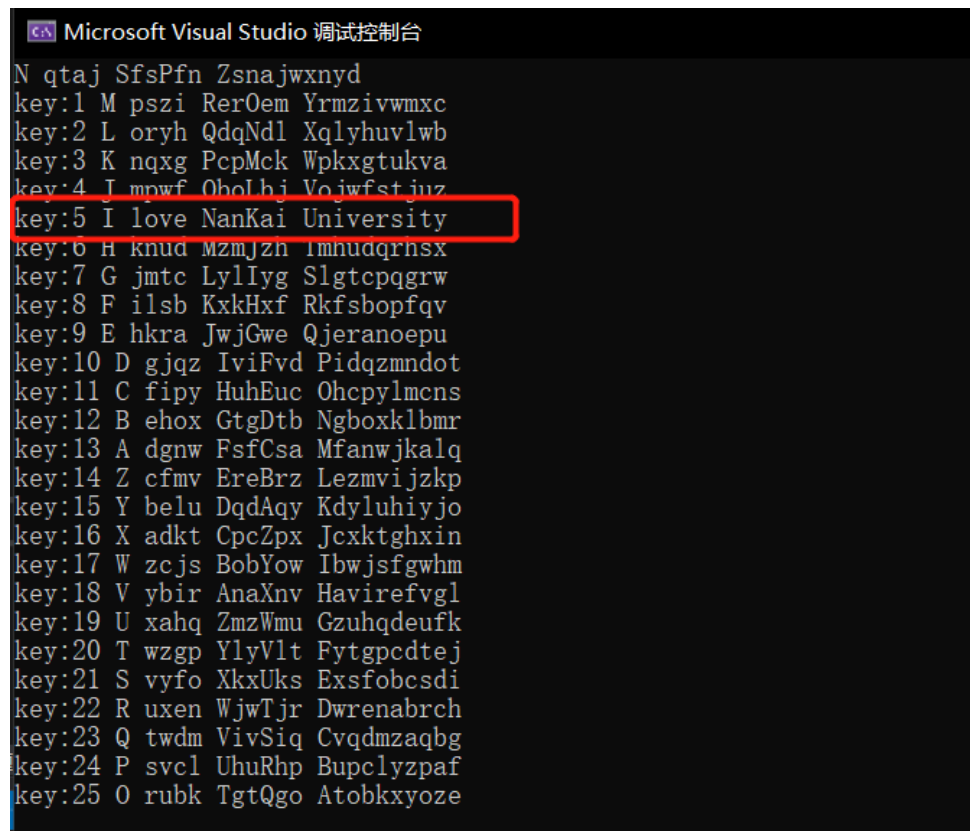
```

1  int main() {
2      int guesskey=1;
3      string dm;
4      getline(cin, dm);
5      for (; guesskey <= 25; guesskey++)
6      {
7          cout << "key:" << guesskey<<" ";
8          for (int i = 0; i < dm.length(); i++)
9          {
10             int c;
11             if (dm[i] >= 'a' && dm[i] <= 'z')
12             {
13                 c = dm[i] - 'a';
14                 cout << char('a' + (c - guesskey + 26) % 26);
15             }
16             else if (dm[i] >= 'A' && dm[i] <= 'Z')
17             {
18                 c = dm[i] - 'A';
19                 cout << char('A' + (c - guesskey + 26) % 26);
20             }
21             else
22                 cout << dm[i];
23         }
24         cout << endl;
25     }

```

```
26     return 0;  
27 }
```

结果如下图



```
Microsoft Visual Studio 调试控制台  
N qtaj SfsPfn Zsnajwxnyd  
key:1 M pszi RerOem Yrmzivwmx  
key:2 L oryh QdqNdl Xqlyhuvlwb  
key:3 K nqyg PcpMck Wpkxgtukva  
key:4 J mpwf OhoLhj Vojwfstjuz  
key:5 I love Nankai University  
key:6 H knud MzmJzn Imnudqrnsx  
key:7 G jmtc LylIyg Slgtcpqgrw  
key:8 F ilsb KxkHxf Rkfsbopfqv  
key:9 E hkra JwjGwe Qjeranoepu  
key:10 D gjqz IviFvd Pidqzmdot  
key:11 C fipy HuhEuc Ohcplymcns  
key:12 B ehox GtgDtb Ngboxklbmr  
key:13 A dgnw FsfCsa Mfanwjkalq  
key:14 Z cfmv EreBrz Lezmvijskp  
key:15 Y belu DqdAqy Kdyluhijjo  
key:16 X adkt CpcZpx Jcxktghxin  
key:17 W zcjs BobYow Ibwjsfgwhm  
key:18 V ybir AnaXnv Havirefvgl  
key:19 U xahq ZmzWmu Gzuhqdeufk  
key:20 T wzgp YlyVlt Fytgpcdtej  
key:21 S vyfo XkxUks Exsfobcsdi  
key:22 R uxen WjwTjr Dwrenabrch  
key:23 Q twdm VivSiq Cvqdmzaqbg  
key:24 P svcl UhuRhp Bupclyzpaf  
key:25 O rubk TgtQgo Atobkxyoze
```

图 4: 穷举攻击结果

从图中可以分析得出加密的密钥 key 为 5, 明文 m 为 I love Nankai University

## 四、 置换密码

### (一) 实验原理

单表置换密码就是根据字母表的置换对明文进行变换的方法, 单表置换实现的一个关键问题是关于置换表的构造。置换表的构造可以有各种不同的途径, 主要考虑的是记忆的方便。如使用一个短语或句子, 删去其中的重复部分, 作为置换表的前面的部分, 然后把没有用到的字母按字母表的顺序依次放入置换表中。

## (二) 算法流程图

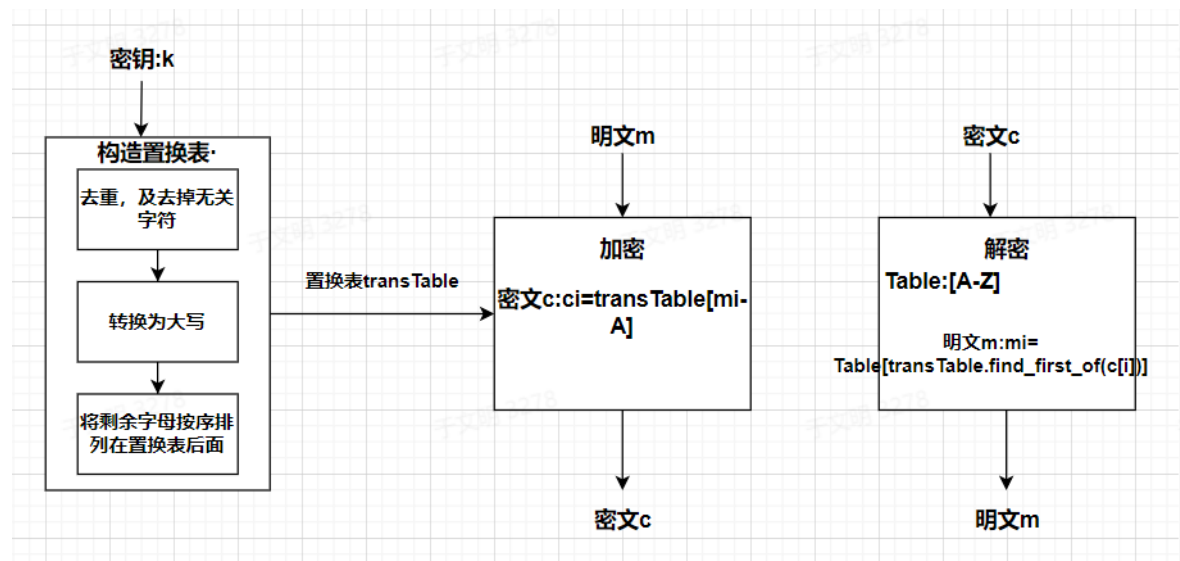


图 5: 单表置换加密解密流程图

## (三) 代码实现

构造置换表

构造置换表

```

1 string process() {
2
3     string trans;
4     for (int i = 0; i < strlen(table); i++)
5     { // 转换为大写
6         if (table[i] >= 'a' && table[i] <= 'z')
7             table[i] = char(table[i] - 'a' + 'A');
8     }
9     for (int i = strlen(table)-1; i >= 0; i--)
10    { // 逆向处理
11        bool flag = false;
12        if (table[i] < 'A' || table[i] > 'Z')
13            continue;
14        for (int j = i - 1; j >= 0; j--)
15        {
16            if (table[i] == table[j])
17                flag = true;
18        }
19        if (!flag)
20            trans += table[i];
21    }
22    reverse(trans.begin(), trans.end());
23    for (int i = 0; i < Table.length(); i++)
24    {
  
```

```

25         if (trans.find(Table[i]) != trans.npos)
26         {
27             continue;
28         }
29         else
30             trans += Table[i];
31     }
32     return trans;
33 }

```

## 加密算法

## 单表置换解密算法

```

1  string encode(string message, string transTable) {
2      string em;
3      for (int i = 0; i < message.length(); i++)
4      {
5          if (message[i] >= 'a' && message[i] <= 'z')
6              message[i] = char(message[i] - 'a' + 'A');
7          else if (message[i] < 'A' || message[i] > 'Z')
8          {
9              em += message[i];
10             continue;
11         }
12         em += char(transTable[message[i] - 'A']);
13     }
14     return em;
15 }

```

## 解密算法

## 单表置换解密算法

```

1  string decode(string em, string transTable)
2  {
3      string dm;
4      for (int i = 0; i < em.length(); i++)
5      {
6          if (em[i] < 'A' || em[i] > 'Z')
7          {
8              dm += em[i];
9              continue;
10         }
11         dm += Table[transTable.find_first_of(em[i])];
12     }
13     return dm;
14 }

```



## (四) 测试结果

密钥  $k = \text{"I like crypto very much"}$ ，加密的明文为  $\text{"I LIKE NANKAI UNIVERSITY"}$ ，结果如下图示：



图 6: 单表置换加密解密结果

## 五、 字母频率攻击

### (一) 介绍

字母频率 (character frequency): 在 1M 字节旧的电子文本中，对字母“A”到“Z”（忽略大小写）分别进行统计。发现近似频率（以百分比表示）：

e 11.67 t 9.53 o 8.22 i 7.81 a 7.73 n 6.71 s 6.55  
r 5.97 h 4.52 l 4.3 d 3.24 u 3.21 c 3.06 m 2.8  
p 2.34 y 2.22 f 2.14 g 2.00 w 1.69 b 1.58 v 1.03  
k 0.79 x 0.30 j 0.23 q 0.12 z 0.09

从该表中可以看出，最常用的单字母英文是 e 和 t，其他字母使用频率相对来说就小得多。这样，攻击一个单表置换密码，首先统计密文中最常出现的字母，并据此猜出两个最常用的字母，并根据英文统计的其他特征（如字母组合等）进行试译。

需要破译的密文为 SIC GCBSPNA XPMHACQ JB GPYXSMEPNXIY JR SINS MF SPNBRQJSSJBE JBFMPQNSJMB FPMQ N XMJBS N SM N XMJBS H HY QCNBR MF N XMRRJHAY JBRCGZPC GINBBCA JB RZGI N VNY SINS SIC MPJEJBNA QCRRNEC GNB MBAY HC PCGMTCPD HY SIC PJEISFZA PCGJXJCBSR SIC XNPSJGJXNBSR JB SIC SPNBRNGSJMB NPC NAJGC SIC MPJEJBNSMP MF SIC QCRRNEC HMH SIC PCGCJTCP NBD MRGNP N XMRRJHAC MXXMBCBS VIM VJRICR SM ENJB ZBNZSIM-PJOCD GMBSPMA MF SIC QCRRNEC

### (二) 解题过程

首先，求出字母频率表并排序

#### 单表置换解密算法

```
1 em="SIC GCBSPNA XPMHACQ JB GPYXSMEPNXIY JR SINS MF SPNBRQJSSJBE JBFMPQNSJMB
  FPMQ N XMJBS N SM N XMJBS H HY QCNBR MF N XMRRJHAY JBRCGZPC GINBBCA JB
  RZGI N VNY SINS SIC MPJEJBNA QCRRNEC GNB MBAY HC PCGMTCPD HY SIC
  PJEISFZA PCGJXJCBSR SIC XNPSJGJXNBSR JB SIC SPNBRNGSJMB NPC NAJGC SIC
  MPJEJBNSMP MF SIC QCRRNEC HMH SIC PCGCJTCP NBD MRGNP N XMRRJHAC MXXMBCBS
  VIM VJRICR SM ENJB ZBNZSIM-PJOCD GMBSPMA MF SIC QCRRNEC"
2 ch="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
```

```

3 p={}
4 for i in ch:
5     p[i]=round(em.count(i)/len(em),4)
6 p=sorted(p.items(),key=lambda x:(x[1],x[0]),reverse=True)

```

结果如下:

[('C', 0.0887), ('S', 0.0813), ('N', 0.0764), ('M', 0.0714), ('J', 0.069), ('B', 0.069), ('P', 0.0567), ('R', 0.0517), ('T', 0.0443), ('G', 0.0345), ('X', 0.0296), ('A', 0.0246), ('H', 0.0222), ('E', 0.0222), ('Q', 0.0197), ('Y', 0.0172), ('F', 0.0172), ('Z', 0.0123), ('V', 0.0074), ('D', 0.0074), ('I', 0.0049), ('O', 0.0025), ('W', 0.0), ('U', 0.0), ('L', 0.0), ('K', 0.0)]

(1) 首先确定定义一个置换表 table, 将前几个频率靠前的字母确定下来, 比如 "C" → "e", "S" → "t", "M" → "o"

(2) 同时因为单个字母为单词的只有 a 和 I, 观察到字母表中单个字母的单词有 N 和 H, 根据前后确定 "N" → "a", "H" → "I"

前两步替换完成, 结果为: tIe GeBtPaA XPoHAeQ JB GPYXtoEPaXIY ...

(3) 观察到单词 tIe, 猜测为单词 the, 所以有 "I" → "h"

(4) 观察到存在... FPoQ a XoJBt a to ... 猜测 FPoQ 为 from, 所以有 "P" → "r", "Q" → "m"

(5) 根据 JR 和频率结合进行分析, 猜测 "J" → "i", "R" → "s"

此时结果为: the GeBtraA XroHAem iB GrYXtoEraXhY is that oF traBsmittiBE iBFormatioB From a XoiBt a to a XoiBt H HY meaBs oF a XossiHAY iBseGZre GhaBBaA iB sZGh a VaY that the oriEiBaA messaEe GaB oBAY He reGoTereD HY the riEhtFZA reGiXieBts the XartiGiXaBts iB the traBsaGtioB are aAiGe the oriEiBator oF the messaEe HoH the reGeiTer aBD osGar a XossiHAe oXXoBeBt Vho Vishes to EaiB ZBaZthoriOeD GoBtroA oF the messaEe

(6) 根据单词 messaEe 猜测为单词 message, 所以有 "E" → "g", "VaY", "Vho", "Vishes" "V" → "w", "iB", "aBD" "B" → "n"

此时结果为 the GentraA XroHAem in GrYXtograXhY ...

(7) GentraA 猜测为 central, GrYXtogrXhy 猜测为 cryptography, 所以有 "G" → "c", "A" → "l", "X" → "p"

(8) proHlem 为 problem, 所以有 "H" → "b"

此时结果中有... s oF a possiblY insecZre channel in sZch a waY ...

(9) 有 "Z" → "u"

(10) 根据 recoTereD 为 recovered, 有 "T" → "v"

(11) 最后一个单词 unauthoriOeD 为 unauthorized, 所以有 "O" → "z"

**置换表和明文** 综上, 置换表为 'A': 'l', 'B': 'n', 'C': 'e', 'E': 'g', 'G': 'c', 'H': 'b', 'I': 'h', 'J': 'i', 'M': 'o', 'N': 'a', 'O': 'z', 'P': 'r', 'Q': 'm', 'R': 's', 'S': 't', 'T': 'v', 'V': 'w', 'X': 'p', 'Z': 'u'

明文为: the central problem in cryptography is that of transmitting information From a point a to a point b bY means oF a possiblY insecure channel in such a waY that the original message can only be recovereD bY the rightFul recipients the participants in the transaction are alice the originator oF the message bob the receiver anD oscar a possible opponent who wishes to gain unauthorizeD control oF the message

## 参考文献

- [1] 吴世忠、宋晓龙、郭涛等译 Paul Garrett 著. *An Introduction to Cryptology*. 机械工业出版社, 2003.

NIKU