Hill密码的加密、解密与破译

李青林, 5110309074
郑辉煌, 5110209289

实验任务1

问题描述

在问题(2)中，若已知密文前4个字母OJWP分别代表TACO，问能否将此段密码破译?

解答

密文: $\begin{pmatrix} \text{O} \\ \text{J} \end{pmatrix} \longrightarrow$ 明文: $\begin{pmatrix} \text{T} \\ \text{A} \end{pmatrix}$, 密文: $\begin{pmatrix} \text{W} \\ \text{P} \end{pmatrix} \longrightarrow$ 明文: $\begin{pmatrix} \text{C} \\ \text{O} \end{pmatrix}$

$$\begin{pmatrix} \text{O} \\ \text{J} \end{pmatrix} \leftrightarrow \beta_1 = \begin{pmatrix} 15 \\ 10 \end{pmatrix} = A\alpha_1 \Leftrightarrow \alpha_1 = \begin{pmatrix} 20 \\ 1 \end{pmatrix} \leftrightarrow \begin{pmatrix} \text{T} \\ \text{A} \end{pmatrix}$$
$$\begin{pmatrix} \text{W} \\ \text{P} \end{pmatrix} \leftrightarrow \beta_1 = \begin{pmatrix} 23 \\ 16 \end{pmatrix} = A\alpha_1 \Leftrightarrow \alpha_1 = \begin{pmatrix} 3 \\ 15 \end{pmatrix} \leftrightarrow \begin{pmatrix} \text{C} \\ \text{O} \end{pmatrix}$$

$\det(\beta_1, \beta_2) = \begin{vmatrix} 15 & 23 \\ 10 & 16 \end{vmatrix} = 10$
$\gcd(10, 26) = 2 \Rightarrow \beta_1, \beta_2$在模26下线性相关
因此无法解密

实验任务2

问题描述

设英文26个字母以下面乱序表与Z_{26}中的整数对应：

A	B	C	D	E	F	G	H	I	J	K	L	M
5	23	2	20	10	15	8	4	18	25	0	16	13
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
7	3	1	19	6	12	24	21	17	14	22	11	9

1. 设$A = \begin{pmatrix} 8 & 6 & 9 & 5 \\ 6 & 9 & 5 & 10 \\ 5 & 8 & 4 & 9 \\ 10 & 6 & 11 & 4 \end{pmatrix}$, 验证矩阵$A$能否作为$\mathrm{Hill}_4$密码体制的加密矩阵。

2. 设明文为
   HILL CRYPTOGRAPHIC SYSTEM IS TRADITIONAL
   利用上面的表值与加密矩阵给此明文加密，并将得到的密文解密.

3. 已知在上述给定表值下的一段Hill_4密码的密文为
   JCOW ZLVB DVLE QMXC
   对应的明文为
   DELAY OPERATIONSU
   能否确定加密矩阵?

解答

1. $\det(A) = 25 \pmod{26}$
   $\gcd(25, 26) = 1 \implies A$在模26下可逆
   因此A可以作为加密矩阵

2. 对明文分组:
   HILL CRYP TOGR APHI CSYS TEMI STRA DITI ONAL
   构造4维向量

$$\begin{pmatrix} 18 \\ 25 \\ 13 \\ 13 \end{pmatrix}, \begin{pmatrix} 20 \\ 12 \\ 9 \\ 19 \end{pmatrix}, \begin{pmatrix} 21 \\ 1 \\ 4 \\ 12 \end{pmatrix}, \begin{pmatrix} 23 \\ 19 \\ 18 \\ 25 \end{pmatrix}, \begin{pmatrix} 20 \\ 24 \\ 9 \\ 24 \end{pmatrix}, \begin{pmatrix} 21 \\ 15 \\ 7 \\ 25 \end{pmatrix}, \begin{pmatrix} 24 \\ 21 \\ 12 \\ 23 \end{pmatrix}, \begin{pmatrix} 10 \\ 25 \\ 21 \\ 25 \end{pmatrix}, \begin{pmatrix} 1 \\ 3 \\ 23 \\ 13 \end{pmatrix}$$

   用A左乘得

$$\begin{pmatrix}8\\8\\17\\5\end{pmatrix},\begin{pmatrix}18\\21\\13\\5\end{pmatrix},\begin{pmatrix}10\\15\\3\\22\end{pmatrix},\begin{pmatrix}13\\25\\18\\18\end{pmatrix},\begin{pmatrix}11\\23\\24\\19\end{pmatrix},\begin{pmatrix}4\\0\\10\\9\end{pmatrix},\begin{pmatrix}21\\25\\23\\18\end{pmatrix},\begin{pmatrix}24\\16\\13\\9\end{pmatrix},\begin{pmatrix}12\\18\\4\\21\end{pmatrix}$$

查表得密文为:
IJMM DSZQ UPHS BQIJ DTZT UFNJ TUSB EJUJ POBM

$$A^{-1}=\begin{pmatrix}23&20&5&1\\2&11&18&1\\2&20&6&25\\25&2&22&25\end{pmatrix}$$

用A^{-1}左乘密文向量得

$$\begin{pmatrix}18\\25\\13\\13\end{pmatrix},\begin{pmatrix}20\\12\\9\\19\end{pmatrix},\begin{pmatrix}21\\1\\4\\12\end{pmatrix},\begin{pmatrix}23\\19\\18\\25\end{pmatrix},\begin{pmatrix}20\\24\\9\\24\end{pmatrix},\begin{pmatrix}21\\15\\7\\25\end{pmatrix},\begin{pmatrix}24\\21\\12\\23\end{pmatrix},\begin{pmatrix}10\\25\\21\\25\end{pmatrix},\begin{pmatrix}1\\3\\23\\13\end{pmatrix}$$

查表即可得明文

3. 对明文分组:
DELA YOPE RATI ONSU
明文向量:

$$\alpha_1=\begin{pmatrix}20\\10\\16\\5\end{pmatrix},\alpha_2=\begin{pmatrix}11\\3\\1\\10\end{pmatrix},\alpha_3=\begin{pmatrix}6\\5\\24\\18\end{pmatrix},\alpha_4=\begin{pmatrix}3\\7\\12\\21\end{pmatrix}$$

$$\det(\alpha_1,\alpha_2,\alpha_3,\alpha_4)=\begin{vmatrix}25&9&20&19\\2&16&17&13\\3&17&16&22\\14&23&10&2\end{vmatrix}=15 \pmod{26}$$

$\gcd(26,15)=1\Rightarrow \alpha_1,\alpha_2,\alpha_3,\alpha_4$在模26下线性无关
密文向量:

$$\beta_1=\begin{pmatrix}25\\2\\3\\14\end{pmatrix},\beta_2=\begin{pmatrix}9\\16\\17\\23\end{pmatrix},\beta_3=\begin{pmatrix}20\\17\\16\\10\end{pmatrix},\beta_4=\begin{pmatrix}19\\13\\22\\2\end{pmatrix}$$

$$\det(\beta_1,\beta_2,\beta_3,\beta_4)=\begin{vmatrix}25&9&20&19\\2&16&17&13\\3&17&16&22\\14&23&10&2\end{vmatrix}=11 \pmod{26}$$

$\gcd(26,11)=1\Rightarrow \beta_1,\beta_2,\beta_3,\beta_4$在模26下线性无关

设加密矩阵为A,则有$A(\alpha_1,\alpha_2,\alpha_3,\alpha_4)=(\beta_1,\beta_2,\beta_3,\beta_4)$
设$C=(\alpha_1,\alpha_2,\alpha_3,\alpha_4),P=(\beta_1,\beta_2,\beta_3,\beta_4)$
$A=PC^{-1}=\begin{pmatrix}8&6&9&5\\6&9&5&10\\5&8&4&9\\10&6&11&4\end{pmatrix}$

实验任务3

问题描述

设已知一份密文为Hill$_2$密码体系，其中出现频数最高的双字母是RH和NI，而在明文语言中，出现频数最高的双字母为TH和HE，由这些信息按下表给出的表值能得到什么样的加密矩阵?

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

解答

$$\begin{pmatrix}R\\H\end{pmatrix}\leftrightarrow\begin{pmatrix}17\\7\end{pmatrix},\begin{pmatrix}N\\I\end{pmatrix}\leftrightarrow\begin{pmatrix}13\\8\end{pmatrix}$$

$$\begin{pmatrix}T\\H\end{pmatrix}\leftrightarrow\begin{pmatrix}19\\7\end{pmatrix},\begin{pmatrix}H\\E\end{pmatrix}\leftrightarrow\begin{pmatrix}7\\4\end{pmatrix}$$

记

$$P=\begin{pmatrix}17&13\\7&8\end{pmatrix},C=\begin{pmatrix}19&7\\7&4\end{pmatrix}$$

设加密矩阵为A,则$P=AC(mod\ 26)$,所以$A=PC^{-1}(mod\ 26)$
通过matlab代码:

```
function Y = invmod( P, C )
    %mod26 inverse matrix
    %for more detail to see <<math experiments>> in page 109
    %D = det(P);
    D = P(2, 2) * P(1, 1) - P(1, 2) * P (2, 1);
    if gcd(D, 26) ~= 1;
        disp('Error');
    else
        for i = 1: 25
            if mod(i * D, 26) == 1
                break;
            end
        end
        invD = i;
        Q(1, 1) =  P(2, 2);
        Q(1, 2) = -P(1, 2);
        Q(2, 1) = -P(2, 1);
        Q(2, 2) = P(1, 1);
        Y = mod(Q * invD, 26);
    end
    Y = mod(C * Y, 26);
end
```

上面针对课本代码的改进是防止了det(P)出现不是整数和inv(P)会有计算机数据误差情况。再通过调用:

```
>> P = [17, 13; 7, 8];
>> C = [19. 7; 7, 4];
>> A = invmod(C, P)
```

得到加密矩阵

$$A = \begin{pmatrix} 3 & 24 \\ 24 & 25 \end{pmatrix}$$

实验任务4

问题描述

如下的密文根据课本表10.1以Hill$_2$加密，密文为
VIKYNOTCLKYRJQETIRECVUZLNOJTUYDIMHRCFITQ
已获知其中相邻字母LK代表字母KE，试破译这份密文。

解答

$$\begin{pmatrix} L \\ K \end{pmatrix} = \begin{pmatrix} 12 \\ 11 \end{pmatrix}$$

$$\begin{pmatrix} K \\ E \end{pmatrix} = \begin{pmatrix} 11 \\ 5 \end{pmatrix}$$

设解密矩阵为$B = \begin{pmatrix} x_1 & x_2 \\ x_3 & x_4 \end{pmatrix}$，则

$$\begin{pmatrix} x_1 & x_2 \\ x_3 & x_4 \end{pmatrix} \begin{pmatrix} 12 \\ 11 \end{pmatrix} = \begin{pmatrix} 11 \\ 5 \end{pmatrix} (mod\ 26)$$

从而解得通解为

$$\begin{pmatrix} x_1 & x_2 \\ x_3 & x_4 \end{pmatrix} = \begin{pmatrix} c_1 & 1 + 6c_1 \\ c_2 & 17 + 6c_2 \end{pmatrix}$$

一开始以C++代码枚举所有可能的c_1和c_2的值，发现所有的密文第5,6个字符“NO”对应的明文都是“OU”，之后猜测明文中KE可能是汉语拼音的“可”，故看后面两个明文是不“YI”对应汉语的“以”，失败了，再转而猜测是英语，猜KE可能是MAKE的后两个字母，用C++代码验证:

```
#include <iostream>
#include <cstdio>
#include <cstdlib>

using namespace std;

const int MAXN = 200;
char code[MAXN] = "VIKYNOTCLKYRJQETIRECVUZLNOJTUYDIMHRCFITQ";
char ans[MAXN];
int length;
int count = 0;
int getNum(char letter)
{
    return (letter - 'A' + 1) % 26;
}

char getLetter(int num)
{
    if(num == 0) return 'Z';
    return 'A' + num - 1;
```

```
}

void printCode(int c1, int c2)
{
    int a1, a2, b1, b2;
    int d1 = (1 + 6 * c1) % 26;
    int d2 = (17 + 6 * c2) % 26;
    for(int i = 0; i < length; i += 2)
    {
        b1 = getNum(code[i]);
        b2 = getNum(code[i + 1]);

        a1 = (c1 * b1 + d1 * b2) % 26;
        a2 = (c2 * b1 + d2 * b2) % 26;

        ans[i] = getLetter(a1);
        ans[i + 1] = getLetter(a2);
        if(code[i] == 'T' && code[i+1] == 'C')
        {
            if(ans[i] != 'M' || ans[i+1] != 'A') return;
            //to see the out file, I can see ans[4] and ans[5]
            //must be 'O' and 'U', so
            //guess ans[3~9] is "you make",
            // so try it and final success!!
        }
        //printf("%c%c", getLetter(a1), getLetter(a2));
    }
    ans[length] = '\0';
    ++count;
    printf("%d\n", count);
    printf(ans);
    printf("\n");
}
int main()
{
    freopen("in.txt", "w", stdout);
    for(length = 0; code[length] != '\0'; ++length);

    for(int i = 0; i < 26; ++i)
        for(int j = 0; j < 26; ++j)
            printCode(i, j);
    return 0;
}
```

得到的输出结果为：

```
1
CANLOUMAKEAAOMEYEGTRWITHOUTBRRAKIAGEGGSS
2
CANYOUMAKEANOMELETTEWITHOUTBREAKINGEGGSS
3
CAALOUMAKENAOMRYRGGRWITHOUTBERAKVAGEGGSS
4
CAAYOUMAKENNOMRLRTGEWITHOUTBEEAKVNGEGGSS
```

注意到第2条，可认为是明文：Can you make an omelette without breaking eggs
(最后一个s为哑字母)

实验任务5

问题描述

若截获一下密文
CKYNOHKQMAXJQBHAZWUHDAOQWXIPQZBKMPUTIPVSWSBYXKKWQHADMBDM
已知它是根据Hill_2体制加密的，能否将它解密？

解答法1

基于字母频数统计的方法：
查阅资料得汉语拼音的字母出现频率（%）头几名为：
I(12.93), N(12.56), G(9.50), U(9.40), A(8.22)
英语出现频率高的为：
E(12.95), T(9.41), A(8.19), O(7.26), N(7.06)

统计得密文共有56个字符，故若明文是拼音，则字母出现频数应满足：

$$I \approx 56 * 12.93\% \approx 7.2$$

$$N \approx 56 * 12.56\% \approx 7.0$$

$$G \approx 56 * 9.5\% \approx 5.3$$

$$U \approx 56 * 9.4\% \approx 5.3$$

$$A \approx 56 * 8.22\% \approx 4.6$$

考虑误差，现在将密文用所有可能的Hill$_2$解密矩阵翻译成明文，首先，明文第一个拼音中出现A,E,I,O,U的概率极小，将这部分数据删去，再将明文中I出现次数小于6个，N小于5 个，G、U、A 小于4 个的数据删去。幸运的是，在这时就得到了答案。（若此时没得到答案，则明文有可能是英语，那么对英文出现频率高的字母同理筛选）下面给出C++代码：

```
#include <iostream>
#include <cstdio>
#include <cstdlib>

using namespace std;

const int MAXN = 200;
char code[MAXN] = ”CKYNOHKQMAXJQBHAZWUHDAOQWXIP
QZBKMPUTIPVSWSBYXKKWQHADMBDM” ;
char ans[MAXN];
int length;
int count = 0;
int getNum(char letter)
{
    return (letter - ’A’ + 1) % 26;
}

char getLetter(int num)
{
    if(num == 0) return ’Z’;
    return ’A’ + num - 1;
}

int gcd(int x, int y)
{
    if(x == 0) return y;
    return gcd(y % x, x);
}
bool analysis()
{
    int count_i = 0,count_n = 0,count_g = 0,count_a = 0,count_u = 0;
    if(ans[0] == ’A’ || ans[0] == ’E’ || ans[0] == ’I’
       || ans[0] == ’O’ || ans[0] == ’U’) return false;
    for(int i = 0; i < length; ++i)
    {
        if(ans[i] == ’A’) ++count_a;
        else if(ans[i] == ’U’) ++count_u;
        else if(ans[i] == ’I’) ++count_i;
        else if(ans[i] == ’N’) ++count_n;
        else if(ans[i] == ’G’) ++count_g;
    }

    if(count_i < 6) return false;
    if(count_n < 5) return false;
    if(count_g < 4) return false;
    if(count_a < 4) return false;
    if(count_u < 4) return false;
    return true;
}

void printCode(int a1, int a2, int a3, int a4)
{
    int det = a1 * a4 - a2 * a3;
    if(det == 0) return;
    if(det > 0 && gcd(det, 26) != 1) return;

    int b1, b2, o1, o2;
    for(int i = 0; i < length; i += 2)
    {
        b1 = getNum(code[i]);
        b2 = getNum(code[i + 1]);

        o1 = (a1 * b1 + a2 * b2) % 26;
        o2 = (a3 * b1 + a4 * b2) % 26;

        ans[i] = getLetter(o1);
        ans[i + 1] = getLetter(o2);


    }
    ans[length] = ’\0’;
    if(analysis())
    {
        ++count;
        printf(”%d\n”, count);
        printf(ans);
        printf(”\n”);
    }
```

```
}
int main()
{
    freopen(”in.txt”, ”w”, stdout);
    for(length = 0; code[length] != ’\0’; ++length);

    for(int i1 = 0; i1 < 26; ++i1)
        for(int i2 = 0; i2 < 26; ++i2)
            for(int i3 = 0; i3 < 26; ++i3)
                for(int i4 = 0; i4 < 26; ++i4)
                    printCode(i1, i2, i3, i4);

    return 0;
}
```

得到的结果为：

```
1
ZAIBENTENGZHIHOUWEIRUANYIJINGTUICHUXINYIDAICAOZUOXITONGG
2
KAGBGNMESGJHUHXUHEARBAIYQJINITLIEHUXINNIMARCPOWUEXWTYNVG
```

注意到第1个答案很符合拼音用法“zai ben teng zhi hou wei ruan yi jing tui chu xin yi dai cao zuo xi tong (哑字母g)”
翻译为汉语：“在奔腾之后微软已经推出新一代操作系统”

解答法2

由于不知道任何加密信息，因此需要枚举所有可能的加密矩阵。
但加密矩阵总数量级达到26^4即上亿级别，人工识别不现实，因此采用计算机过滤+人工识别方法
具体做法如下:

1. 枚举所有合法的解密矩阵
2. 使用这些合法的矩阵对字符串解密，并使用动态规划算法作字符串匹配(字典来自网络)
3. 对匹配位数高的字符串人工识别

解密后的字符串为
zaibentengzhihouweiruanyijingtuichuxinyidaicaozuoxitongg
对应汉语“在奔腾之后，微软已经推出新一代操作系统”
C++代码：

```
#include <cstdio>
#include <cstdlib>
#include <cmath>
#include <cstring>
#include <string>
#include <algorithm>
using namespace std;
const int DICT_SIZE = 1050;
const int STR_LEN = 60;
char dict[DICT_SIZE][30],s[STR_LEN];
int f[STR_LEN],length[DICT_SIZE];
char *str = "CKYNOHKQMAXJQBHAZWUHDAOQWXIPQZBKMPUTIPVSWSBYXKKWQHADMBDM";
int gcd(int a,int b)
{
    return b ? gcd(b,a%b) : a;
}
int match(char *a, char *b)
{
    for(;*a && *b; a++, b++)
        if(*a != *b)
            return 0;
    return !*b;
}
void check(int a, int b, int c, int d)
{
    int det = ((a*d-b*c)%26+26)%26;
    if(gcd(det,26) > 1) return;
    memset(f,0,sizeof(f));
    int len = strlen(str);
    for(int i = 0; i < len; i += 2)
    {
        s[i]   = (a * (str[i] - 64) + b * (str[i+1] - 64)) % 26 + 96;
        s[i+1] = (c * (str[i] - 64) + d * (str[i+1] - 64)) % 26 + 96;
        if(s[i]  ==96) s[i] = 'z';
        if(s[i+1]==96) s[i+1]  = 'z';
    }
    for(int i = 0; i < len; i++)
    {
        f[i] = f[i-1];
        for(int j = 0; j <= 1000; j++)
            if (i+1 >= length[j] && match(s+i-length[j], dict[j]))
                f[i] = max(f[i], (i-length[j]>0 ? f[i-length[j]] : 0)
                +length[j]);
    }
    if(f[len-1]>30)
        printf(”%s␣%d\n”, s, f[len-1]);
}
int main()
{
    FILE *fdict = fopen(”dict1.txt”, ”r”);
    for (int i = 0; i < 1000; ++i)
    {
        fscanf(fdict, ”%s”, dict[i]);
        length[i] = strlen(dict[i]);
    }

    for (int i = 0; i < 25; ++i)
        for (int j = 0; j < 25 ; ++j)
            for (int k = 0; k < 25; ++k)
                for (int l = 0; l < 25; ++l)
                    check(i,j,k,l);
    return 0;
}
```

任务分工

任务1，任务2，任务5解法2，李青林
任务3，任务4，任务5解法1，郑辉煌