



華東師範大學

EAST CHINA NORMAL UNIVERSITY

# 第四讲

---

## 矩阵模运算与古典密码

—— 编程实现

# Hill<sub>2</sub> 加密与解密

## □ 加密过程

- ① 确定加密矩阵 ( 密钥)  $A$  和字母的表值
- ② 将明文字母分组，通过查表列出每组字母对应的向量
- ③ 用  $A$  左乘得新向量，反查字母表值得相应的密文字母

## □ 解密过程

- ① 将密文字母分组，通过查表列出每组字母对应的向量
- ② 求出加密矩阵  $A$  的模  $m$  逆矩阵  $B$
- ③ 用  $B$  左乘得新向量，反查字母表值得相应的明文字母

若所给的明文或密文只含奇数个字母，则需补充一个哑元

# 加密与解密的编程实现

---

□ 可分为下面三个子问题

问题一：建立字母与其表值之间的转换关系

问题二：编程实现加密过程

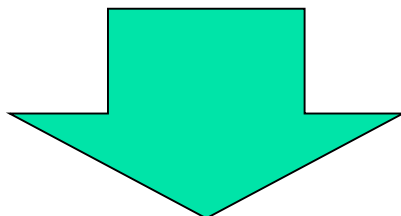
问题三：编程实现解密过程

这里假定加密矩阵及其在模运算下的逆矩阵都已知

# 字母与表值

- 建立 A~Z 与 0~25 之间的一一对应关系：

A	B	C	...	X	Y	Z
↕	↕	↕		↕	↕	↕
1	2	3		24	25	0



字母	A	B	C	...	X	Y	Z
ASCII码	65	66	67	...	88	89	90
表值	1	2	3	...	24	25	0

- 一般规律：表值 = ASCII码 - 64
- 特例：表值为 0 的字母 Z 不满足上述公式，需特殊处理

# 字母与表值

## ① Matlab 编程：计算给定大写字母的表值

```
clear;
str=input('请输入一个大写字母: ');
    % 输入字符时要加单引号
asc=double(str); % 计算该字母的ASCII码

asc=asc-64;      % 计算表值
if asc==26       % 若字母的ASCII码为90, 则表值为0
    asc=0;
end
fprintf('字母 %s 对应的表值为 %d \n', str, asc);
```

# 字母与表值

② 修改上述程序，要求**对输入进行判断**：

如果输入的不是大写字母，则要求重新输入。

```
str=input('请输入一个大写字母：');
```

```
asc=double(str); % 计算该字母的ASCII码
```

```
while (asc>90 | asc<65)
```

```
    str=input('输入错误！请输入一个大写字母：');
```

```
    asc=double(str);
```

```
end
```

```
asc=asc-64; % 计算表值
```

```
if asc==26, asc=0; end
```

```
fprintf('字母 %s 对应的表值为 %d \n', str,asc);
```

# 字母与表值

③ 修改上述程序，当输入大写字母组成的字符串时，计算出该字符串中所有字符的表值。

```
str=input('请输入字符串（全部为大写字母）：');  
asc=double(str); % 计算ASCII码，此时 asc为行向量  
while any(asc>90 | asc<65)  
    str=input('输入错误！应该全部为大写字母：');  
    asc=double(str);  
end  
asc=asc-64; % 计算表值  
for i=1:length(asc); % 对字符串中的 Z 特殊处理  
    if asc(i)==26, asc(i)=0; end  
end  
fprintf('字符串对应的表值为：'); disp(asc);
```

# 字母与表值

④ 编程：计算给定数字(0~25)所对应的大写字母。

```
asc=input('请输入一个数字（0到25之间）：');  
asc=asc+64; % 计算所对应字母的ASCII码  
if asc==64 % 如果输入的数字为 0，则其对应的字母为 z  
    asc=90;  
end  
str=char(asc); % 根据ASCII码算出所对应的字母  
fprintf('对应的字母为 %s \n', str);
```



# 字母与表值

⑤ 修改上述程序，计算一组数字所对应的字符串。

```
m=26;  
asc=input('请输入一行向量: '); % 输入时要加中括号  
asc=mod(asc,m)+64; % 计算模运算后所对应的ASCII码  
for i=1:length(asc); % 对数组中的 0 特殊处理  
    if asc(i)==64, asc(i)=90; end  
end  
str=char(asc); % 根据ASCII码算出所对应的字符串  
fprintf('对应的字符串为 %s \n', str);
```

# 加密过程的 Matlab 实现

## 问题二：编程实现加密过程

- 在模运算意义下，给定加密矩阵，对任意大写字母组成的字符串进行加密。

例：  $m=26$ ，加密矩阵和字母表值分别为：

$$A = \begin{bmatrix} 1 & 2 \\ 0 & 3 \end{bmatrix}$$

A	B	C	D	E	F	G	H	I	J	K	L	M
1	2	3	4	5	6	7	8	9	10	11	12	13

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
14	15	16	17	18	19	20	21	22	23	24	25	0

h1106.m

# 解密过程的 Matlab 实现

## 问题三：编程实现解密过程

- 在模运算意义下，给定加密矩阵在模运算下的逆矩阵，对密文进行解密。

例：  $m=26$ ，加密矩阵模26逆矩阵和字母表值分别为：

$$B = \begin{bmatrix} 1 & 8 \\ 0 & 9 \end{bmatrix}$$

A	B	C	D	E	F	G	H	I	J	K	L	M
1	2	3	4	5	6	7	8	9	10	11	12	13

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
14	15	16	17	18	19	20	21	22	23	24	25	0

h11107.m

# 哑元的选取

例：  $m=26$ , 加密矩阵和字母表值分别为：

$$A = \begin{bmatrix} 1 & 2 \\ 0 & 5 \end{bmatrix}, \quad A^{-1}(\bmod 26) = \begin{bmatrix} 1 & 10 \\ 0 & 21 \end{bmatrix}$$

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

`hill08.m` % 将最后一个字符作为哑元

`hill09.m` % 将表值为0的字符作为哑元

- 思考：
- 1) 哑元对加密解密会产生什么影响？
  - 2) 以上两种哑元的取法分别在什么情况下有效？

# 几个需要注意的问题

# 几个问题

## ❑ 哑元问题

- 与加密矩阵相关
- 必须是合法字符
- 不一定是一个固定的字符
- 哑元选取的原则

$$A \begin{bmatrix} x \\ \Delta \end{bmatrix} = \begin{bmatrix} \tilde{x} \\ \Delta \end{bmatrix}$$

$A$  为加密矩阵,  $x$  代表任意一个合法字符,  $\tilde{x}$  表示加密后的密文中的第一个字符,  $\Delta$  代表哑元

# 几个问题

## ■ 一般的 $n$ 阶矩阵的模 $m$ 逆的计算方法

$$B = (\det(A))^{-1} \bmod(m) \times A^*$$

### ● 伴随矩阵的计算

```
M = A([1:j-1,j+1:n], [1:i-1,i+1:n]);  
B(i,j) = (-1)^(i+j)*det(M);
```

### ● 模 $m$ 逆的计算

```
B = mod(k*B,m); % k 是 det(A) 的模 m 倒数
```

hill11.m

# 几个问题

- 输入的合法性判断

- 求补集；交集和并集的 Matlab 实现

- ASCII码与表值的转换

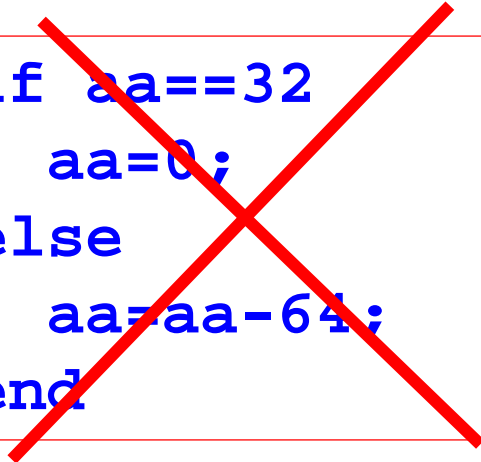
- 特殊情况特殊处理
  - 也可分段处理

- 字符与表值转换的一般方法

hill12.m

- 密文第一个字符可能为空格

例：加密矩阵为  $A = \begin{bmatrix} 1 & 2 \\ 0 & 4 \end{bmatrix}$ , 明文为 YANG



```
if aa==32
    aa=0;
else
    aa=aa-64;
end
```