CRYPTOGRAPHY LAB

مختبر تشفير 2021-2022

المرحلة الثالثة

التدريسين:

م.د. ضحى عبد الهادي عبد الجبار

م.م. سمية سعد سليمان

م.م. امنية حميد جاعد

باشراف:

أ.د. سراب مجيد حميد

Additive Cipher

- The simplest monoalphabetic cipher is the additive cipher. This cipher is sometimes called a shift cipher and sometimes a Caesar cipher
- The **Caesar cipher** involves replacing each letter of the alphabet with the letter standing **three places** further down the alphabet.

Example:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z

- **Encryption with $k = 3$**
  Plaintext : hello
  ciphertext: khoor
- **Decryption with $k = 3$**
  Ciphertext: khoor
  Plaintext : hello

Additive Cipher: How to encrypt

- Thus to cipher a given text we need an integer value, known as shift **(key)** which indicates the number of position each letter of the text has been moved down.

- Convert the letter into the number that matches its order in the alphabet starting from 0, and call this number P. $(a = 0, b = 1, c = 2, \ldots, y = 24, z = 25)$
- Calculate: $C = E\ (P, K)\ =\ (P\ +\ K)\ mod\ n,$ where n is the size of the alphabet.
- Convert the number **C** into a letter that matches its order in the alphabet.

Q/ Based on the following Equation

$$C = E\ (P, K)\ =\ (P + K)\ mod\ n,$$

where n is the size of the alphabet. **Write a java program to encrypt a given plaintext using Additive Cipher with $\boldsymbol{key\ = 4}$.**

Additive Cipher: How to decrypt

For every letter in the cipher text:

- Convert the letter into the number that matches its order in the alphabet starting from 0, and call this number C $P = D\ (C, K)\ =\ (C - K)\ mod\ n$, where n is the size of the alphabet
- Convert the number P into a letter that matches its order in the alphabet starting from 0. $(a = 0, b = 1, c = 2, \ldots, y = 24, z = 25)$.

Cryptography Lab 2021-2022 **3rd Class**
Assignment: 1

Q/ Based on the following Equation

$$P = D\ (C, K)\ =\ (C - K)\ mod\ n,$$

where n is the size of the alphabet. **Write a java program to decrypt a given cipher text using Additive Cipher with** $\boldsymbol{key\ = 4}$**.**

Mixed Alphabet Cipher

The cipher text alphabet is constructed by picking a keyword and writing it down, ignoring repeated letters. Follow it with the letters of the alphabet that have not yet been used.

Example:

- **Encryption using mixed alphabet cipher:**

Plaintext “computer”
keyword “information”

Solution:

Keyword : information
Remove duplicate : informat

Plaintext alphabet	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext alphabet	i	n	f	o	r	m	a	t	b	c	d	e	g	h	j	k	l	p	q	s	u	v	w	x	y	z

Plain text is : “computer”
Cipher text is: “fjgkusrp”

Q/ Write a java program to encrypt a given plaintext using Mixed Alphabet Cipher with keyword “information”.

- ## Decryption using mixed alphabet cipher:

ciphertext “fjgkusrp”
keyword “information”

Solution:

Keyword : information
Remove duplicate : informat

Plaintext alphabet	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext alphabet	i	n	f	o	r	m	a	t	b	c	d	e	g	h	j	k	l	p	q	s	u	v	w	x	y	z

Cipher text is: fjgkusrp
Plain text is : computer

Q/ Write a java program to decrypt a given ciphertext using Mixed Alphabet Cipher with keyword “information”.

Multiplicative Cipher

- To encrypt plain text using multiplicative cipher, each plaintext character is multiplied by $\boldsymbol{K}$, according to the following Equation:

  $$C = E\ (P, K) = (P \times K) mod\ n$$

  where $GCD\ (K, n)\ = 1$.

  For example, 15 and 26 have no factors in common, so 15 is an acceptable value for key however 12 and 26 have factors in common (e.g. 2) so 12 cannot be used for a value of key.

- To decrypt cipher text using multiplicative cipher, we begin by finding the key inverse of K, then apply the following Equation:

  $$P = D(C, K) = C \times K^{-1}\ mod\ n$$

The Key Domain for Any Multiplicative Cipher

Only 12 possible keys: 1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, and 25 have $GCD\ (K, 26) = 1.$ Here are the possible multipliers and their inverses:

$\boldsymbol{K}$	1	3	5	7	9	11	15	17	19	21	23	25
$\boldsymbol{K^{-1}}$	1	9	21	15	3	19	7	23	11	5	17	25

Example

➢ Encryption

We use a multiplicative cipher to encrypt the plaintext “computer” with a key of 9. The cipher text is “swefypkx”.

Plain text	**c**	**o**	**m**	**p**	**u**	**t**	**e**	**r**
Value	2	14	12	15	20	19	4	17
Key ($k = 9$)	9	9	9	9	9	9	9	9
$\boldsymbol{Plain\ text * 9}$	18	126	108	135	180	171	36	135
Mod 26	18	22	4	5	24	15	10	23
Cipher text	s	w	e	f	y	p	k	x

Q/ Based on the following Equation

$$C = E\ (P, K) = (P \times K) mod\ n$$

where $GCD\ (K, n) = 1$ and n is the size of the alphabet. **Write a java program to encrypt a given plaintext using Multiplicative Cipher with $\boldsymbol{key} = 9$.**

➢ Decryption

We use a multiplicative cipher to decrypt the cipher text “swefypkx” with the inverse key of 9 which is $k = 3$. The plaintext “computer”.

Cryptography Lab 2021-2022 **3rd Class**
Assignment: 3

Cipher text	**s**	**w**	**e**	**f**	**y**	**p**	**k**	**x**
Value	18	22	4	5	24	15	10	23
(K^{-1} = 3)	3	3	3	3	3	3	3	3
$\boldsymbol{Cipher\ text * K^{-1}}$	45	66	12	15	72	45	30	69
Mod 26	2	14	12	15	20	19	4	17
Plain text	c	o	m	p	u	t	e	r

Q/ Based on the following Equation

$$P = D(C, K) = C \times K^{-1} \, mod \, n$$

where n is the size of the alphabet. **Write a java program to decrypt a given cipher text using Multiplicative Cipher with $K^{-1} = 3$.**

Affine Cipher

$C = E(P) = (PK_1 + K_2) mod\ n,$ where GCD $(K_1, n) = 1$
$P = D(C) = \left((C - K_2) \times K_1^{-1}\right) mod\ n,$ where GCD $(K_1, n) = 1$

The additive cipher is a special case of an affine cipher in which $K_1 = 1$. The multiplicative cipher is a special case of affine cipher in which $K_2 = 0$.

Example

- **Encryption**

Encrypt the plain text “its cool” using **Affine Cipher** with $K_1 = 5$ and $K_2 = 8$.

$$C = E(P) = (5 \times P + 8) mod\ 26$$

Plain text	i	t	s	c	o	o	l
P	8	19	18	2	14	14	11
$P * 5 + 8$	48	103	98	18	78	78	63
$(P * 5 + 8)$ mod 26	22	25	20	18	0	0	11
Cipher text	w	z	u	s	a	a	l

Q/ Based on the following Equation

$$C = E(P) = (PK_1 + K_2) mod\ n,$$

where GCD $(K_1, n) = 1$ and n is the size of the alphabet. **Write a java program to encrypt a given plaintext using Affine Cipher with $K_1 = 5$ and $K_2 = 8$.**

➢ Decryption

Decrypt the cipher text “**hpccxaq**” using **Affine Cipher** with $K_1 = 5$ and $K_2 = 8$.

Solution : we begin by finding the key inverse of K_1, the key inverse of 5 is 21. Then apply:

$$P = D(C) = \left((C - K_2) \times K_1^{-1}\right) mod\ n$$
$$P = D(C) = \left((C - 8) \times 21\right) mod\ 26$$

Cipher text	h	p	c	c	x	a	q
C	7	15	2	2	23	0	16
$\boldsymbol{C - 8}$	-1	7	-6	-6	15	-8	8
$\boldsymbol{(C - 8) * 21}$	-21	147	-126	-126	315	-168	168
$\boldsymbol{(C - 8) * 21}$ **mod 26**	5	17	4	4	3	14	12
Plain text	f	r	e	e	d	o	m

Q/ Based on the following Equation

$$P = D(C) = \left((C - K_2) \times K_1^{-1}\right) mod\ n,$$

where GCD $(K_1, n) = 1$ and n is the size of the alphabet. **Write a java program to decrypt a given cipher text using Affine Cipher with $K_1^{-1} = 21$ and $K_2 = 8$.**

Vigenere Cipher

- The best known, and one of the simplest, polyalphabetic ciphers is the Vigenère Cipher.
- Each cipher is denoted by a key letter, which is the cipher text letter that substitutes for the plaintext letter.

$$c_i = (p_i + k_i) mod\ n$$
$$p_i = (c_i + k_i) mod\ n$$

- key is needed that is as long as the message. Usually, the key is a repeating keyword.

Example

➢ Encryption

Encrypt the plain text "javatpoint", with the key "best" using Vigenère Cipher.

$$c_i = (p_i + k_i) mod\ 26$$

Plain text	j	a	v	a	t	p	o	i	n	t
Plain text value (P)	09	00	21	00	19	15	14	08	13	19
Key	b	e	s	t	b	e	s	t	b	e
Key value (K)	01	04	18	19	01	04	18	19	01	04
Cipher text value (E)	10	04	13	19	20	19	06	01	14	23

Cipher text	k	e	n	t	u	t	g	b	o	x

Q/ Based on the following Equation

$$c_i = (p_i + k_i)\ mod\ n$$

where n is the size of the alphabet. **Write a java program to encrypt a given plaintext using Vigenere Cipher with the key "best".**

➢ **Decryption**

Decrypt the cipher text "kentutgbox", with the key "best" using Vigenère Cipher.

$$p_i = (c_i + k_i) mod\ 26$$

Cipher text	k	e	n	t	u	t	g	b	o	x
Cipher text value (E)	10	04	13	19	20	19	06	01	14	23
Key	b	e	s	t	b	e	s	t	b	e
Key value (K)	01	04	18	19	01	04	18	19	01	04
Plaintext value (P)	09	00	21	00	19	15	14	08	13	19
Plaintext	j	a	v	a	t	p	o	i	n	t

Q/ Based on the following Equation

$$p_i = (c_i + k_i) mod\ n$$

where n is the size of the alphabet. **Write a java program to decrypt a given cipher text using Vigenere Cipher with the key “best”.**

Beafort Cipher

- Is a polyalphabetic ciphers similar to the Vigenère Cipher.
- Each cipher is denoted by a key letter, which is the cipher text letter that substitutes for the plaintext letter.

$$c_i = (k_i - p_i) mod\ n$$
$$p_i = (k_i - c_i) mod\ n$$

- Key is needed that is as long as the message. Usually, the key is a repeating keyword.

Q/ Based on the following Equation

$$c_i = (k_i - p_i)\ mod\ n$$

where n is the size of the alphabet. **Write a java program to encrypt a given plaintext using Beaufort Cipher with the key “computer”.**

➢ Decryption

In Beaufort cipher, encryption and decryption uses the same algorithm.

Assignment: 6

Q/ Based on the following Equation

$$p_i = (k_i - c_i) mod\ n$$

where n is the size of the alphabet. **Write a java program to decrypt a given cipher text using Beaufort Cipher with the key “computer”.**

Playfair Cipher

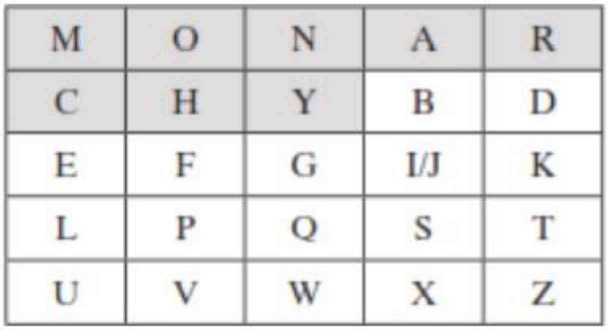
- Is a polyalphabetic ciphers. It was the first practical digraph substitution cipher.
- The technique encrypts pairs of letters (digraphs), instead of single letters as in the simple substitution cipher.
- The Palyfair algorithm based on a 5x5 matrix of letter constructed using a keyword.
- It include the following:

Key generation:

- The 'key' for a Playfair cipher is generally a word.
- The matrix is constructed by filling in the letters of the keyword (minus duplicates) from left to right and from top to bottom, and then filling in the remainder of the matrix with the remaining letters in alphabetic order.
- The letters I and J count as one letter

Example:

When the key word is (monarchy), the matrix will be:



M	O	N	A	R
C	H	Y	B	D
E	F	G	I/J	K
L	P	Q	S	T
U	V	W	X	Z

Assignment: 6

➢ Encryption:

- If the plaintext has an odd number of characters, append an 'x' to the end to make it even.

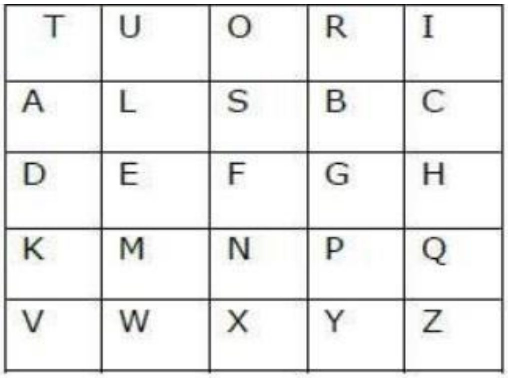
  “come to the window”□ “come to the windowx”
- Repeating plaintext letters that are in the same pair are separated with a filler letter, such as 'x'.

  “balloon” □ ba lx lo on.
- Plaintext is encrypted two letters at a time, according to the following rules:
  1. Two plaintext letters that fall in the same row of the matrix are each replaced by the letter to the right, with the first element of the row circularly following the last.
  2. Two plaintext letters that fall in the same column are each replaced by the letter beneath, with the top element of the column circularly following the last.
  3. Otherwise, each plaintext letter in a pair is replaced by the letter that lies in its own row and the column occupied by the other plaintext letter.

<u>Example:</u>

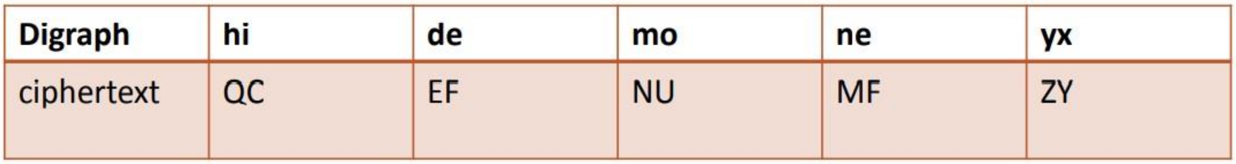
Encrypt the plaintext “hide money” using Palyfair cipher with key “tutorials”

First remove replicate from the key, and prepare the keyword matrix:



T	U	O	R	I
A	L	S	B	C
D	E	F	G	H
K	M	N	P	Q
V	W	X	Y	Z

Then split plain text into pair of letters to encrypt as in the following:



Digraph	**hi**	**de**	**mo**	**ne**	**yx**
ciphertext	QC	EF	NU	MF	ZY

Q/ Write a java program to encrypt a given plaintext using Playfair Cipher with the key “monarchy”.

Playfair Cipher

➢ Decryption:

- Decryption is the exact reverse procedure to the encryption.
- Cipher text decrypted two letters at a time, according to the following rules:
    1. Two cipher text letters that fall in the same row of the matrix are each replaced by the letter to the left, with the first element of the row circularly following the last.
    2. Two cipher text letters that fall in the same column are each replaced by the letter above, with the top element of the column circularly following the last.
    3. Otherwise, each plaintext letter in a pair is replaced by the letter that lies in its own row and the column occupied by the other plaintext letter.

Q/ Write a java program to decrypt a given cipher text using Playfair Cipher with the key "subistitution".

Rail fence Cipher

- Is a transposition cipher where the plaintext characters of a message are systematically rearranged.
- After transposing a message, the same characters are still present, but the order of the letters is changed.

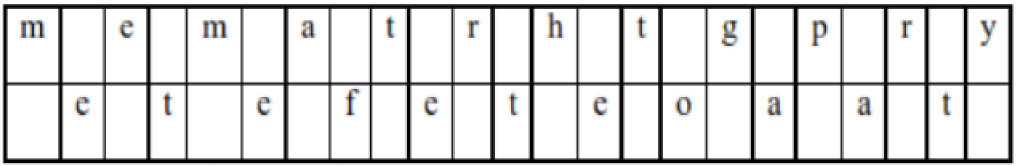
Example:

➢ Encryption

The plaintext is written down as a sequence of diagonals according to a specific depth and then read off as a sequence of rows.

Encrypt the plaintext “meet me after the toga party” .

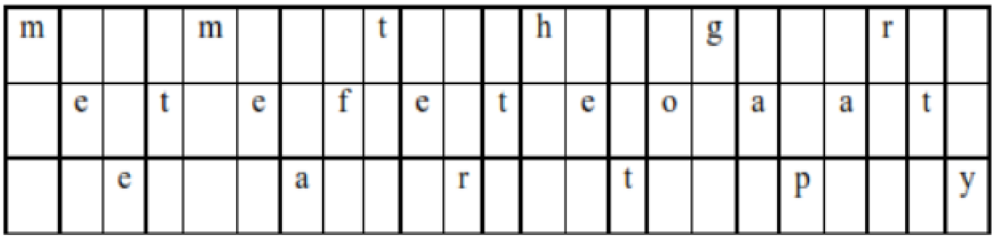
- If the depthe =2:



m		e		m		a		t		r		h		t		g		p		r		y
	e		t		e		f		e		t		e		o		a		a		t	

Ciphertext: MEMATRHTGPRYETEFETEOAAT

- If the depth =3:



m				m				t				h				g				r		
	e		t		e		f		e		t		e		o		a		a		t	
		e				a				r				t				p				y

Ciphertext: MMTHGRETEFETEOAATEARTPY

Q/ Write a java program to encrypt a plaintext “now is the time for all good men” using rail fence Cipher with the depth=2.

➢ **Decryption**

- To decrypt, write half the letters on one line, half on the second.(if the depth=2)
- Note that if there are an odd number of letters, we include the “middle” letter on the top line.

Decrypt the ciphertext “**MKHSE LWYAE ATSOL**” .

- If the depthe =2:

M		K		H		S		E		L		W		Y
	A		E		A		T		S		O		L	

Plaintext: make haste slowly

Q/ Write a java program to decrypt a ciphertext “MEMATRHTGPRYETEFETEOAAT” **using rail fence Cipher with the depth=2.**

Modern encryption Algorithms and java

Modern encryption is the key to advanced computer and communication security. Different algorithms have come up with powerful encryption mechanisms incorporated in them. It gave rise to two new ways of encryption mechanism for data security. These are:

- **Symmetric ke**y encryption (the same key is implemented for both encrypting and decrypting the information, such as AES, DES algorithms).
- **Asymmetric ke**y encryption (it integrates two cryptographic keys for implementing data security. These keys are termed as Public Key and Private Key, such as RSA algorithm).

Implementing Symmetric Encryption in Java

There are classes, interfece and methods used to Implement Symmetric Encryption in Java, they are included in the javax.crypto package:

Interface	Description
SecretKey	A secret (symmetric) key. This interface contains no methods or constants. Its only purpose is to group (and provide type safety for) secret keys.
Class	Description
Cipher	This class provides the functionality of a cryptographic cipher for encryption and decryption.

KeyGenerator	This class provides the functionality of a secret (symmetric) key generator.
Method	Description

`KeyGenerator.getInstance(str)`
`.generateKey();`
	Generates a secret key.

`Cipher.getInstance(str)`	Returns a Cipher object that implements the specified transformation.
`cipher.init(OPERATION MODE, key);`	Initializes this cipher with a key.
`cipher.doFinal( )`	Finishes a multiple-part encryption or decryption operation, depending on how this cipher was initialized.

DES Cipher

- The Data Encryption Standard (DES) is a symmetric-key block cipher.
- DES is an implementation of a Feistel Cipher.
- It uses 16 rounds Feistel structure.
- The block size is 64-bit.
- The key length is 64-bit. DES has an effective key length of 56 bits.
- To encrypt and decrypt a String with DES one should perform the following steps:
  1. Generate a SecretKey using DES algorithm, with the KeyGenerator generateKey() API method.

2. Initialize two Ciphers, one in encryption mode and the other one in decryption mode. Use them to encrypt the String message and then decrypt the encrypted String.

Example:

➢ Encryption

The encryption is performed in the `String encrypt(String str)` method. It encodes the string into a sequence of bytes using the named charset, storing the result into a new byte array. Then it `doFinal(byte[] input)` API method of Cipher to make the encryption. It uses the `com.sun.mail.util.BASE64EncoderStream` to encode the encrypted

byte array and returns the String created from the byte array.

➢ Decryption

The decryption is performed in the `String decrypt(String str)` method. It uses the `com.sun.mail.util.BASE64DecoderStream` to decode the String to byte array. Then it calls `doFinal(byte[] input)` API method of Cipher to make the decryption. It creates a new string based on the specified charset from the decrypted byte array.

Code for DES:

```
import java.security.Security;
import javax.crypto.Cipher;
import javax.crypto.KeyGenerator;
import javax.crypto.SecretKey;
import com.sun.mail.util.BASE64DecoderStream;
import com.sun.mail.util.BASE64EncoderStream;

private static Cipher ecipher;
private static Cipher dcipher;
private static SecretKey key;
```

key generation:

```
  // generate secret key using DES algorithm
     key = KeyGenerator.getInstance("DES").generateKey();
     ecipher = Cipher.getInstance("DES");
     dcipher = Cipher.getInstance("DES");

  // initialize the ciphers with the given key
  ecipher.init(Cipher.ENCRYPT_MODE, key);
  dcipher.init(Cipher.DECRYPT_MODE, key);
```

Encryption:

```
  String encrypted = encrypt("This is a classified message!");

    public static String encrypt(String str) {
   // encode the string into a sequence of bytes using the named charset

    // storing the result into a new byte array.
 byte[] utf8 = str.getBytes("UTF8");
 byte[] enc = ecipher.doFinal(utf8);

return new String(enc);

  }
```

Decryption:

```
  String decrypted = decrypt(encrypted);

  public static String decrypt(String str) {

  // decode with base64 to get bytes

byte[] dec = BASE64DecoderStream.decode(str.getBytes());
byte[] utf8 = dcipher.doFinal(dec);

return new String(utf8, "UTF8");}
```