UIT
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

BÁO CÁO THỰC HÀNH LAB03

GVHD: Nghi Hoàng Khoa

MÔN HỌC:

Nhập môn bảo đảm và an ninh thông tin

Tác giả: Nhóm 02

PRACTICE REPORT
Lab 3: Classical Cryptography

A. OVERVIEW

1. **Introduction and learning objectives**
2. **Backgrounds and Prerequisites**

B. LAB TASKS

1. **1. Group information:**

NUMERICAL ORDER	FULL NAME	STUDENT CODE	EVALUATE
1	**Nguyễn Phú Hào**	**21520223**	**100%**
2	**Đỗ Gia Bảo**	**21520602**	**100%**
3	**Trần Nguyễn Tuấn Anh**	**21521840**	**100%**
4	**Lê Bá Khánh Dũng**	**21521975**	**100%**
5	**Phan Huỳnh Thiên Trang**	**21522698**	**100%**

DETAILED REPORT

Task 1:

1.1 Result :042

Explain:

A. 6 8 2 – One Number is correct and well placed – Number cannot be 6 as that will make Statement B wrong so number in code is 2 which is correctly placed also (on Position 3)

B. 6 1 4 – One Number is correct but wrong place – 6 cannot be that number as per Statement A, It cannot be 1 also as if number is 1 it can be on either on 1st or 3rd place those are already taken by digits 0 and 2 so only number left is 4

B. 2 0 6 – Two Numbers are correct but Wrong Places – One of digit which is correct is 0 as per Statement E but its place cannot be at second(as per statement C) and third position(as per Statement E) so it can be only one first place, other digit is 2 as per statement A.

C. 7 3 8 – Nothing is correct – We can rule out these 3 digits from other statements E. 7 8 0 – One Number is correct but wrong place – We know 0 is one of digit in code but place for it would be not 3

From statement A, B and E we got our numbers. Statement A gave position for Number 2 . Statement C and E gave position for other 2 numbers.

Statement A -> X X 2

Statement A, C and E-> 0 X 2

Statement B -> last digit 4 which will come in middle so code would be 0 4 2

1.2

△	△	◁	◯	?
♡	♡	♠	♡	◊◊
?	?	◁	♣	●●
?	♡	♠	♡	●▷
●♡	◊◊	●●	●◊	

Result:

7	7	5	4	23
9	9	6	9	33
8	8	5	1	22
5	9	6	9	28
29	33	22	23	

Task 2:

- **Code for encryption in Caesar cipher**

```python
# encryption in caesar cipher
def encrypt(text, secret_key):
    result = ""
    for char in text:
        if char.isalpha():
            ascii_offset = 65 if char.isupper() else 97
            # Xác định giá trị offset ASCII tương ứng với chữ cái là chữ hoa
            encrypted_char = chr((ord(char) - ascii_offset + secret_key) % 26 + ascii_offset)
            # ord(char) - ascii_offset: chuyển đổi ký tự thành một số nguyên
            # (... + secret_key) % 26: công thức mã hóa
            # (... + secret_key) % 26 + ascii_offset: trả về giá trị chuỗi tương ứng
            result += encrypted_char
        else:
            result += char
    return result
# input
text = input("Nhập vào chuỗi cần mã hóa: ")
secret_key = int(input("Nhập vào khóa: "))
# Output
encrypted_text = encrypt(text, secret_key)
print("Ciphertext: ", encrypted_text)
```

- **Example:**

  Input: university

  Output: xqlyhuvlwb

```
Nhập vào chuỗi cần mã hóa: university
Nhập vào khóa: 3
Ciphertext:  xqlyhuvlwb
```

- **Code for decryption in Caesar cipher**

```python
def decrypt(ciphertext, secret_key):
    result = ""
    for char in ciphertext:
        if char.isalpha():
            shifted_char = chr((ord(char.lower()) - ord('a') - secret_key) % 26 + ord('a'))
            if char.isupper():
                result += shifted_char.upper()
            else:
                result += shifted_char
        else:
            result += char
    return result
# Mfwzpn Rzwfpfrn bfx gtws ns Pdtyt ns 1949 fsi stb qnajx sjfw Ytpdt. Mj nx ymj fzymtw tk rfsd

#Input
ciphertext = input("Nhập vào chuỗi cần giải mã: ")
secret_key = int(input("Nhập vào khóa: "))
#Output
decrypted_text = decrypt(ciphertext, secret_key)
print("Plaintext: ",decrypted_text)
```

- **Example:**

```
Nhập vào chuỗi cần giải mã: Mfwzpn Rzwfpfrn bfx gtws ns Pdtyt ns 1949 fsi stb qnajx sjfw Ytpdt. Mj nx ymj fzymt
w tk rfsd stajqx fx bjqq fx xmtwy xytwnjx fsi sts-knhynts. Mnx btwpx nshqzij Stwbjlnfs Btti, Ymj Bnsi-Zu Gnwi H
mwtsnhqj, Pfkpf ts ymj Xmtwj, Fkyjw Ifwp fsi Bmfy N Yfqp Fgtzy Bmjs N Yfqp Fgtzy Wzssnsl. Mnx btwp mfx gjjs ywf
sxqfyji nsyt rtwj ymfs ktwyd qfslzfljx, fsi ymj rtxy wjhjsy tk mnx rfsd nsyjwsfyntsfq mtstzwx nx ymj Ojwzxfqjr
Uwnej, bmtxj uwjantzx wjhnunjsyx nshqzij O.R. Htjyejj, Rnqfs Pzsijwf, fsi A.X. Sfnufzq.
Nhập vào khóa: 3
Plaintext:  Jctwmk Owtcmqok ycu dqtp kp Maqvq kp 1949 cpf pqy nkxgu pgct Vqmaq. Jg ku vjg cwvjqt qh ocpa pqxgnu
 cu ygnn cu ujqtv uvqtkgu cpf pqp-hkevkqp. Jku yqtmu kpenwfg Pqtygikcp Yqqf, Vjg Ykpf-Wr Dktf Ejtqpkeng, Mchmc
qp vjg Ujqtg, Chvgt Fctm cpf Yjcv K Vcnm Cdqwv Yjgp K Vcnm Cdqwv Twppkpi. Jku yqtm jcu dggp vtcpuncvgf kpvq oqt
g vjcp hqtva ncpiwcigu, cpf vjg oquv tgegpv qh jku ocpa kpvgtpcvkqpcn jqpqwtu ku vjg Lgtwucngo Rtkbg, yjqug rtg
xkqwu tgekrkgpvu kpenwfg L.O. Eqgvbgg, Okncp Mwpfgtc, cpf X.U. Pckrcwn.
```

Task 3:

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

"Visit the Tabular N-gram Analysis - CrypTool Portal to check the frequency of letters and phrases in the text, and you'll find the phrase 'ytn' appearing frequently, along with the letters 'y', 'v', and 'u'. Commonly occurring English words include: 'e', 't', 'a', 'o', 'i', 'n'. The phrase 'ytn lvd ytn' suggests three-word combinations often found in English, such as 'the', 'one', 'not', 'got', 'for'... However, the repetition of 'ytn' at the beginning of the paragraph may indicate 'the' and 'not'. Since there are many combinations that go with 'ytn', there's an 80% chance it is 'the'. We also have 'ytnhn', where 'not' is unlikely, but 'the' could be 'there' or 'theme', so 'ytn' is likely 'the', and 'h' is either 'r' or 'm'. We have 'ytnu', and considering 'u' is a frequently used word, 'u' is likely 'n'. 'xu' is a frequently occurring phrase, and since 'xu' and 'mu' have individual words, they could be 'i', 'o', or 'a'. The standalone 'v' is likely 'a'. So, 'x' and 'm' are 'o' and 'i'. Considering the word 'xy' is 'to', 'x' is 'o', and 'm' is 'i'. Fill in these findings into the table and continue the search in the text."

Task 3.1:

Result: 'A good glass in the bishop's hostel in the devil's seat forty-one degrees and thirteen minutes northeast and by north main branch seventh limb east side shoot from the left eye of the death's-head a bee line from the tree through the shot fifty feet out.'

‡ and 8 appear frequently and are repeated, so ‡ could be 'n', 'e', and 'o'.

';' appears quite often and is next to 8, so ';' could be 't' or 'n'.

We have the sequence ';', '4', '8' appearing frequently, possibly indicating 'the'. With '8' as 'e', ';' as 't', and '4' as 'h'.

□ ‡ could be 'o'.

We see the sequence ')4‡' separated by '.'. Try '*ho*ho', so it can be split or ‡ is not 'o'.

Additionally, 5 is also a frequently appearing number, so 5 could be 'o', 'a', or 'i'.

There's ' ;5', so '5' could be 'a' or 'o'. If ‡ is 'o', then 'a' is '5'.

Usually, before 'the' is a preposition. So '6*' could be 'in'.

This sequence might be a phrase '53‡‡†305))', which is 'a*oo***a**'.

')' might be a letter after 'h', which could be 'c' or 's'. Try with 's'.

Try everything in the paragraph:

A good glass in the bishop's hostel in the east in the northeast: one easy than thirteen minutes. . . .(long passage)...

According to my guess, † is 'd' as it is near east, and 3 could be 'g' due to the context of 'good'.

From there, I got a fragment 'A good glass in the bishop's hostel'.

Analyze more English words, guess meanings, and search on Google Chrome for a translated text.

Task 4:

4.1

- **Result:**

```
Key text: university
Plain Text: toitenlanguyenphuhaocomasosinhvienlahaimotnamhaikhonghaihaiba
CipherText: altduiotecvruiuprpgzgkoyaktnukevuiotorvllaesoktelkkeboteorudge
```

- Compare the result with other cryptography tools (like Cryptool 2) to verify

4.2

- "ionlyregretthatihavebutonelifetogiveformycountry" will be divided into: "io nl yr eg re tx ha ti ha ve bu to ne li fe to gi ve fo rm yc ou nt ry" - The result: "MAAPZOQHGKHWHMLITMIAKHPBASDGMCDHROCAFKRAFOFANPBLZ

Y”

Explain:

- io -> MA (if two characters form a rectangle, then the encoding character is the character that is on the same line but in opposite corners.)
- nl -> AP
- yr -> ZO
- eg -> QH (If these two characters are in the same column, the encoded character is the character in the line immediately below, in the same column.)

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

Task 5:

- **Code for encryption in Vigenere cipher**

```
def vigenere_encrypt(message, key):
    encrypted_text = ''
    key_length = len(key)
    for i in range(len(message)):
        char = message[i]
        if char.isalpha():
            key_char = key[i % key_length]
            shift = ord(key_char.lower()) - ord('a')
            if char.isupper():
                encrypted_text += chr((ord(char) + shift - ord('A')) % 26 + ord('A'))
            else:
                encrypted_text += chr((ord(char) + shift - ord('a')) % 26 + ord('a'))
        else:
            encrypted_text += char
    return encrypted_text
```

- **Code for decryption in Vigenere cipher**

```python
def vigenere_decrypt(encrypted_text, key):
    decrypted_text = ''
    key_length = len(key)
    for i in range(len(encrypted_text)):
        char = encrypted_text[i]
        if char.isalpha():
            key_char = key[i % key_length]
            shift = ord(key_char.lower()) - ord('a')
            if char.isupper():
                decrypted_text += chr((ord(char) - shift - ord('A')) % 26 + ord('A'))
            else:
                decrypted_text += chr((ord(char) - shift - ord('a')) % 26 + ord('a'))
        else:
            decrypted_text += char
    return decrypted_text
```

- **Example**

```python
# Test hàm với ví dụ
message = "HOCHIMINH"
key = "HAO"
encrypted_message = vigenere_encrypt(message, key)
decrypted_message = vigenere_decrypt(encrypted_message, key)

print("Original Message:", message)
print("Encrypted Message:", encrypted_message)
print("Decrypted Message:", decrypted_message)
```

- **Result**

```
Original Message: HOCHIMINH
Encrypted Message: OOQOIAPNV
Decrypted Message: HOCHIMINH
```

Task 6:

Base to ASCII table below:

Dec	Hex	Oct	Chr	Dec	Hex	Oct	HTML	Chr	Dec	Hex	Oct	HTML	Chr	Dec	Hex	Oct	HTML	Chr
0	0	000	NULL	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SoH	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	SoTxt	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	EoTxt	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EoT	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	Enq	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	Ack	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	Bell	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	Bsp	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	HTab	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LFeed	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VTab	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FFeed	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SOut	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SIn	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAck	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	Syn	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	EoTB	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	Can	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EoM	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	Sub	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	Esc	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FSep	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GSep	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RSep	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	USep	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	Delete

charstable.com

The Decode: 87 101 108 99 111 109 101 32 116 111 32 67 114 121 112 116 111 32 73 115 108 97 110 100 33 33 33 32

Will be translate into a text: “Welcome to Cryptology Island!!! ”

Explain:

Base to the ASCII table

- 87 is W
- 101 is e
- 108 is l

  ……

- 33 is !

Task 7:

Result: UIT@Crypto2018

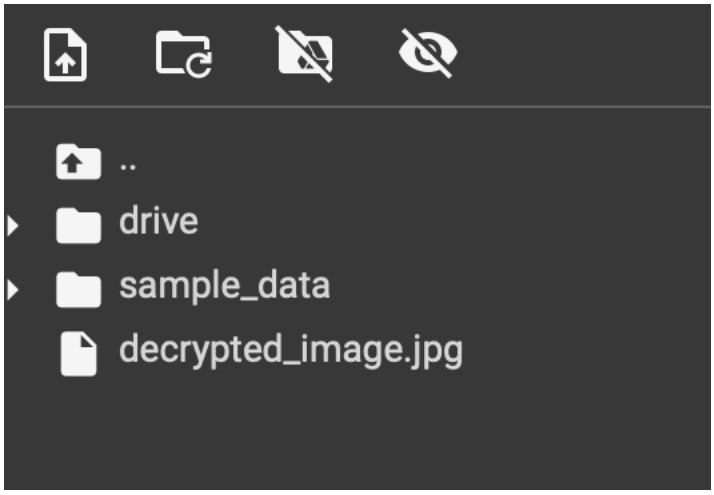
First, I converted the image to Notepad to search for the key. Then, I noticed a repeated sequence 'uithcm' many times, so I thought it might be the key. After that, I wrote code to convert it to the correct format.

Image decrypted:

This is my code:

```python
def xor_decrypt(ciphertext, key):
    # Hàm giải mã XOR
    decrypted = bytearray()
    for i in range(len(ciphertext)):
        decrypted.append(ciphertext[i] ^ key[i % len(key)])
    return bytes(decrypted)

def read_image(file_path):
    # Hàm đọc dữ liệu từ tệp ảnh
    with open(file_path, 'rb') as file:
        return bytearray(file.read())

def write_image(file_path, data):
    # Hàm ghi dữ liệu vào tệp ảnh
    with open(file_path, 'wb') as file:
        file.write(data)

# Thay thế 'crypto01.jpg' bằng đường dẫn đến tệp ảnh đã được mã hóa của bạn
image_path = r'/content/drive/MyDrive/img.jpg'

# Thay thế 'your_key' bằng khóa 6 ký tự được sử dụng cho quá trình mã hóa
encryption_key = b'uithcm'

# Đọc ảnh đã được mã hóa
encrypted_image = read_image(image_path)

# Giải mã ảnh
decrypted_image = xor_decrypt(encrypted_image, encryption_key)

# Lưu ảnh đã giải mã
write_image('decrypted_image.jpg', decrypted_image)
```

Save the decrypted image:



7.1

- **Atbash cipher code**

```
def atbash_cipher(text):
    result = ""
    for char in text:
        if char.isalpha():
            if char.islower():
                result += chr(219 - ord(char))
            else:
                result += chr(155 - ord(char))
        else:
            result += char
    return result
```

- **Example**

```
#ví dụ
message = "toi la nguyen phu hao sinh vien nam 3"
encrypted_message = atbash_cipher(message)

print("Original Message:", message)
print("Encrypted Message:", encrypted_message)
```

- **Result**

```
Original Message: toi la nguyen phu hao sinh vien nam 3
Encrypted Message: glr oz mtfbvm ksf szl hrms ervm mzn 3
```

Description: Atbash Cipher works by replacing each letter with its opposite letter in the alphabet. In the standard alphabet, if we place the letters 'a' to 'z' in reverse, then 'a' will be replaced by 'z', 'b' will be replaced by ' y', and vice versa.

HẾT