TTKS0600, Encryption Techniques and Systems, Lecture assignment 1

Lehosvuo Timo

Bordi Tuukka

1 Review questions

1.1
Confidentiality = Basically means that data should be accessible only to authorized personnel only

Integrity = data should not be able to be modified improperly.
This means modifying the data or deleting it

Availability = data should be accessible when needed

1.2
Passive attack = trying to learn or use information from the system. This does not effect the system resources. For example traffic analysis

Active attack = trying to change how the systems resources operate or alter them. For example DoS or modification of messages

1.3
Plaintext
Encryption algorithm
Secret key
Ciphertext
Decryption algorithm

1.4
suptitutions and transformations

1.5
1 if using symmetric cipher, 2 if using asymmetric cipher

1.6
block cipher processes one input block at a time producing one output
block per input block whereas stream cipher processes input elements continuously producing output element as it goes.

1.7

cryptanalysis and brute-force attack

1.8
1) Attacker knows: encryption algorithm and ciphertect
Type of attack: ciphertext only

2) Attacker knows: encryption algorithm, ciphertext, one or more plaintext-ciphertext pairs formed with the secret key
Type of attack: known plaintext

3) Attacker knows: encryption algorithm, ciphertext, plaintext message chosen by cryptanalyst together with its corresponding ciphertext generated with the secret key
Type of attack: Chosen plaintext

4) Attacker knows: encryption algorithm, ciphertext, ciphertext message chosen by cryptanalyst together with its corrseponding decrypted plaintext generated with the secret key
type of attack: chosen ciphertext

5) Attacker knows: encryption algorithm, ciphertext, plaintext message chosen by cryptanalyst together with its corresponding ciphertext generated with the secret key, ciphertext chosen by cryptanalyst together with its corrseponding decrypted plaintext generated with the secret key
Type of attack: chosen text

plus these two

6) Chosen-key attack: attacker has some knowledge between the relations between the different keyes, not very practical

7) Rubber-hose cryptanalysis: cryptanalyst threatens, blackmails or tortures someone until they give him the key. Bribery is sometimes referred as purchase-key attack. Powerful attacks, often the best

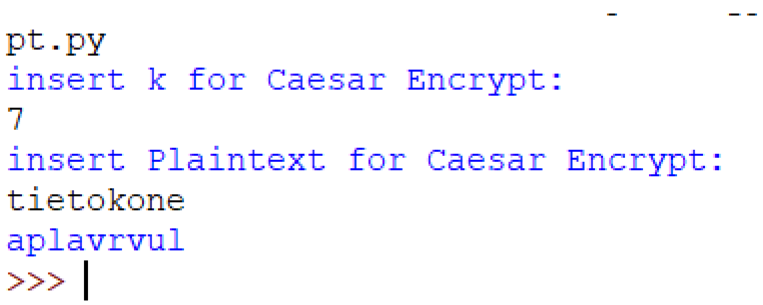
1.9
You need to make alot of random keys
    you might need millions of random characers on a reqular basis
mammoth key distribution problem
    for every message to be sent a key of equal length is needed by both sender and the receiver

1.10
steganography is a message hidden for example in a text in plain sight.
The words are hidden in the text and only the receiver knows what to look for.
For example the last words of the rows might form a message.
Message can be hidden in a picture as well, often in the least significant bit.

when a message is encrypted you can tell that the message is ecrypted but it is impossible to decode without the key. Steganography message can be very easy to decode but its existence can be hard to realize

2 Caesar Cipher

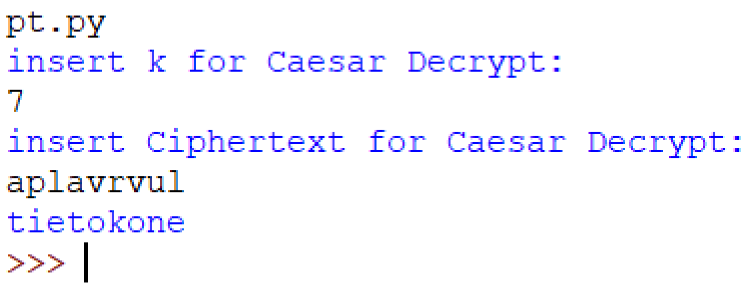
2.1 Encrypt tulokset



```
pt.py
insert k for Caesar Encrypt:
7
insert Plaintext for Caesar Encrypt:
tietokone
aplavrvul
>>> 
```

Figure 1:Encrypt

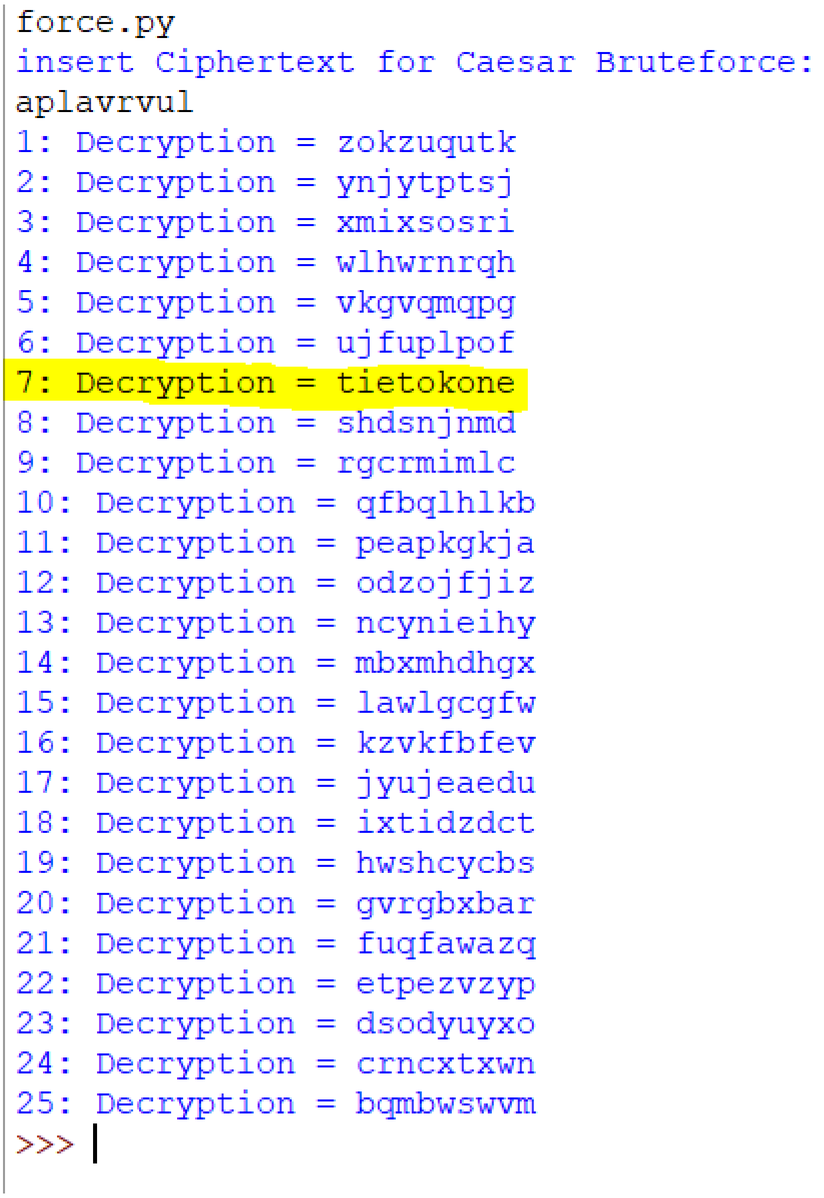
Decrypt tulokset



```
pt.py
insert k for Caesar Decrypt:
7
insert Ciphertext for Caesar Decrypt:
aplavrvul
tietokone
>>> 
```

Figure 2: Decrypt

2.2 Brute-forcen tulokset

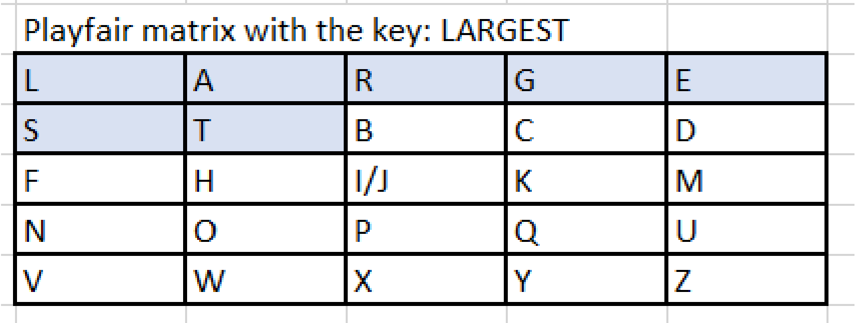


```
force.py
insert Ciphertext for Caesar Bruteforce:
aplavrvul
1:  Decryption = zokzuqutk
2:  Decryption = ynjytptsj
3:  Decryption = xmixsosri
4:  Decryption = wlhwrnrqh
5:  Decryption = vkgvqmqpg
6:  Decryption = ujfuplpof
7:  Decryption = tietokone
8:  Decryption = shdsnjnmd
9:  Decryption = rgcrmimlc
10:  Decryption = qfbqlhlkb
11:  Decryption = peapkgkja
12:  Decryption = odzojfjiz
13:  Decryption = ncynieihy
14:  Decryption = mbxmhdhgx
15:  Decryption = lawlgcgfw
16:  Decryption = kzvkfbfev
17:  Decryption = jyujeaedu
18:  Decryption = ixtidzdct
19:  Decryption = hwshcycbs
20:  Decryption = gvrgbxbar
21:  Decryption = fuqfawazq
22:  Decryption = etpezvzyp
23:  Decryption = dsodyuyxo
24:  Decryption = crncxtxwn
25:  Decryption = bqmbwswvm
>>> 
```

Figure 3: Brute-force

3 Playfair Code

3.1

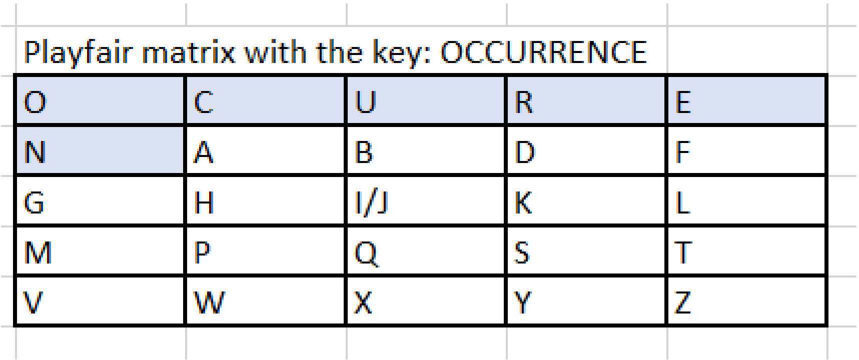


Playfair matrix with the key: LARGEST

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

Figure 4: key matrix with the key LARGEST

3.2

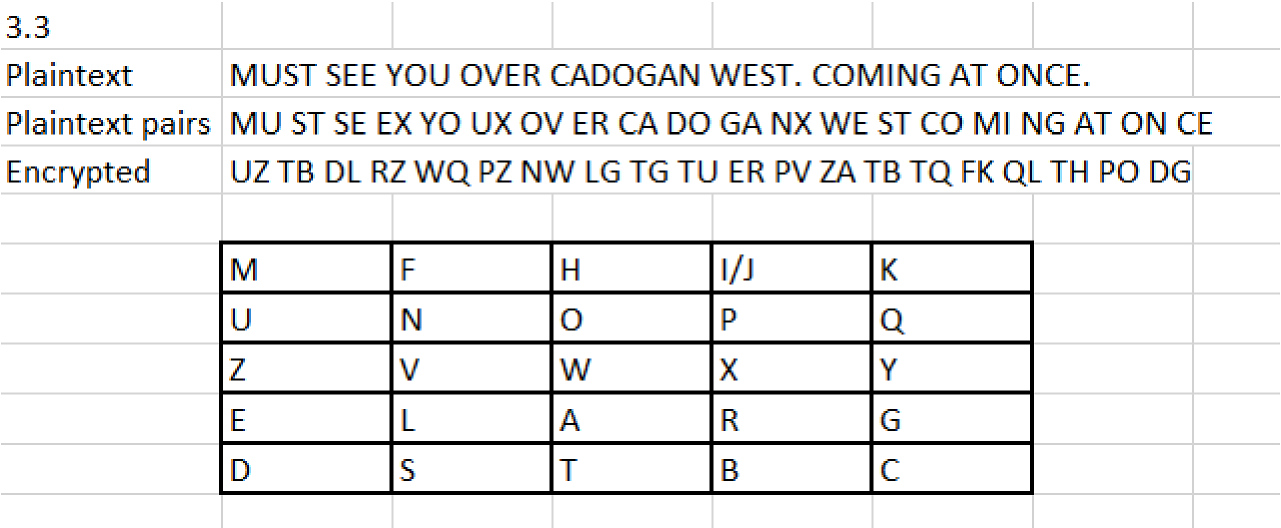


Playfair matrix with the key: OCCURRENCE

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

Figure 5: Key matrix with the key: OCCURRENCE

3.3

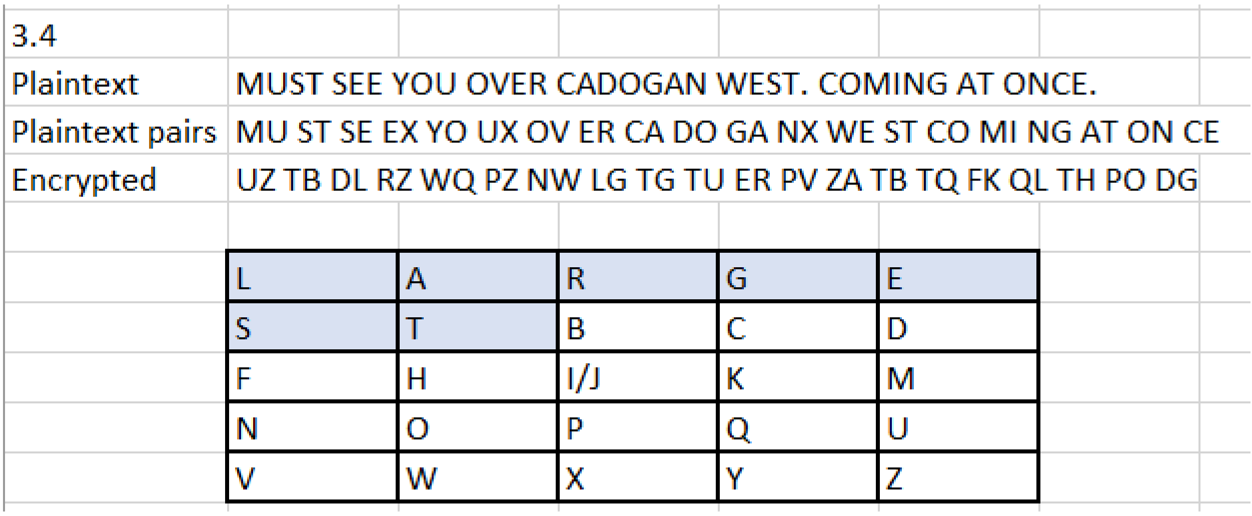


3.3	
Plaintext	MUST SEE YOU OVER CADOGAN WEST. COMING AT ONCE.
Plaintext pairs	MU ST SE EX YO UX OV ER CA DO GA NX WE ST CO MI NG AT ON CE
Encrypted	UZ TB DL RZ WQ PZ NW LG TG TU ER PV ZA TB TQ FK QL TH PO DG

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

Figure 6: Key matrix and an encrypted message

3.4

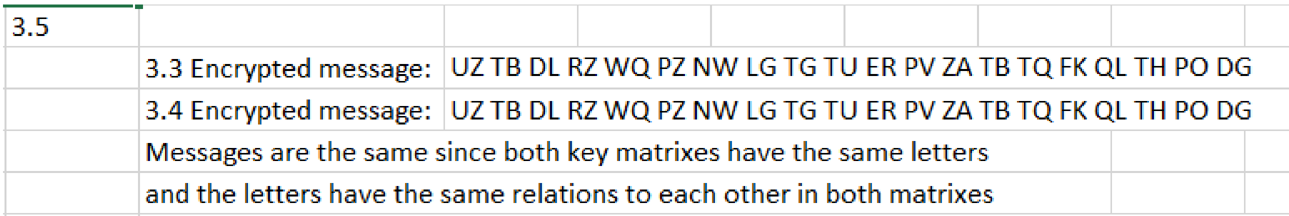


3.4	
Plaintext	MUST SEE YOU OVER CADOGAN WEST. COMING AT ONCE.
Plaintext pairs	MU ST SE EX YO UX OV ER CA DO GA NX WE ST CO MI NG AT ON CE
Encrypted	UZ TB DL RZ WQ PZ NW LG TG TU ER PV ZA TB TQ FK QL TH PO DG

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

Figure 7: Message encrypted using key matrix from 3.1

3.5



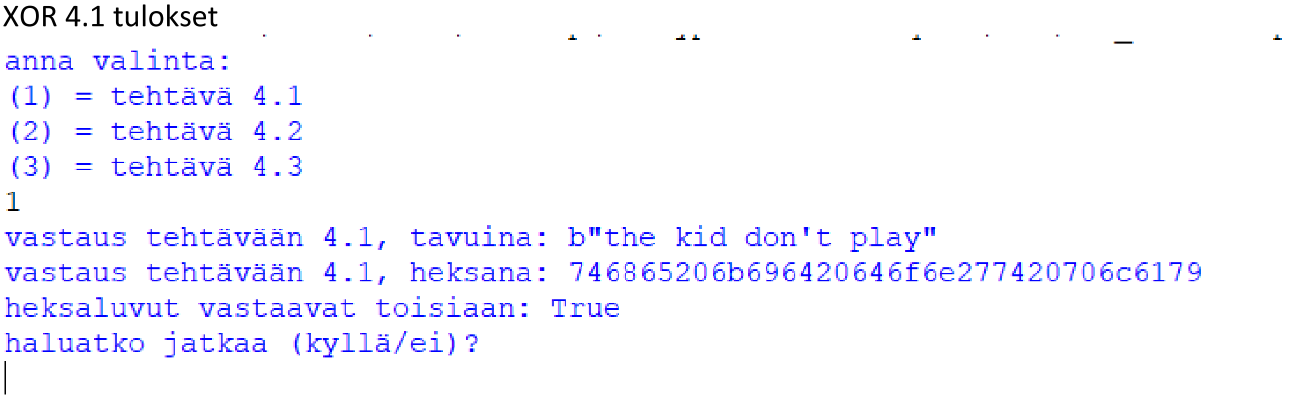
3.5		
	3.3 Encrypted message:	UZ TB DL RZ WQ PZ NW LG TG TU ER PV ZA TB TQ FK QL TH PO DG
	3.4 Encrypted message:	UZ TB DL RZ WQ PZ NW LG TG TU ER PV ZA TB TQ FK QL TH PO DG
	Messages are the same since both key matrixes have the same letters	
	and the letters have the same relations to each other in both matrixes	

Figure 8: Encrypted messages and an explanation

For example: E is next to L (in 3.4 these are in the edges of the matrix but because how the matrix is read it is basically the same as they were next to each other)
R is next to G
T is above H (in 3.3 these are in the edges of the matrix but because how the matrix is read it is basically the same as they were next to each other)
etc…

4 XOR

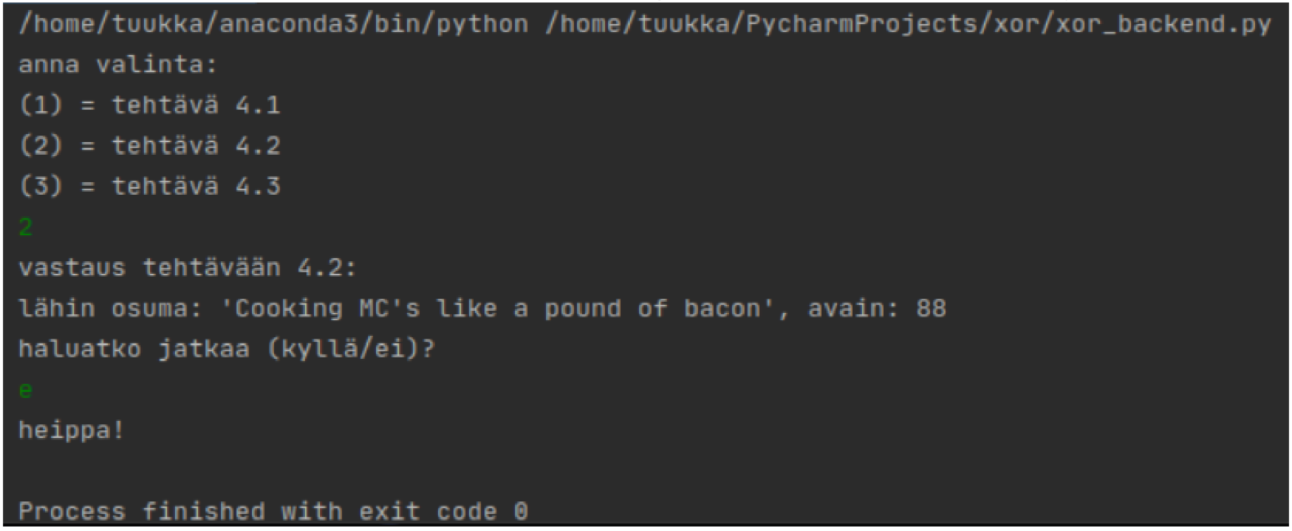
4.1 XOR 4.1 tulokset



```
anna valinta:
(1) = tehtävä 4.1
(2) = tehtävä 4.2
(3) = tehtävä 4.3
1
vastaus tehtävään 4.1, tavuina: b"the kid don't play"
vastaus tehtävään 4.1, heksana: 746865206b696420646f6e277420706c6179
heksaluvut vastaavat toisiaan: True
haluatko jatkaa (kyllä/ei)?
```

Figure 9: XOR 4.1

4.2 Tämä tehtävä ei toimi ilman että on asentanut langdetect moduulia. Testi on tehty linuxilla

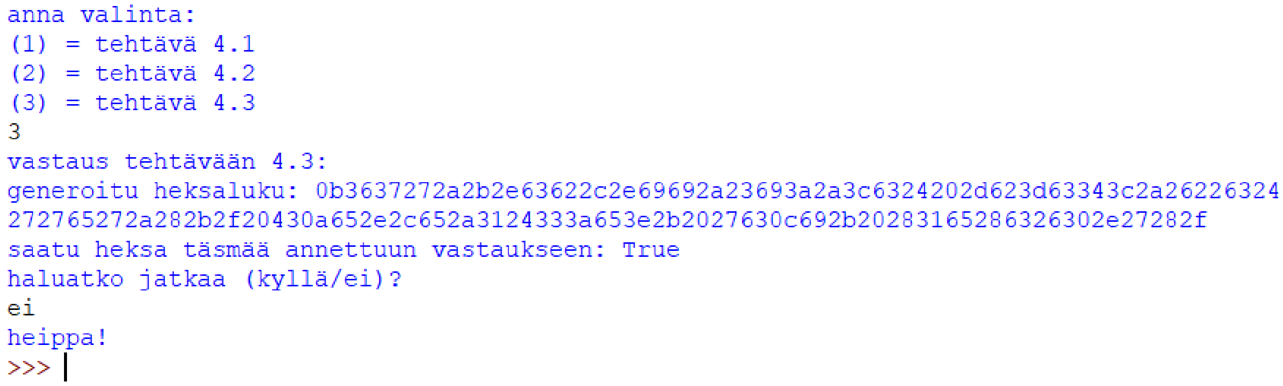


```
/home/tuukka/anaconda3/bin/python /home/tuukka/PycharmProjects/xor/xor_backend.py
anna valinta:
(1) = tehtävä 4.1
(2) = tehtävä 4.2
(3) = tehtävä 4.3
2
vastaus tehtävään 4.2:
lähin osuma: 'Cooking MC's like a pound of bacon', avain: 88
haluatko jatkaa (kyllä/ei)?
e
heippa!

Process finished with exit code 0
```

Figure 10: XOR 4.2

4.3 XOR 4.3 tulokset



```
anna valinta:
(1) = tehtävä 4.1
(2) = tehtävä 4.2
(3) = tehtävä 4.3
3
vastaus tehtävään 4.3:
generoitu heksaluku: 0b3637272a2b2e63622c2e69692a23693a2a3c6324202d623d63343c2a26226324
272765272a282b2f20430a652e2c652a3124333a653e2b2027630c692b20283165286326302e27282f
saatu heksa täsmää annettuun vastaukseen: True
haluatko jatkaa (kyllä/ei)?
ei
heippa!
>>> 
```

Figure 11: XOR 4.3

General comments about assignment

- Cipherin teko oli aikaa vievää ja siitä sai vähiten pisteitä ☹
- Playfair code oli helppo tehdä käsin. Koodaamalla menisi luultavasti huomattavasti kauemmin
- XOR oli haastava tehtävä ainakin pythonilla mutta hauska