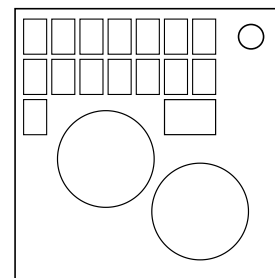


On the Subject of Mechanus Cipher

A cipher named after the plane of order, where a creature lurks to bring structure.

The module displays two rows of lettered panels, an extra lettered panel, a submit panel, and two lettered gears.



You can press any of the panels on a row to flip that row. The top row shows an encrypted word, a keyword and a binary string. The bottom row shows two sets of numbers and a second keyword. The extra panel shows a keyletter.

To solve the module, decrypt the word following the steps below and submit it using the gears. To submit, press the gears to turn them, and press the sub when you want to enter the selected letter. You can always flip the panels to clear your input.

Decrypting the word

Step 0 - Creating your key

For steps 2 and 3 you need a key. To make this key, take your keyword from the top row and repeat it until you have 26 letters. Then go off the following condition for all letters in the key in reading order: If a letter has already occurred before, shift the current letter forward in the alphabet until it isn't.

Example: TROUPER -> TROUPERTROUPERTROUPERTROUP -> TROUPE\$VWQXYFZABCDGHIJKLMN

Step 1 - Pancake Transposition

Take your encrypted word from the top row and the two number strings on the bottom row. Take the first number from both number strings and mirror the group of letters that are positioned between and including those two numbers. Repeat for the other numbers in the strings.

Example: EOISDYN, 1223236 2456565

EOISDYN 12 -> OEISDYN

OEISDYN 24 -> OSIEDYN

OSIEDYN 25 -> ODEISYN

ODEISYN 36 -> ODYSIEN

ODYSIEN 25 -> OISYDEN

OISYDEN 36 -> OIEDYSN

OIEDYSN 56 -> OIEDSYN

Step 2 - Binary Gear Cipher

Take your encrypted word from previous step, the key from step 0, the keyletter on the separate panel and the binary string on the top row. Make a column of your key and a column of decreasing 4-bit binary numbers starting from 1111. Start at 0000 on the binary column and start at your keyletter on the key column. Then shift up on both columns simultaneously, wrapping around if necessary. Shift up until you hit the first letter of your encrypted word and note down the binary. Repeat this for the other letters, starting from where you just ended. If your target letter is the same letter as the previous (also counts if the first letter is the same as the keyletter), go up on both columns until you hit it again. Append all these binary numbers into one big string and split it into four equal groups and set it up as a table.

Read the first column of the binary table from top to bottom. Starting from the place where you previously ended, shift up on both columns until you hit this binary. If the first digit of the binary string is a 1, shift up until you hit it again. The letter you land on is your new letter. Repeat for the other columns and binary digits.

Example: OIEDSYN, A, 1100000

T	1111
R	1110
O	1101
U	1100
P	1011
E	1010
S	1001
V	1000
W	0111
Q	0110
X	0101
Y	0100
F	0011
Z	0010
A	0001
B	0000
C	
D	
G	
H	
I	
J	
K	
L	
M	
N	

1	1	0	0	0	1	0
0	0	0	1	1	0	0
0	1	1	1	0	0	0
0	0	1	1	1	0	1

O -> 1100	(+1)1000 -> M
I -> 0100	(+1)1010 -> S
E -> 0011	(+0)0011 -> L
D -> 0001	(+0)0111 -> H
S -> 1100	(+0)0101 -> E
Y -> 0001	(+0)1000 -> O
N -> 1101	(+0)0001 -> H

Step 3 - Strangely Elusive Letter Cipher

Take your encrypted word from previous step, the key from step 0 and the keyword on the bottom row. Make a 3-dimensional 3×3×3 grid of your key. The remaining space will be labeled with a #. Look up the first letters of both the encrypted word and the keyword in this grid. If both positions match on an axis, your letter will lie on the same position on that axis. If they don't match on an axis, your letter will lie on the unused position on that axis. Your first letter will be the position found after evaluating all three axes. If this is a '#', it is the letter of the encrypted word instead. Repeat for the other letters

Example: MSLHEOH, CHIMERA

z=1			z=2			z=3		
T	R	O	Q	X	Y	G	H	I
U	P	E	F	Z	A	J	K	L
S	V	W	B	C	D	M	N	#

M+C -> W

S+H -> A

L+I -> # -> L

H+M -> L

E+E -> E

O+R -> T

H+A -> S