Bahan kuliah II4021 Kriptografi

04 - Kriptografi Klasik

(Bagian 2)

Oleh: Rinaldi M

Pogram Studi Sistem dan Teknologi Informasi

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung

2026

- Pada Bagian 2 dan 3 akan dibahas bebeapa *cipher* klasik popular lainnya:
    1. Vigenere Cipher
    2. Playfair Cipher
    3. Affine Cipher
    4. Hill Cipher
    5. Enigma Cipher

1. Vigènere Cipher

- Penemu cipher ini sebenarnya adalah Giovan Batista Belaso, kare
  ia menggambarkan pertama kali pada tahun 1553 seperti ditulis di
  dalam bukunya *La Cifra del Sig. Giovan Batista Belaso.*

- Namun, *cipher* ini disempurnakan dan dipublikasikan oleh diplomat (sekaligus seorang kriptologis) Perancis, Blaise de Vigènere pada abad 16 (tahun 1586).

- Pada abat ke-19, banyak orang yang mengira Vigenère adalah penemu cipher ini, sehingga dikenal luas sebagai *Vigenère Cipher*.

- *Cipher* ini berhasil dipecahkan oleh Babbage dan Kasiski pada pertengahan Abad 19 (akan dijelaskan pada materi selanjutnya).

- Kasiski menguraikan langkah-Langkah untuk menemukan panjang kunci (bukan huruf-huruf kuncikunci).

- *Vigènere Cipher* digunakan oleh Tentara Konfiderasi (*Confederate Army*) pada Perang Sipil Amerika (*American Civil war*).

- Perang Sipil terjadi setelah *Vigènere Cipher* berhasil dipecahkan.

- Vigenere Cipher termasuk ke dalam kelompok *cipher* abjad-majemuk (*polyalpabetic cipher*). Ini berbeda dengan cipher abjad-tunggal (*monolphabetic cipher*).

- Perbedaan *polyalphabetic cipher* dengan *monoalphabetic cipher*:
  - *Cipher* abjad-tunggal: satu kunci untuk semua huruf plainteks

    Contoh: Caesar cipher
  - *Cipher* abjad-majemuk: setiap huruf menggunakan kunci berbeda.

    Contoh: Vigenere cipher

- *Cipher* abjad-majemuk dibuat dari sejumlah *cipher* abjad-tunggal, masing-masing dengan kunci yang berbeda.

Bentuk umum *cipher* abjad-majemuk:

- Kunci:

  $K = k_1 k_2 \ldots k_m$ (ket: m = panjang kunci)

- Plainteks:

  $P = p_1 p_2 \ldots p_m p_{m+1} \ldots p_{2m} \ldots$

- Cipherteks:

  $$C = E_k(P) = f_{k_1}(p_1)\, f_{k_2}(p_2) \ldots f_{k_m}(p_m) f_{k_1}(p_{m+1}) f_{k_2}(p_{m+2}) \ldots f_{k_m}(p_{2m}) \ldots$$

  $f(.)$ adalah fungsi enkripsi pada cipher abjad-tunggal

- Untuk m = 1, *cipher*-nya ekivalen dengan *cipher* abjad-tunggal.

- Kembali *ke Vigenere Cipher*

- *Vigènere Cipher* menggunakan tabel *Vigènere* – dinamakan *Vigenere square,* untuk melakukan enkripsi dan dekripsi.

- Setiap baris *i* di dalam bujursangkar menyatakan huruf-huruf cipherteks yang diperoleh menggunakan *Caesar Cipher* dengan kunci *k* = *i*.

- Artinya, setiap baris *i* merupakan pergeseran huruf alfabet sejauh *i* ke kanan

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

Plaintext

Key

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

→ baris ke-0

→ baris ke-25

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

- Kunci adalah string: $K = k_1 k_2 \ldots k_m$

  k_i untuk $1 \leq i \leq m$ menyatakan huruf-huruf alfabet

- Jika panjang kunci lebih pendek daripada panjang plainteks, maka kunci diulang secara periodik.

- Misalkan panjang kunci m = 10, maka 10 huruf pertama plainteks dienkripsi dengan kunci K, setiap huruf ke-i menggunakan kunci k_i.

Contoh: kunci = `soreang` (dalam angka: 18, 14, 17, 4, 0, 13, 6)

Plainteks: `terimapesanangulaikambing`

Kunci: `soreangsoreangsoreangsore`

Ini artinya setiap huruf plainteks dienkripsi menggunakan *Caesar Cipher* dengan kunci *k* yang berbeda-beda

Untuk 8 karakter berikutnya, kembali menggunakan pola enkripsi yang sama.

- Enkripsi dilakukan dengan mencari titik potong huruf plainteks dengan huruf kunci:

Plainteks : **t**erimapesanangulaikambing
Kunci : **s**oreangsoreangsoreangsore
Cipherteks: **L**SIMMNVWGRRAAMMZRMKNSTWEK

Plainteks

KUNCI	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
a	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
b	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	A
c	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
d	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
e	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
f	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
g	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
h	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
i	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
j	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
l	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
m	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
n	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
o	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
p	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
r	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
s	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	**L**	M	N	O	P	Q	R
t	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
u	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
v	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
w	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
x	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
z	Z	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

Gambar 4.3 Enkripsi huruf T dengan kunci s

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

- Hasil enkripsi seluruhnya adalah sebagai berikut:

Plainteks : `terimapesanangulaikambing`
Kunci : `soreangsoreangsoreangsore`
Cipherteks : `LSIMMNVWGRRAAMMZRMKNSTWEK`

- Tanpa menggunakan *Vigenere Square* pun enkripsi tetap dapat dihitung secara *Caesar Cipher* dengan menjumlahkan plainteks p_j dengan kunci k_i dalam modulus 26:

Enkripsi: $c_j = E(p_j) = (p_j + k_i) \bmod 26$ (1)

Dekripsi: $p_j = D(c_j) = (c_j - k_i) \bmod 26$ (2)

Contoh:

(t + s) mod 26 = (19 + 18) mod 26 = 37 mod 26 = 11 = L

(e + o) mod 26 = (4 + 14) mod 26 = 18 mod 26 = 18 = S, dst

- Kelebihan Vigenere Cipher: huruf plainteks yang sama tidak selalu dienkripsi menjadi huruf cipheteks yang sama pula, bergantung huruf kunci yang digunakan.

  Contoh: pada contoh di atas, huruf plainteks `T` dapat dienkripsi menjadi **`L`** atau **`H`**, dan huruf cipherteks **`V`** dapat merepresentasikan huruf plainteks `H`, `I`, dan `X`

- Hal di atas merupakan karakteristik dari *cipher* abjad-majemuk: setiap huruf cipherteks dapat memiliki kemungkinan banyak huruf plainteks.

- Bandingkan dengan *cipher* abjad-tunggal, setiap huruf cipherteks selalu menggantikan huruf plainteks tertentu.

Plainteks:

Dinas Pendidikan Kota Ternate meminta kepada pihak sekolah dan orang tua siswa untuk jenjang pendidikan SD dan SMP se-Kota Ternate untuk melarang para siswa membawa permainan lato-lato yang sedang tren itu ke sekolah, karena akan mengganggu kegiatan belajar mengajar yang dinilai berbahaya sehingga mengantisipasi kecelakaan bagi anak di daerah itu.

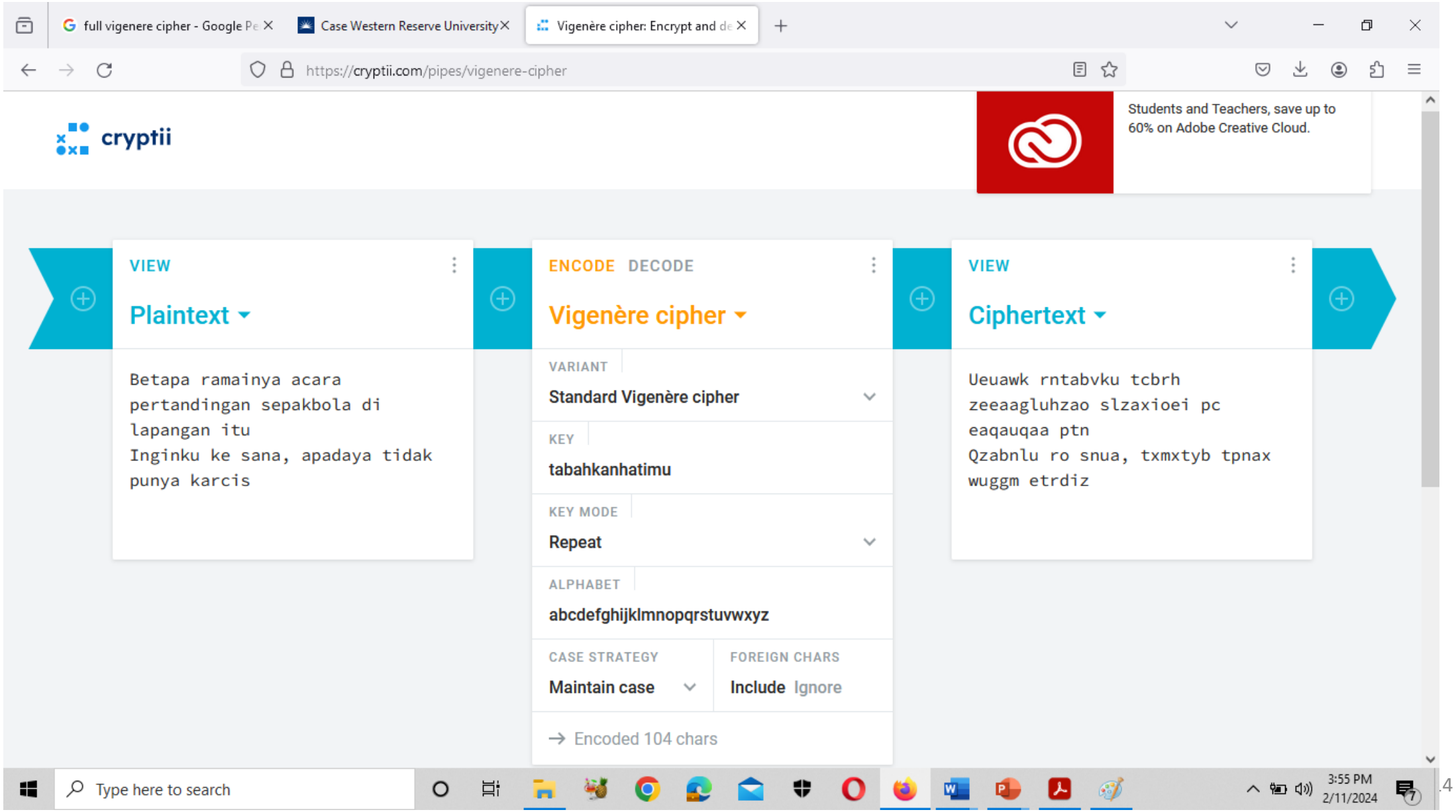
Kunci: selatsunda

Cipherteks:

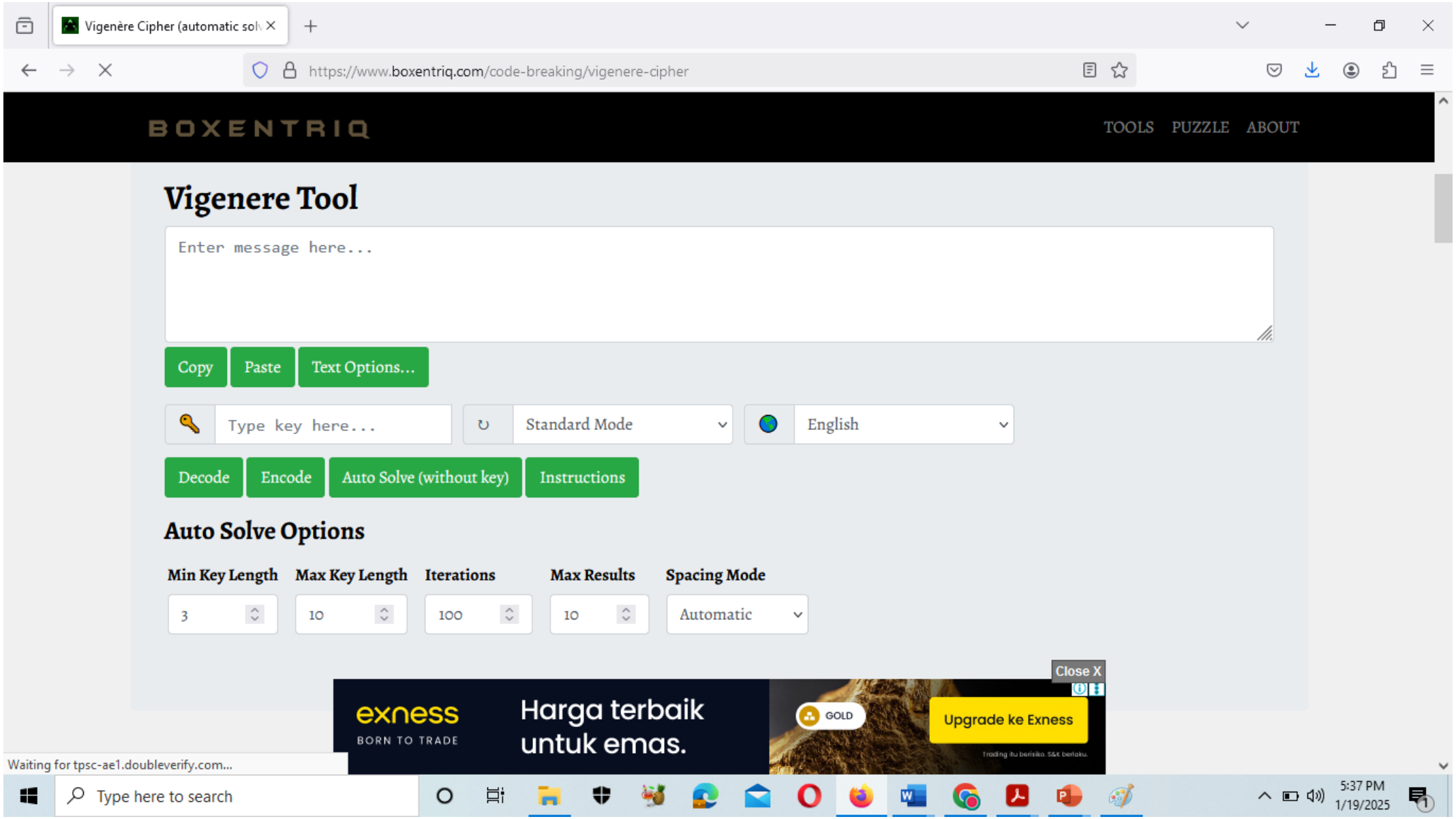
(dikelompokkan 4-huruf)

VMYAL HYAGI VMVAG CIGDT WVYAM WGRPI FXLKX HUQDP ALLKL WEBOA
ZHLNH JUAJT MEDIL OUHQT MOUEG BUAJP WROIW AENQS VHLNL EJFHK
GXLTX JHNWE MREUD EYYDR SRRPT JUFLS OEXEF TUJDP WVXAB FUAOA
LSWAM GSNQG KIOAG YNEHN AXFKX KYXRL SLVAK WHNDK SRXEG YANQG
YYVEZ AUGDN TIWAC SLZHN YEUAK QUAJD ARTLT AVRUB SLLYT KYULN YKLMX
FANQT AWTPT KCXHC WPLKT SHODG AEYAD VCQDE JESIM M

- Demo Vigenere Cipher online: https://cryptii.com/pipes/vigenere-cipher



https://www.boxentriq.com/code-breaking/vigenere-cipher



[7]:
```python
def Vigenere_cipher_encrypt(plainteks, kunci):
    cipherteks = ""
    kunci = kunci.lower()
    indeks_kunci = 0
    for char in plainteks:
        if char.isalpha():           # Hanya memproses huruf alfabet saja
            start = ord('A') if char.isupper() else ord('a')
            k = ord(kunci[indeks_kunci]) -  ord('a')
            c = (ord(char) - start + k) % 26   # Kodekan huruf ke angka 0 s/d 25, lalu enkripsi dengan Caesar Cipher
            c = c + start            # Kembalikan ke posisi semula
            cipherteks = cipherteks + chr(c)      # sambung setiap huruf cipherteks
            indeks_kunci = (indeks_kunci + 1) % len(kunci)
        else:
            cipherteks = cipherteks + char     # Pesan yang bukan huruf tidak dienkripsi, dibiarkan saja
    return cipherteks
```

[23]:
```python
def Vigenere_cipher_decrypt(cipherteks, kunci):
    plainteks = ""
    kunci = kunci.lower()
    indeks_kunci = 0
    for char in cipherteks:
        if char.isalpha():           # Hanya memproses huruf alfabet saja
            start = ord('A') if char.isupper() else ord('a')
            k = ord(kunci[indeks_kunci]) -  ord('a')
            c = (ord(char) - start - k) % 26   # Kodekan huruf ke angka 0 s/d 25, lalu enkripsi dengan Caesar Cipher
            c = c + start            # Kembalikan ke posisi semula
            plainteks = plainteks + chr(c)      # sambung setiap huruf plainteks
            indeks_kunci = (indeks_kunci + 1) % len(kunci)
        else:
            plainteks = plainteks + char     # Pesan yang bukan huruf tidak didekripsi, dibiarkan saja
    return plainteks
```

Vigenere Cipher dalam Python

[13]:
```
pesan = input("ketikkan pesan anda: ")
```

```
ketikkan pesan anda:  Sekarang tahun 2025, semoga kita sehat selalu, aamin!!
```

[5]:
```
kunci = input("kunci: ")
```

```
kunci:  apakabar
```

[21]:
```
cipherteks = Vigenere_cipher_encrypt(pesan, kunci)
```

[22]:
```
print(f"Pesan terenkripsi: {cipherteks}")
```

```
Pesan terenkripsi: Stkkrbnx tphen 2025, tedova uiua jewad sflrlj, akmjn!!
```

[24]:
```
plainteks = Vigenere_cipher_decrypt(cipherteks, kunci)
```

[25]:
```
print(f"Pesan hasil dekripsi: {plainteks}")
```

```
Pesan hasil dekripsi: Sekarang tahun 2025, semoga kita sehat selalu, aamin!!
```

Extended Vigenere Cipher

- Vigenere cipher untuk 26 huruf dapat dikembangkan untuk enkripsi 256 karakter ASCII

Enkripsi: $c_j = E(p_j) = (p_j + k_i) \bmod 256$

Dekripsi: $p_j = D(c_j) = (c_j - k_i) \bmod 256$

Latihan

- Ubahlah program Vigenere Cipher di atas sehingga dapat melakukan enkripsi pesan berupa file teks.

Varian *Vigenere Cipher*

Untuk mengatasi serangan dengan metode Kasiski, maka dibuat varian Vigenere Cipher sebagai berikut:

1. *Full Vigènere cipher*
   - Setiap baris di dalam tabel tidak menyatakan pergeseran huruf, tetapi merupakan permutasi huruf-huruf alfabet (lihat contoh table pada halaman berikut).
   - Tabel tersebut harus dirahasiakan.

   - Proses enkripsi dan dekripsi tetap sama seperti Vigenere standard:

Contoh sebuah full Vigènere square

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

2. *Auto-Key Vigènere cipher*

- Jika panjang kunci lebih kecil dari panjang plainteks, maka kunci disambung dengan plainteks tersebut.

- Misalnya,

  Pesan: `negara penghasil minyak mentah di dunia`

  Kunci: `INDO`

  maka kunci tersebut disambung dengan plainteks semula sehingga panjang kunci menjadi sama dengan panjang plainteks:

  Plainteks: `negarapenghasilminyakmentahdidunia`

  Kunci: `INDONEGARAPENGHASILMINYAKMENTAHDID`

  Cipherteks: `VRJOEEVEEGWEFOSMAVJMSZCNDMLQBDBQQD`

3. Running-Key Vigènere cipher

- Kunci adalah string yang sangat panjang yang diambil dari teks bermakna (misalnya naskah proklamasi, naskah Pembukaan UUD 1945, terjemahan ayat di dalam kitab suci, dan lain-lain).

- Misalnya,

  Pesan: `negarapenghasilminyakmentahdidunia`

  Kunci: `KERAKYATANYANGDIPIMPINOLEHHIKMATPE`

- Selanjutnya enkripsi dan dekripsi dilakukan seperti Vigenere cipher biasa.

2. Playfair Cipher

- Ditemukan oleh Sir Charles Wheatstone namun dipromosikan oleh Baron Lyon Playfair pada tahun 1854.

Sir Charles Wheatstone

Baron Lyon Playfair

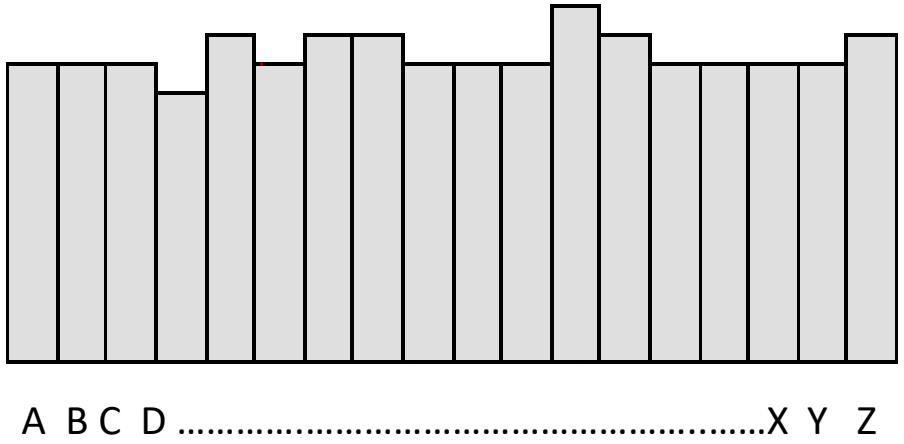
- Playfair cipher termasuk ke dalam kelompok *polygram cipher*

- Pada polygram cipher, satu blok huruf plainteks disubstitusi dengan satu blok huruf cipherteks.

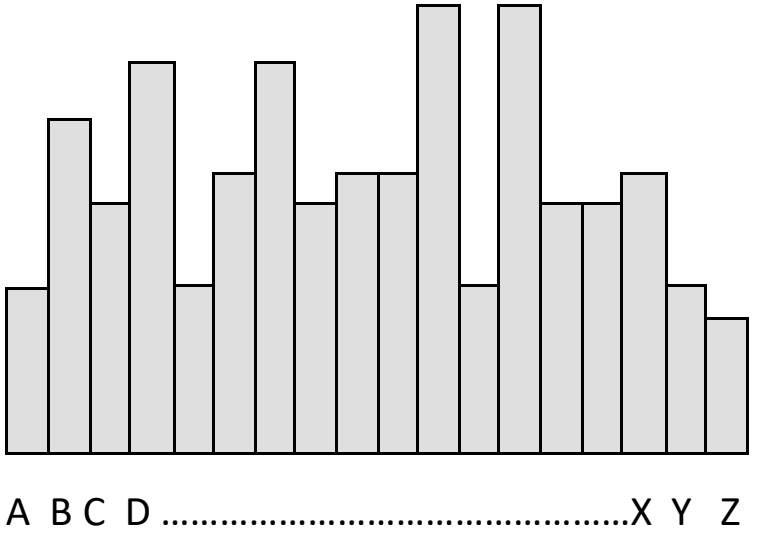
- Jika satu blok panjangnya 2 huruf, maka ia disebut digram (*bigram*), jika 3 huruf disebut trigram, dst. Playfair cipher melakukan enkripsi/dekripsi dalam bentuk bigra.

  Misalnya, bigram `AS` diganti dengan **`RT`**, `BY` diganti dengan **`SL`**

- Tujuannya: distribusi kemunculan huruf menjadi *flat* (datar), dan hal ini menyulitkan kriptanalisis dengan metode analisis frekuensi (akan dijelaskan pada materi kriptanalisis).



Flat histogram



Bukan flat histogram

- Kunci enkripsi/dekripsi pada Playfair cipher disusun berbentuk matriks berukuran 5 x 5
- Setiap elemen matriks adalah huruf-huruf alfabet A..Z
- Karena ada 25 elemen matriks, dan ada 26 huruf alfabet, maka ada satu huruf alfabet dibuang, yaitu huruf J

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

- Jumlah kemungkinan kunci yang bisa dibentuk:

  25!=15.511.210.043.330.985.984.000.000

Alasan mengapa huruf J dibuang:

1. Matriks Playfair hanya 5×5

Playfair cipher menggunakan tabel 5×5, jadi hanya bisa memuat 25 huruf. Sementara alfabet latin ada 26 huruf (A–Z). Artinya satu huruf harus “dikorbankan”.

2. I dan J dianggap mirip secara historis

Secara historis, dalam alfabet Latin lama, I dan J tidak dibedakan. J dianggap hanya variasi dari I. Karena itu, Playfair menggabungkan I dan J. Huruf J dihilangkan. Semua J dalam plaintext diganti menjadi I

Contoh: JALAN → IALAN

3. Dampaknya ke proses enkripsi/dekripsi

Saat enkripsi: setiap J diperlakukan sebagai I. Saat dekripsi: huruf I bisa berarti I atau J, tergantung konteks bahasa. Ini tidak mengurangi keamanan, karena Playfair sudah bersifat substitusi digraf

Ambiguitas kecil ini masih bisa diselesaikan lewat konteks

4. Apakah bisa buang huruf lain?

Bisa saja secara teori, tapi I/J paling masuk akal secara linguistik

Sudah jadi standar konvensi Playfair. Memudahkan implementasi dan pembacaan hasil

Sumber: ChatGPT

Kunci dapat dipilih dari sebuah kalimat yang mudah diingat, misalnya:

JALAN GANESHA SEPULUH

Buang huruf yang berulang dan huruf J jika ada:

ALNGESHPU

Lalu tambahkan huruf-huruf yang belum ada (kecuali J):

ALNGESHPUBCDFIKMOQRTVWXYZ

Masukkan ke dalam bujursangkar:

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

Pesan yang akan dienkripsi diatur terlebih dahulu sebagai berikut (langkah *preprocessing*):

1. Buang semua spasi di dalam pesan
2. Ganti huruf `j` (bila ada) dengan `i`. Contoh: jalan → ialan
3. Tulis pesan dalam pasangan huruf (*bigram*).
4. Jangan sampai ada bigram dengan pasangan huruf yang sama. Jika ada, sisipkan `x` di tengahnya
5. Jika jumlah huruf di dalam pesan ganjil, tambahkan huruf `x` pada bigram terakhir

Contoh plainteks: `temui ibu nanti malam`

→ Tidak ada huruf `j`, maka langsung tulis pesan dalam pasangan huruf (setelah semua spasi dibuang):

`te mu ii bu na nt im al am`

→ Ada bigram dengan huruf yang berulang (`ii`), sisipkan huruf `x` di tengahnya:

`te mu ix ib un an ti ma la m`

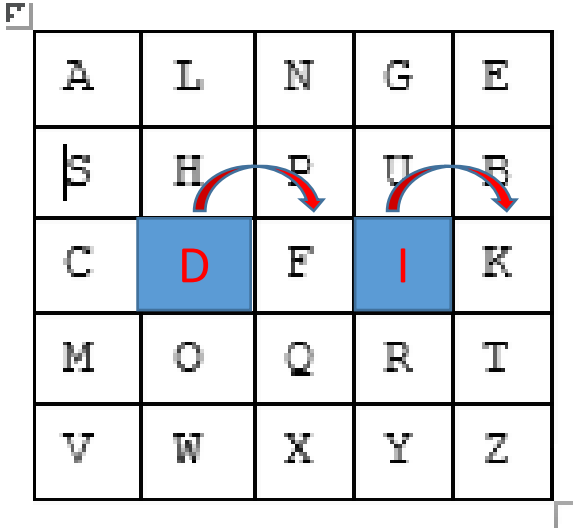
→ Tambahkan huruf `x` jika bigram terakhir hanya satu huruf:

`te mu ix ib un an ti ma la mx`

Algoritma enkripsi:

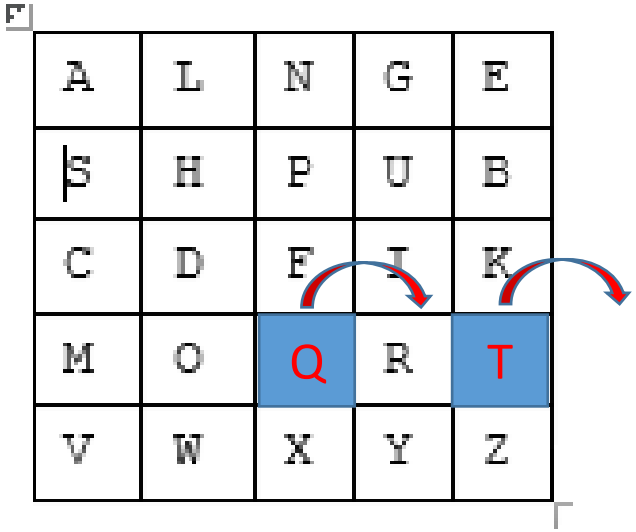
1. Jika dua huruf terdapat pada baris kunci yang sama maka tiap huruf diganti dengan huruf di kanannya (bersifat siklik).

Bigram: `di`



Cipherteks: `FK`

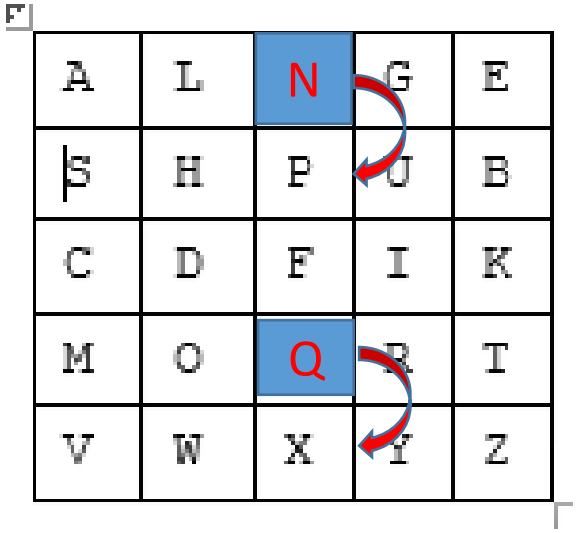
Bigram: `qt`



Cipherteks: `RM`

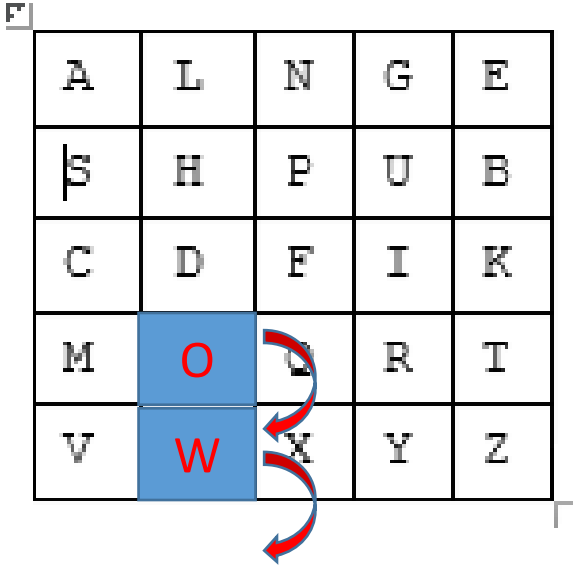
2. Jika dua huruf terdapat pada kolom kunci yang sama maka tiap huruf diganti dengan huruf di bawahnya (bersifat siklik).

Bigram: `nq`



Cipherteks: `PX`

Bigram: `ow`

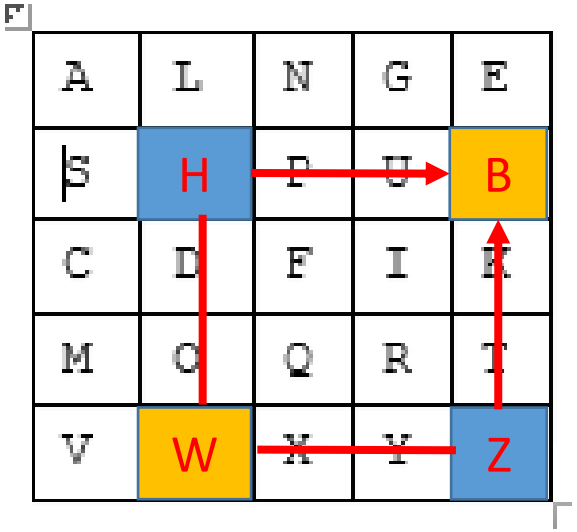


Cipherteks: `WL`

3. Jika dua huruf tidak pada baris yang sama atau kolom yang sama, maka:

- huruf pertama diganti dengan huruf pada perpotongan baris huruf pertama dengan kolom huruf kedua.
- huruf kedua diganti dengan huruf pada titik sudut keempat dari persegi panjang yang dibentuk dari tiga huruf yang digunakan sampai sejauh ini.

Bigram: `hz`



Cipherteks: `BW`

Plainteks: temui ibu nanti malam

Bigram: te mu ix ib un an ti ma la mx

Kunci:

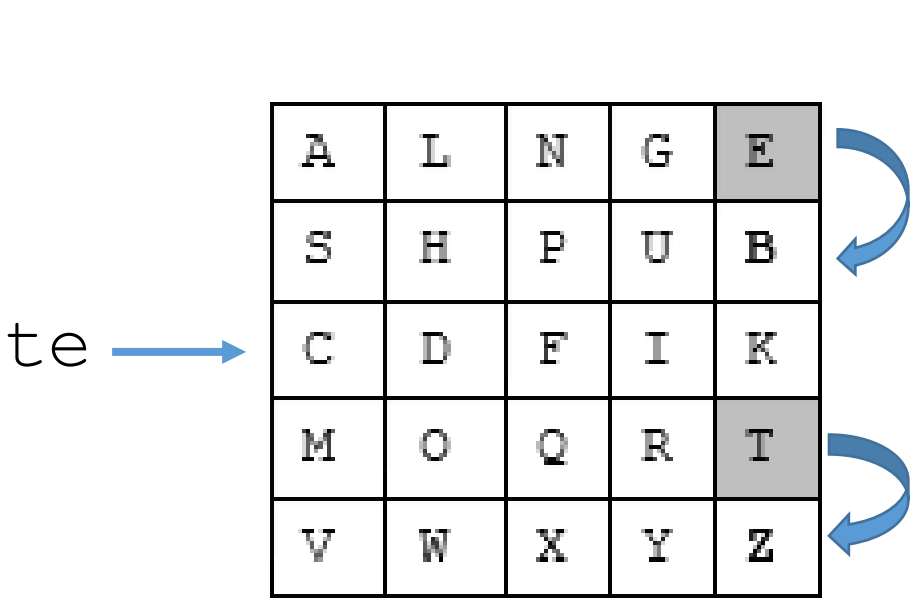
A	L	N	G	E
S	H	P	U	B
C	D	F	I	K
M	O	Q	R	T
V	W	X	Y	Z

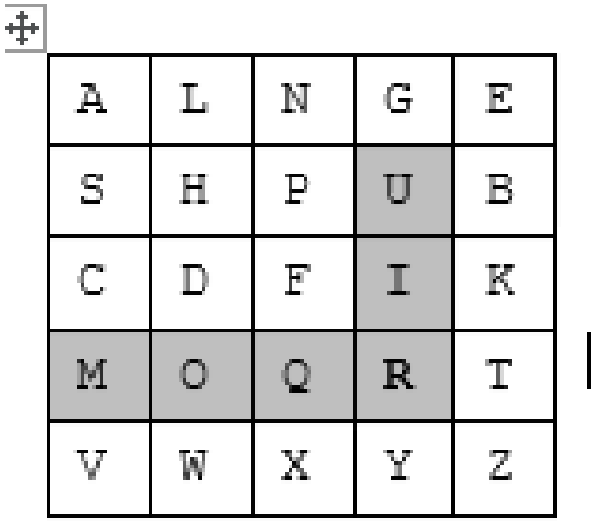
Cipherteks: ZB RS FY KU PG LG RK VS NL QV

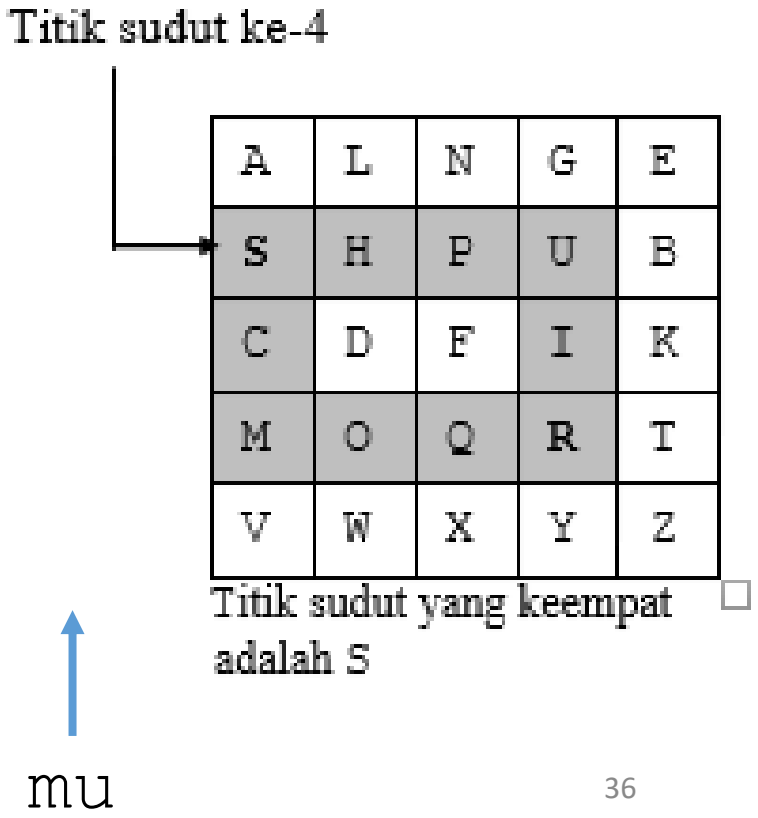
Cara enkripsinya sebagai berkut:

Bigram: te mu ix ib un an ti ma la mx

Cipherteks: ZB RS FY KU PG LG RK VS NL QV



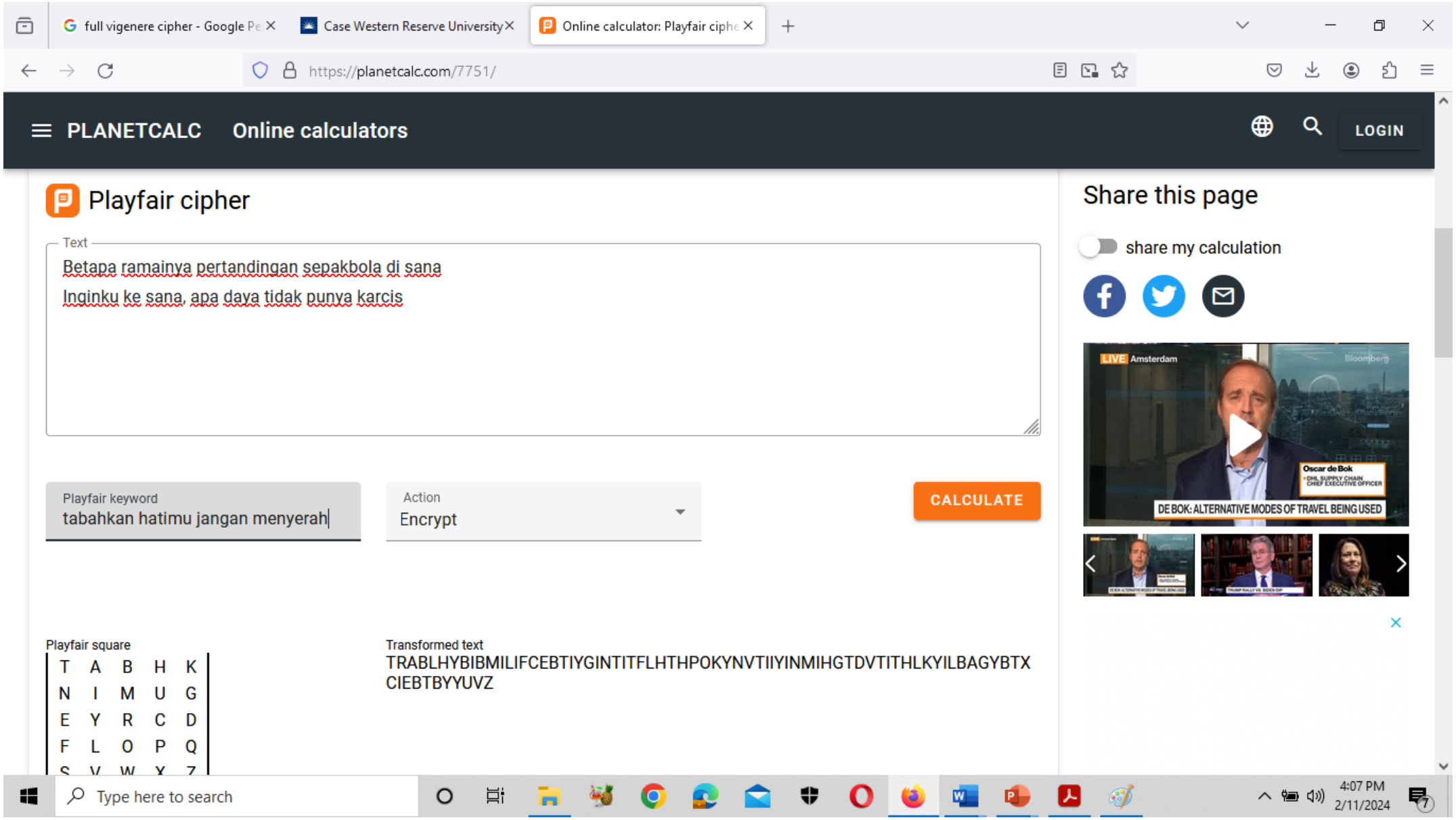




Algoritma dekripsi kebalikan dari algoritma enkripsi. Langkah-langkahnya adalah sebagai berikut:

1. Jika dua huruf terdapat pada baris bujursangkar yang sama maka tiap huruf diganti dengan huruf di kirinya.
2. Jika dua huruf terdapat pada kolom bujursangkar yang sama maka tiap huruf diganti dengan huruf di atasnya.
3. Jika dua huruf tidak pada baris yang sama atau kolom yang sama, maka huruf pertama diganti dengan huruf pada perpotongan baris huruf pertama dengan kolom huruf kedua. Huruf kedua diganti dengan huruf pada titik sudut keempat dari persegi panjang yang dibentuk dari tiga huruf yang digunakan sampai sejauh ini.
4. Buanglah huruf X yang tidak mengandung makna.

Demo Playfair cipher online: https://planetcalc.com/7751/



3. Affine Cipher

- Merupakan perluasan *Caesar cipher*

  Enkripsi: $C \equiv mP + b \pmod{n}$

  Dekripsi: $P \equiv m^{-1}(C - b) \pmod{n}$

- Kunci: *m* dan *b*

Keterangan:

1. *n* adalah ukuran alfabet
2. *m* bilangan bulat yang relatif prima dengan *n*
3. *b* adalah jumlah pergeseran
4. *Caesar cipher* adalah khusus dari *affine cipher* dengan $m = 1$
5. m^{-1} adalah inversi *m* (mod *n*), yaitu $m \cdot m^{-1} \equiv 1 \pmod{n}$

- Contoh:

  Plainteks: `kripto` (10 17 8 15 19 14)

  n = 26, ambil m = 7 (7 relatif prima dengan 26)

  Enkripsi: $C \equiv 7P + 10 \pmod{26}$

  $p_1 = 10 \rightarrow c_1 \equiv 7 \cdot 10 + 10 \equiv 80 \equiv 2 \pmod{26}$ (huruf 'C')

  $p_2 = 17 \rightarrow c_2 \equiv 7 \cdot 17 + 10 \equiv 129 \equiv 25 \pmod{26}$ (huruf 'Z')

  $p_3 = 8 \rightarrow c_3 \equiv 7 \cdot 8 + 10 \equiv 66 \equiv 14 \pmod{26}$ (huruf 'O')

  $p_4 = 15 \rightarrow c_4 \equiv 7 \cdot 15 + 10 \equiv 115 \equiv 11 \pmod{26}$ (huruf 'L')

  $p_5 = 19 \rightarrow c_5 \equiv 7 \cdot 19 + 10 \equiv 143 \equiv 13 \pmod{26}$ (huruf 'N')

  $p_6 = 14 \rightarrow c_6 \equiv 7 \cdot 14 + 10 \equiv 108 \equiv 4 \pmod{26}$ (huruf 'E')

  Cipherteks: `CZOLNE`

- Dekripsi:

  - Mula-mula hitung m^{-1} yaitu $7^{-1} \pmod{26}$ dengan memecahkan $7x \equiv 1 \pmod{26}$

    Solusinya: $x \equiv 15 \pmod{26}$ sebab $7 \cdot 15 = 105 \equiv 1 \pmod{26}$.

  - Jadi, $P \equiv 15\,(C - 10) \pmod{26}$

$c_1 = 2 \quad \rightarrow p_1 \equiv 15 \cdot (2 - 10) = -120 \equiv 10 \pmod{26}$ (huruf ‘k’)

$c_2 = 25 \rightarrow p_2 \equiv 15 \cdot (25 - 10) = 225 \equiv 17 \pmod{26}$ (huruf ‘r’)

$c_3 = 14 \rightarrow p_3 \equiv 15 \cdot (14 - 10) = 60 \equiv 8 \pmod{26}$ (huruf ‘i’)

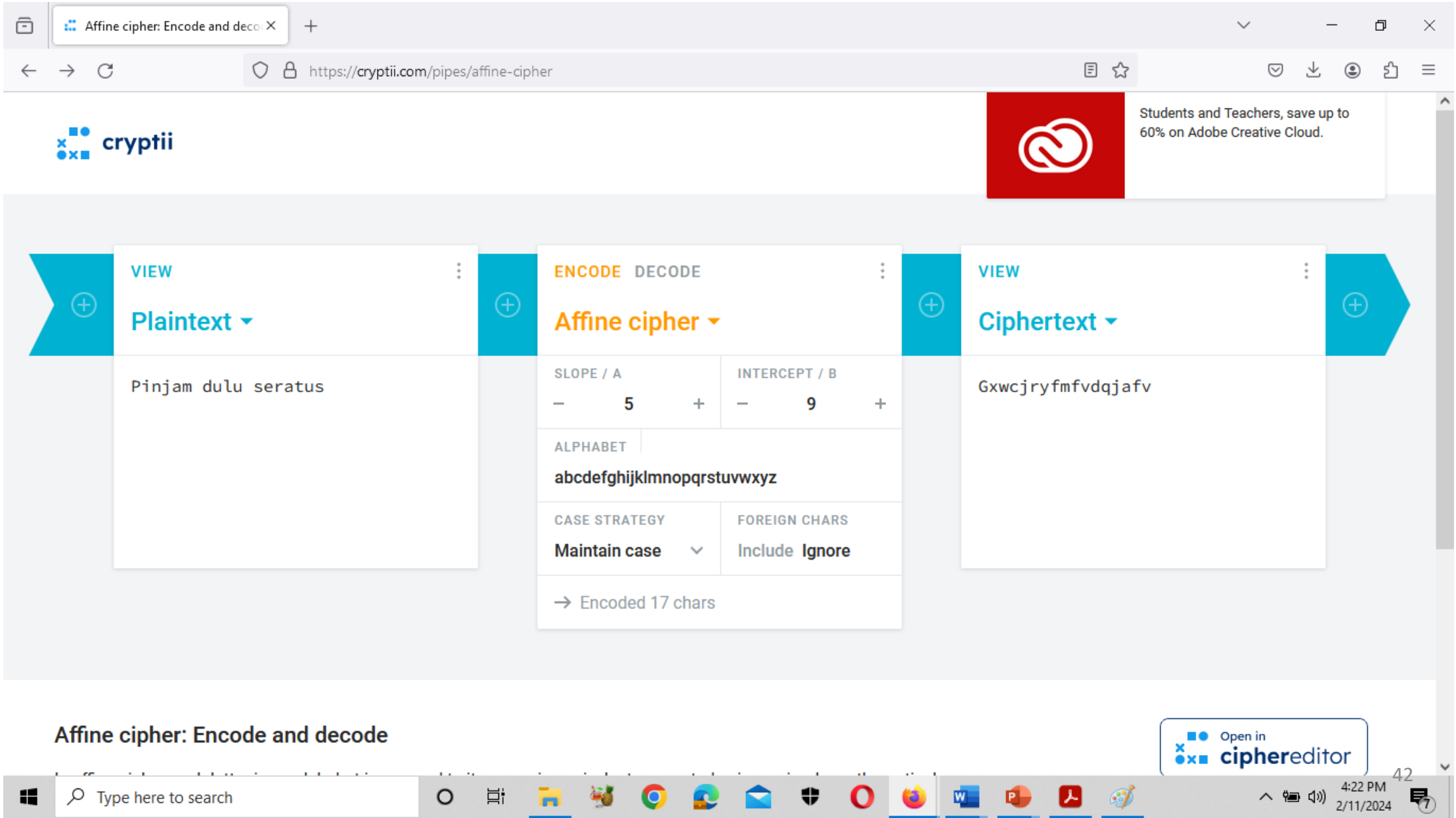
$c_4 = 11 \rightarrow p_4 \equiv 15 \cdot (11 - 10) = 15 \equiv 15 \pmod{26}$ (huruf ‘p’)

$c_5 = 13 \rightarrow p_5 \equiv 15 \cdot (13 - 10) = 45 \equiv 19 \pmod{26}$ (huruf ‘t’)

$c_6 = 4 \quad \rightarrow p_6 \equiv 15 \cdot (4 - 10) = -90 \equiv 14 \pmod{26}$ (huruf ‘o’)

Plainteks yang diungkap kembali: `kripto`

Demo affine cipher online: https://cryptii.com/pipes/affine-cipher



- *Affine cipher* tidak aman, karena kunci mudah ditemukan dengan *exhaustive search*,

- sebab ada 25 pilihan untuk *b* dan 12 buah nilai *m* yang relatif prima dengan 26 (yaitu 1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, dan 25).

- Salah satu cara memperbesar faktor kerja untuk *exhaustive key search:* enkripsi tidak dilakukan terhadap huruf individual, tetapi dalam blok huruf.

- Misal, pesan `kriptografi` dipecah menjadi kelompok 4-huruf:

  `krip togr afi`

  (ekivalen dengan 10170815 19140617 000508, dengan memisalkan

  'A' = 0, 'B' = 1, …, 'Z' = 25)

- Nilai terbesar yang dapat muncul untuk merepresentasikan blok: 25252525 (ZZZZ), maka 25252525 dapat digunakan sebagai modulus *n*.
- Nilai *m* yang relatif prima dengan 25252525, misalnya 21035433,
- *b* dipilih antara 1 dan 25252525, misalnya 23210025.

- Fungsi enkripsi menjadi:

$$C \equiv 21035433P + 23210025 \pmod{25252525}$$

- Fungsi dekripsi, setelah dihitung, menjadi

$$P \equiv 5174971\,(C - 23210025) \pmod{25252525}$$

4. Hill Cipher

- Dikembangkan oleh Lester Hill (1929), berbasis aljabar linier
- Menggunakan *m* buah persamaan linier
- Untuk *m* = 3 (enkripsi setiap 3 huruf), system persamaan linernya menjadi:

$$C_1 = (k_{11}\, p_1 + k_{12} p_2 + k_{13}\, p_3) \bmod 26$$
$$C_2 = (k_{21}\, p_1 + k_{22} p_2 + k_{23}\, p_3) \bmod 26$$
$$C_3 = (k_{31}\, p_1 + k_{32} p_2 + k_{33}\, p_3) \bmod 26$$

$$\begin{pmatrix} C_1 \\ C_2 \\ C_3 \end{pmatrix} = \begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix}$$

C = KP

- Contoh:

$$\mathbf{K} = \begin{pmatrix} 17 & 17 & 5 \\ 21 & 18 & 21 \\ 2 & 2 & 19 \end{pmatrix}$$

Plainteks: `paymoremoney`

Enkripsi tiga huruf pertama: `pay = (15, 0, 24)`

Cipherteks: $C = \begin{pmatrix} 17 & 17 & 5 \\ 21 & 18 & 21 \\ 2 & 2 & 19 \end{pmatrix} \begin{pmatrix} 15 \\ 0 \\ 24 \end{pmatrix} = \begin{pmatrix} 375 \\ 819 \\ 486 \end{pmatrix} \bmod 26 = \begin{pmatrix} 11 \\ 13 \\ 18 \end{pmatrix}$ = `LNS`

Cipherteks selengkapnya: `LNSHDLEWMTRW`

- Dekripsi perlu menghitung $\mathbf{K}^{-1}$ sedemikian sehingga $\mathbf{K}\mathbf{K}^{-1} = \mathbf{I}$ (**I** matriks identitas).

$$\mathbf{K}^{-1} = \begin{pmatrix} 4 & 9 & 15 \\ 15 & 17 & 6 \\ 24 & 0 & 17 \end{pmatrix}$$

sebab

$$\begin{pmatrix} 17 & 17 & 5 \\ 21 & 18 & 21 \\ 2 & 2 & 19 \end{pmatrix} \begin{pmatrix} 4 & 9 & 15 \\ 15 & 17 & 6 \\ 24 & 0 & 17 \end{pmatrix} = \begin{pmatrix} 443 & 442 & 442 \\ 858 & 495 & 780 \\ 494 & 52 & 365 \end{pmatrix} \bmod 26 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- Cara menghitung matriks invers 2 x 2:

$$K = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \qquad K^{-1} = \frac{1}{\det(K)} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

$$= \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

Contoh: $K = \begin{pmatrix} 3 & 10 \\ 15 & 9 \end{pmatrix}$

det(K) = (3)(9) – (15)(10) = 27 – 150 = –123 mod 26 = 7

$$\mathbf{K}^{-1} = \frac{1}{7}\begin{pmatrix} 3 & 10 \\ 15 & 9 \end{pmatrix} = 7^{-1}\begin{pmatrix} 9 & -10 \\ -15 & 3 \end{pmatrix}$$

$$= 15\begin{pmatrix} 9 & -10 \\ -15 & 3 \end{pmatrix} = 15\begin{pmatrix} 9 & 16 \\ 11 & 3 \end{pmatrix} = \begin{pmatrix} 135 & 240 \\ 165 & 45 \end{pmatrix} \bmod 26 = \begin{pmatrix} 5 & 6 \\ 9 & 19 \end{pmatrix}$$

Keterangan (ingat kembali teori bilangan di dalam Matematika Diskrit):

(i) $7^{-1} \pmod{26} \equiv 15$, karena (7)(15) = 105 mod 26 = 1

(ii) $-10 \equiv 16 \pmod{26}$

(iii) $-15 \equiv 11 \pmod{26}$)

Periksa bahwa:

$$\mathbf{K}\mathbf{K}^{-1} = \begin{pmatrix} 3 & 10 \\ 15 & 9 \end{pmatrix}\begin{pmatrix} 5 & 6 \\ 9 & 19 \end{pmatrix} = \begin{pmatrix} 3\cdot 5+10\cdot 9 & 3\cdot 6+10\cdot 19 \\ 15\cdot 5+9\cdot 9 & 15\cdot 6+9\cdot 19 \end{pmatrix}$$

$$= \begin{pmatrix} 105 & 208 \\ 156 & 261 \end{pmatrix} \bmod 26 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

- Untuk matriks 3 x 3, matriks dapat dicari dengan metode Eliminasi Gauss-Jordan, atau dihitung langsung dengan rumus berikut::

$$\mathbf{K} = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}, \quad \mathbf{K}^{-1} = \frac{1}{\det(\mathbf{K})} \begin{pmatrix} A & B & C \\ D & E & F \\ G & H & I \end{pmatrix}^T = \frac{1}{\det(\mathbf{K})} \begin{pmatrix} A & D & G \\ B & E & H \\ C & F & I \end{pmatrix}$$

- yang dalam hal ini,

$A = (ei - hf)$ $B = -(di - fg)$ $C = (dh - eg)$

$D = -(bi - hc)$ $E = (ai - cg)$ $F = -(ah - bg)$

$G = (bf - ec)$ $H = -(af - cd)$ $I = (ae - bd)$

dan

$$\det(\mathbf{K}) = aA + bB + cC$$

- Dekripsi:

$$\mathbf{P} = \mathbf{K}^{-1}\,\mathbf{C}$$

Cipherteks: LNS atau C = (11, 13, 18)

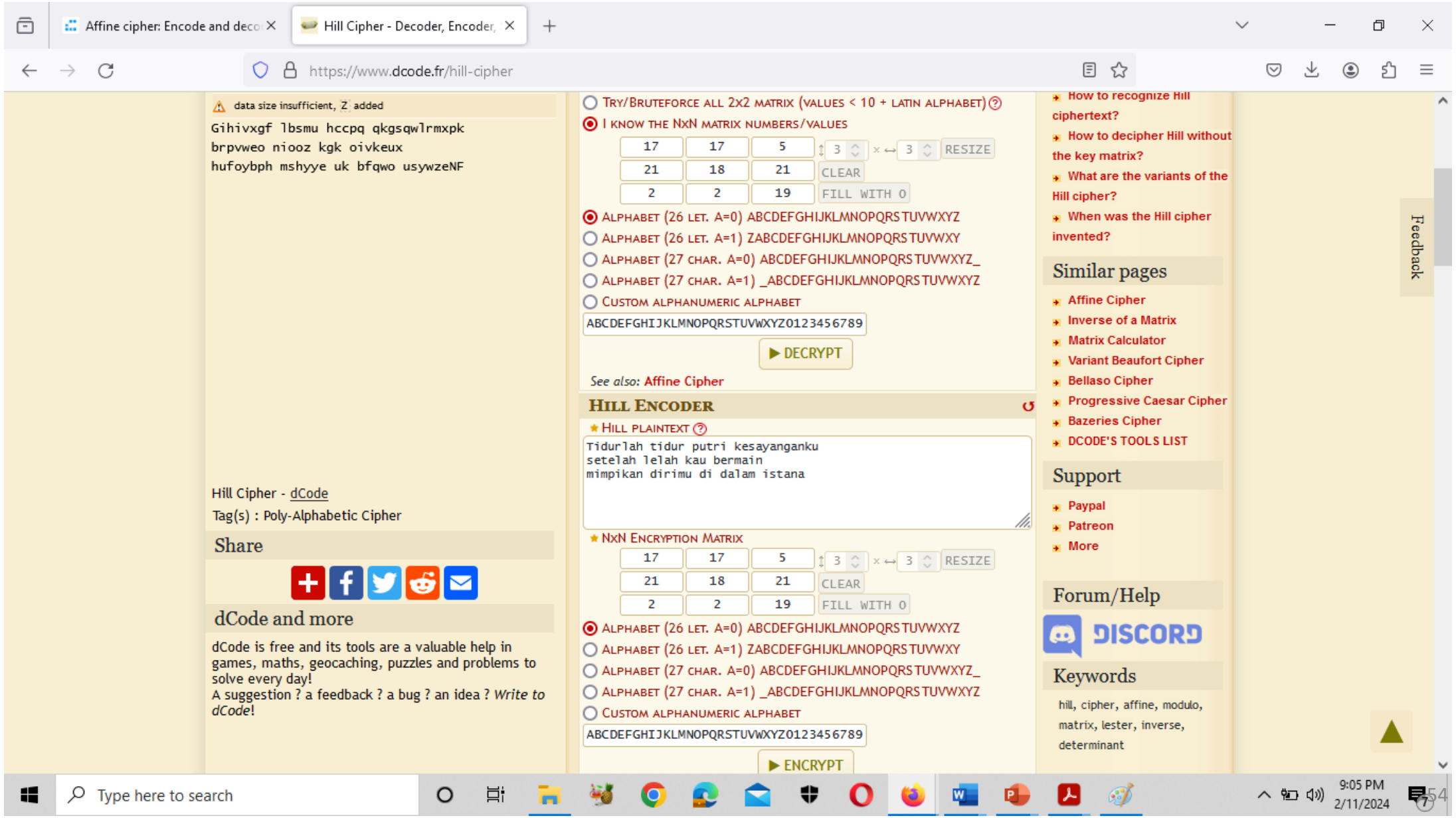
Plainteks:
$$\begin{pmatrix} 4 & 9 & 15 \\ 15 & 17 & 6 \\ 24 & 0 & 17 \end{pmatrix} \begin{pmatrix} 11 \\ 13 \\ 18 \end{pmatrix} = \begin{pmatrix} 431 \\ 494 \\ 570 \end{pmatrix} \bmod 26 = \begin{pmatrix} 15 \\ 0 \\ 24 \end{pmatrix}$$

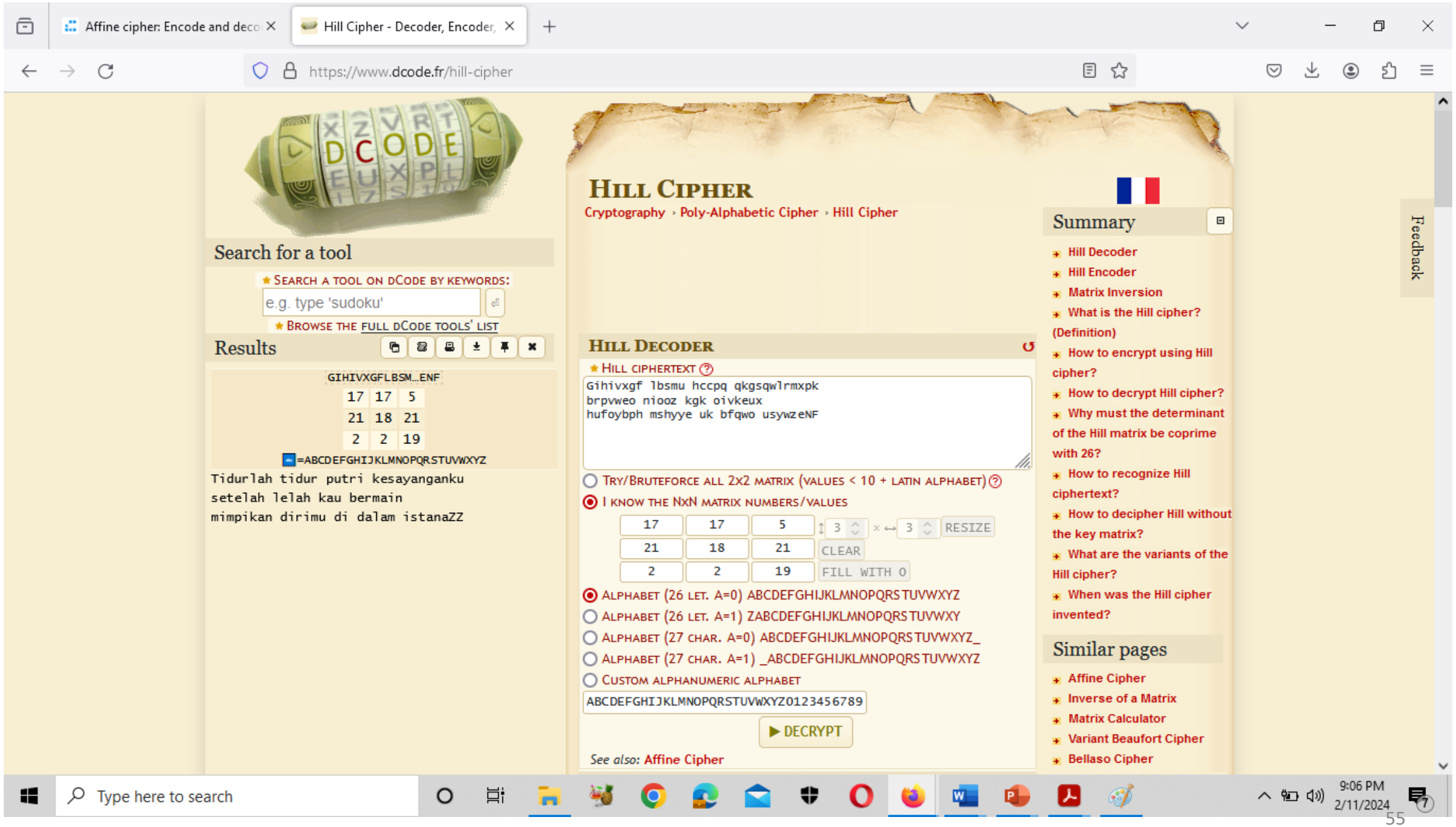
C = (15, 0, 24) = (P, A, Y)

- Kekuatan Hill cipher terletak pada penyembunyian frekuensi huruf tunggal

- Huruf plainteks yang sama belum tentu dienkripsi menjadi huruf cipherteks yang sama.

Demo Hill Cipher online: https://www.dcode.fr/hill-cipher




Affine cipher: Encode and deco
Hill Cipher - Decoder, Encoder,
https://www.dcode.fr/hill-cipher
Search for a tool
SEARCH A TOOL ON DCODE BY KEYWORDS:
e.g. type 'sudoku'
BROWSE THE FULL DCODE TOOLS' LIST
Results
GIHIVXGFLBSM...ENF
17 17 5
21 18 21
2 2 19
=ABCDEFGHIJKLMNOPQRSTUVWXYZ
Tidurlah tidur putri kesayanganku
setelah lelah kau bermain
mimpikan dirimu di dalam istanaZZ
HILL CIPHER
Cryptography › Poly-Alphabetic Cipher › Hill Cipher
HILL DECODER
HILL CIPHERTEXT
Gihivxgf lbsmu hccpq qkgsqwlrmxpk
brpvweo niooz kgk oivkeux
hufoybph mshyye uk bfqwo usywzeNF
TRY/BRUTEFORCE ALL 2X2 MATRIX (VALUES < 10 + LATIN ALPHABET)
I KNOW THE NXN MATRIX NUMBERS/VALUES
17 17 5
21 18 21
2 2 19
3 × 3 RESIZE
CLEAR
FILL WITH 0
ALPHABET (26 LET. A=0) ABCDEFGHIJKLMNOPQRSTUVWXYZ
ALPHABET (26 LET. A=1) ZABCDEFGHIJKLMNOPQRSTUVWXY
ALPHABET (27 CHAR. A=0) ABCDEFGHIJKLMNOPQRSTUVWXYZ_
ALPHABET (27 CHAR. A=1) _ABCDEFGHIJKLMNOPQRSTUVWXYZ
CUSTOM ALPHANUMERIC ALPHABET
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
▶ DECRYPT
See also: Affine Cipher
Summary
Hill Decoder
Hill Encoder
Matrix Inversion
What is the Hill cipher? (Definition)
How to encrypt using Hill cipher?
How to decrypt Hill cipher?
Why must the determinant of the Hill matrix be coprime with 26?
How to recognize Hill ciphertext?
How to decipher Hill without the key matrix?
What are the variants of the Hill cipher?
When was the Hill cipher invented?
Similar pages
Affine Cipher
Inverse of a Matrix
Matrix Calculator
Variant Beaufort Cipher
Bellaso Cipher
Feedback
Type here to search
9:06 PM
2/11/2024

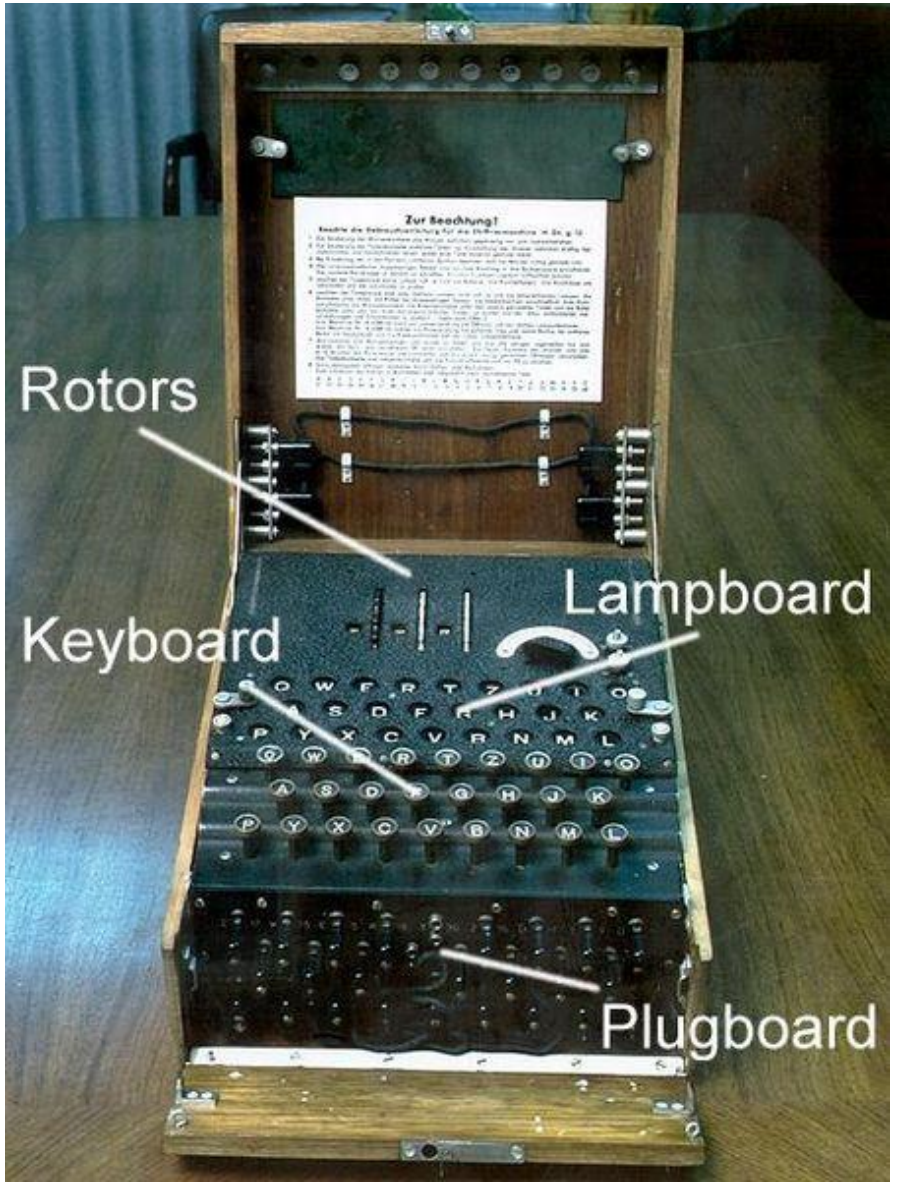
5. Enigma Cipher

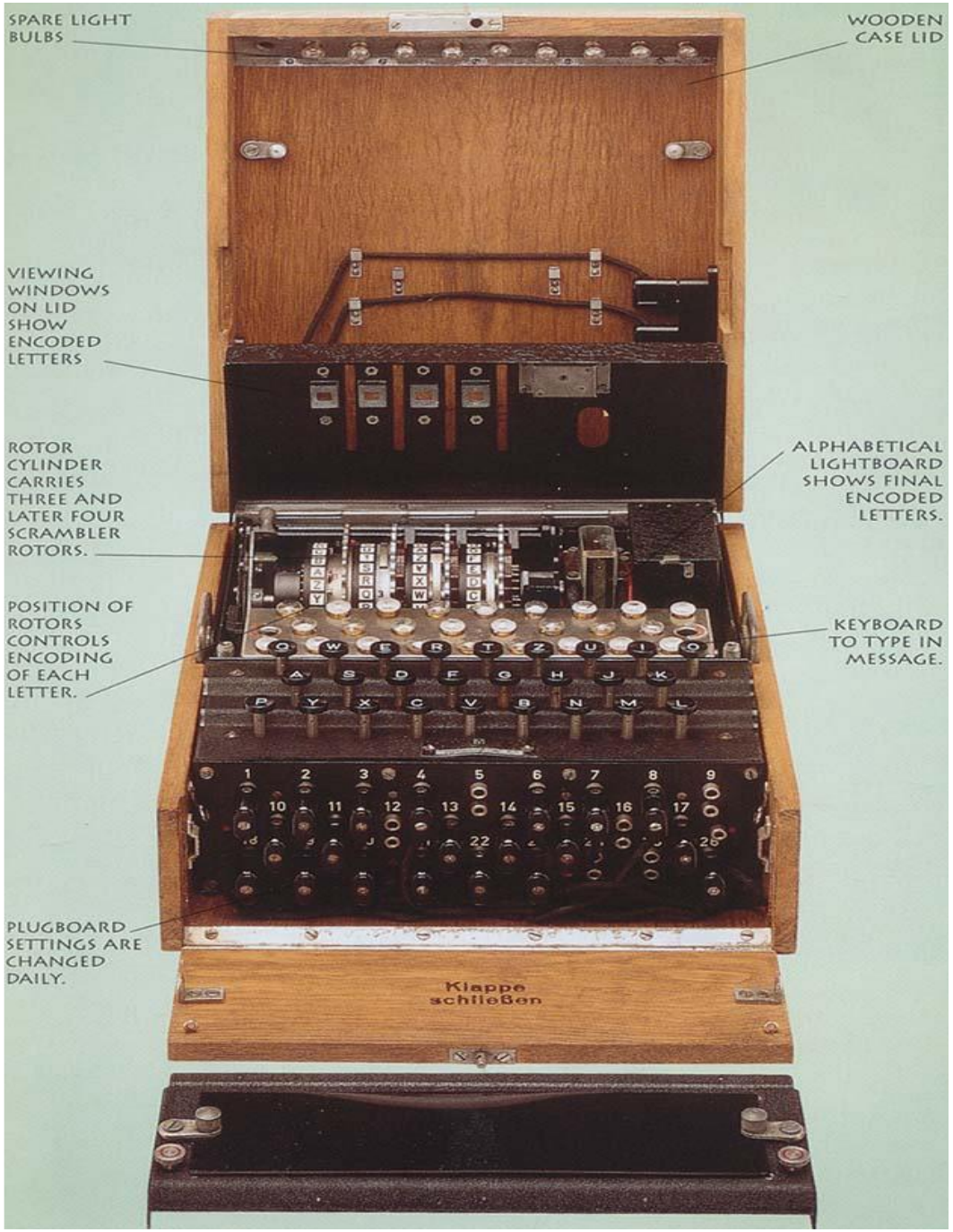
- Enigma adalah mesin enkripsi elektromekanik untuk melakukan enkripsi dan dekripsi.

- Ditemukan dan dipatenkan oleh insinyur Jerman, Arthur Scherbius, untuk tujuan komersil, diplomatik, dan militer

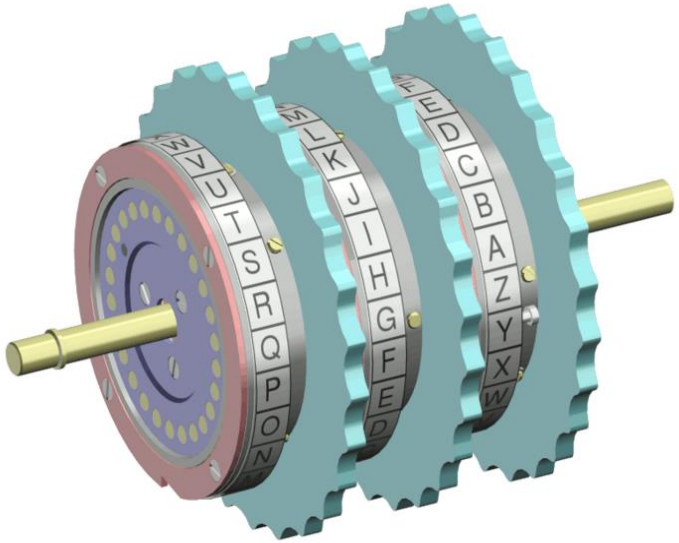
- Menjadi terkenal karena digunakan oleh tantara Nazi Jerman selama Perang Dunia II untuk mengenkripsi/dekripsi pesan-pesan militer.

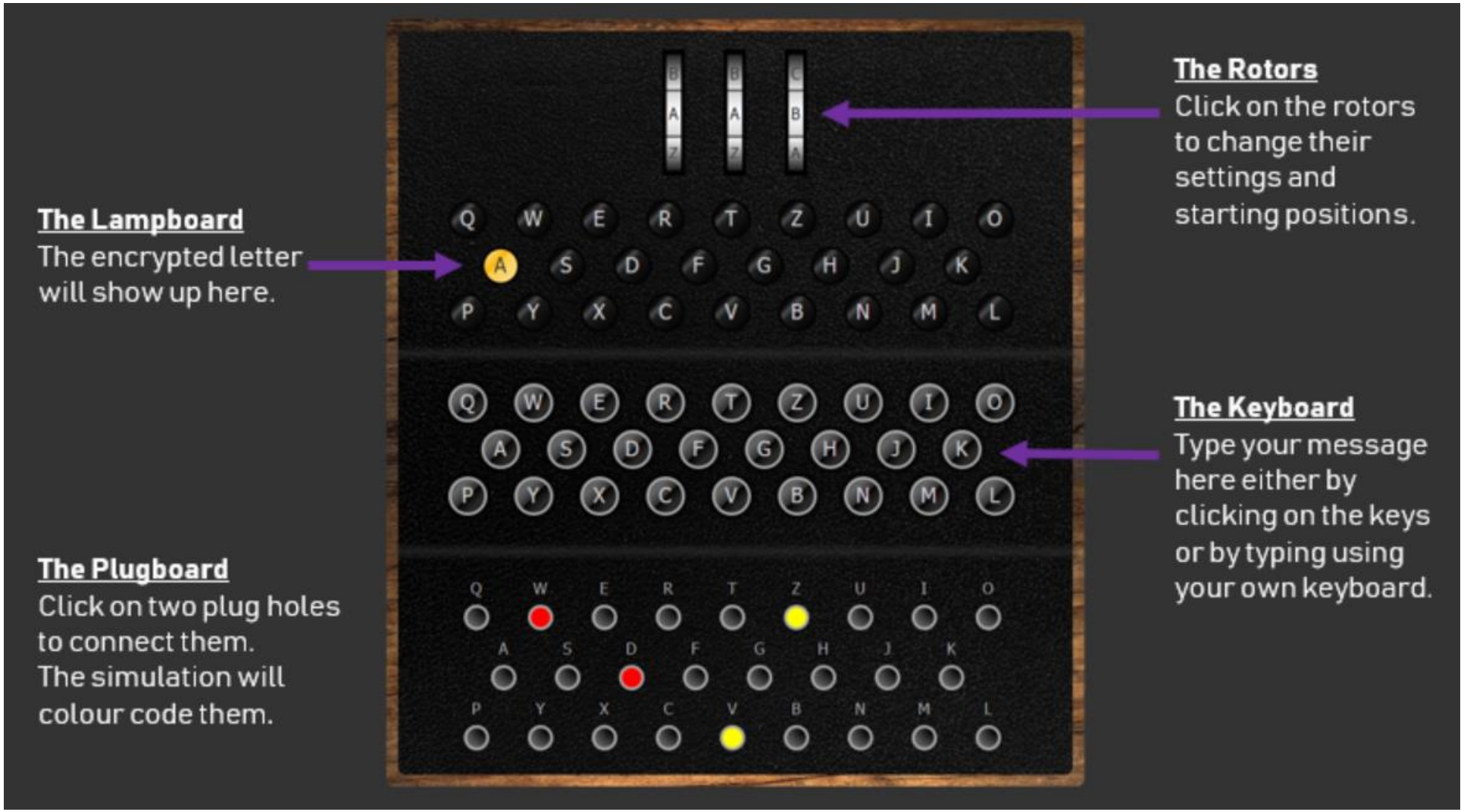
- Enigma berasal dari bahasa latin, *enigmae*, yang artinya teka-teki.





Enigma Rotors





Sumber gambar: https://www.101computing.net/enigma/enigma-instructions.html

Operator mengetik huruf plainteks pada keyboard, lalu menyalin ulang huruf cipherteks yang menyala pada *lampboard*. Cipherteks dikirim ke penerima pesan

- Enigma menggunakan sistem *rotor* (roda berputar) untuk membentuk huruf cipherteks yang berubah-ubah.

- Setiap rotor melakukan substitusi abjad-tunggal (*monoalphabetic cipher*).

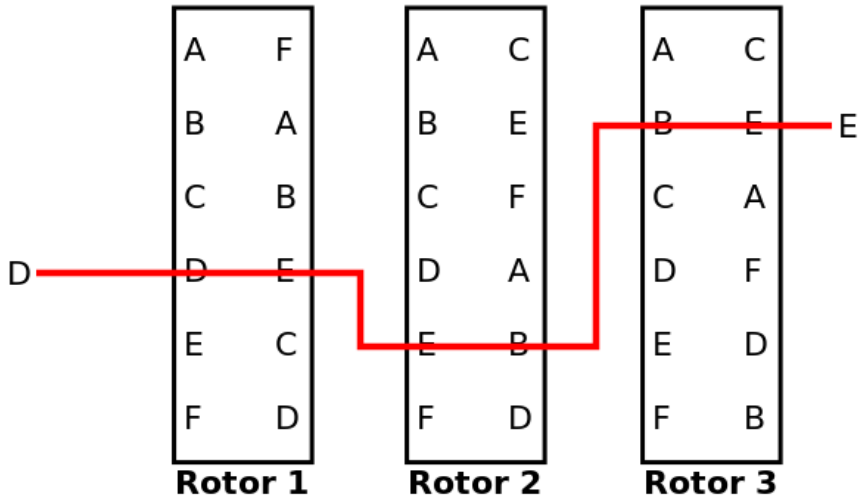
- Hasil substitusi oleh suatu rotor menjadi huruf input untuk operasi substitusi rotor selanjutnya.

- Hasil substitusi oleh rotor terakhir menjadi huruf cipherteks.

- Setiap kali sebuah huruf dienkripsi oleh sebuah rotor, *rotor* berputar satu huruf untuk membentuk huruf cipherteks baru bagi huruf plainteks berikutnya.

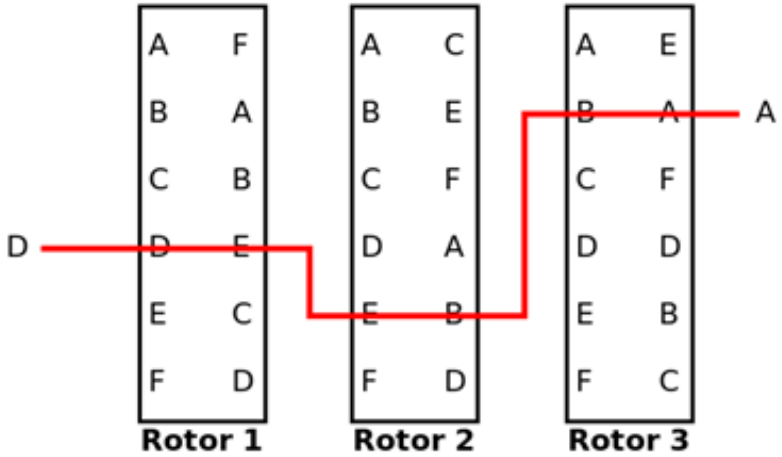
- Setelah berputar 26 huruf rotor kembali pada posisi semula. Jadi, diperoleh cipher abjad-majemuk dengan periode 26.

- Sebagai contoh, tinjau 3 rotor yang disederhanakan menjadi hanya 6 huruf alfabet:



- Misalkan huruf plainteks D ditekan pada keyboard
- Huruf D dienkripsi oleh roto pertama menjadi E
- Huruf E menjadi input untuk rotor kedua, dienkripsi menjadi B
- Huruf B menjadi input untuk rotor ketiga, dienkripsi menjadi E
- Jadi, huruf D dienkripsi menjadi E

- Setelah D dienkripsi menjadi E, rotor ketiga bergeser satu huruf.
- Jika D dienkripsi kembali, maka hasilnya adalah A



Sumber: https://www.usna.edu/Users/cs/nchamber/courses/ic211/s19/lab/l04/realenigma.html

- Model mesin enigma ada yang menggunakan 3 rotor atau 4 rotor, setiap rotor melakukan operasi substitusi cipher abjad-tunggal.

- Untuk mesin enigma 4-rotor, berarti terdapat $26 \times 26 \times 26 \times 26 = 456.976$ kemungkinan huruf cipherteks sebagai pengganti huruf plainteks sebelum terjadi perulangan urutan cipherteks.

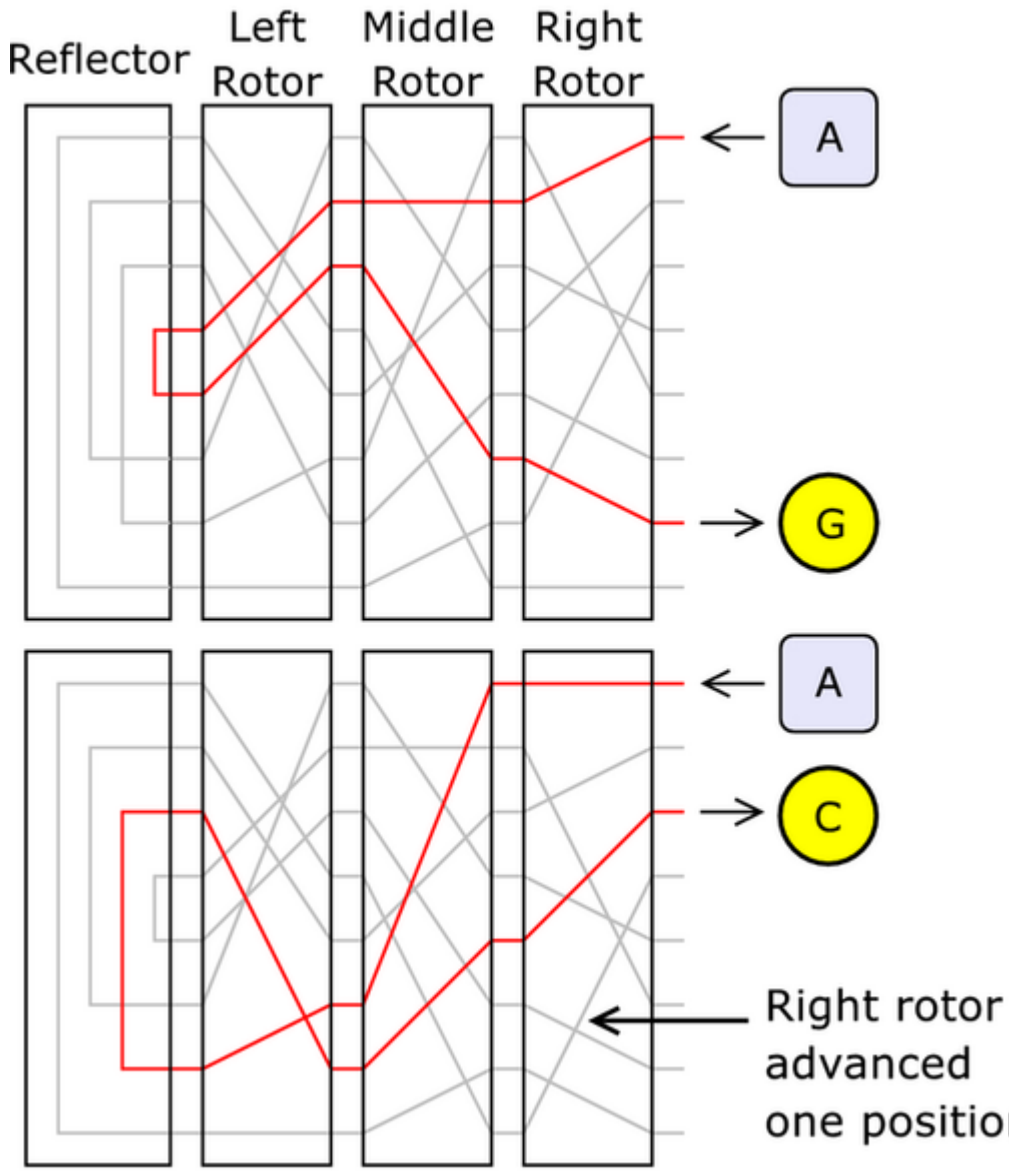
- Setiap kali sebuah huruf selesai disubstitusi, *rotor* pertama bergeser satu huruf.

- Setiap kali *rotor* pertama selesai bergeser 26 kali, rotor kedua bergeser satu huruf. Setelah rotor kedua bergeser 26 kali, rotor ketiga bergeser satu huruf. Setelah rotor ketiga bergeser 26 kali, rotor keempat bergeser satu huruf.

- Posisi awal keempat *rotor* dapat di-*set*; dan posisi awal ini menyatakan kunci dari Enigma.

- Kriptanalisis mesin Enigma pertama kali ditemukan pada tahun 1932 oleh kriptografer Polandia, yaitu Marian Rejewski, Jerzy Różycki dan Henryk Zygalski.

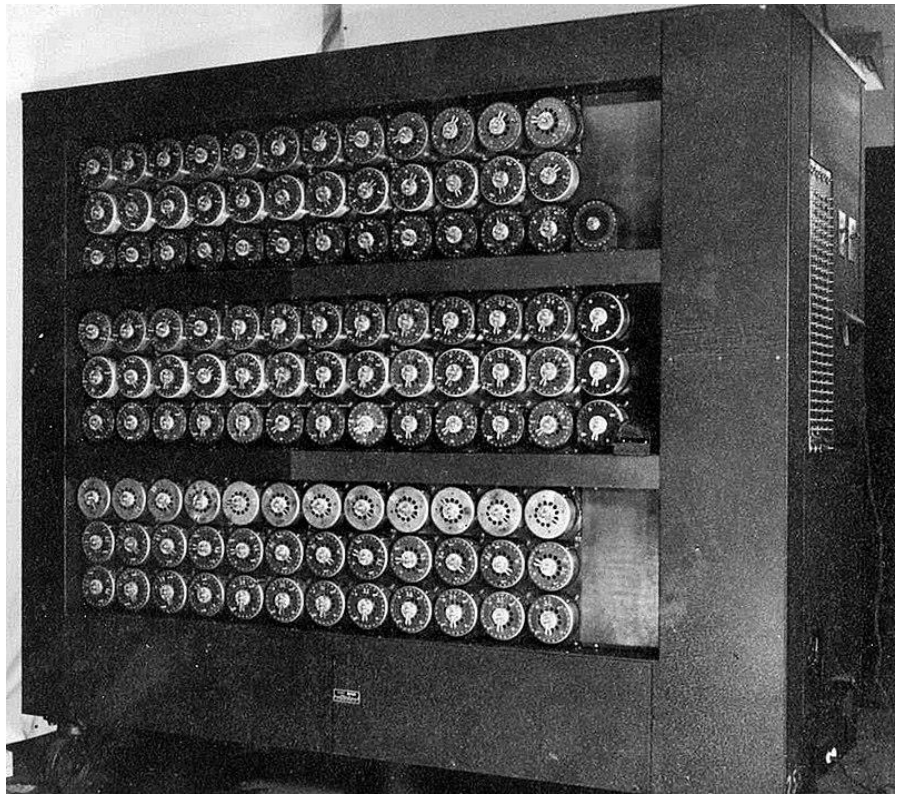
- Pemerintahan Nazi Jerman kemudian mendesain ulang mesin Enigma pada tahun 1939 dengan menambahkan *plugboard* dan *refelector*, sehingga proses enkripsi menjadi lebih kompleks. Metode kriptanalisis Enigma sebelumnya tidak dapat digunakan lagi.



- Jerman sangat percaya diri bahwa Enigma tidak akan dapat dipecahkan.

- Namun, dengan bantuan Polandia, Perancis dan Inggris kemudian membuat mesin pemecah Enigma baru ini, yang diberi nama *bombe*.

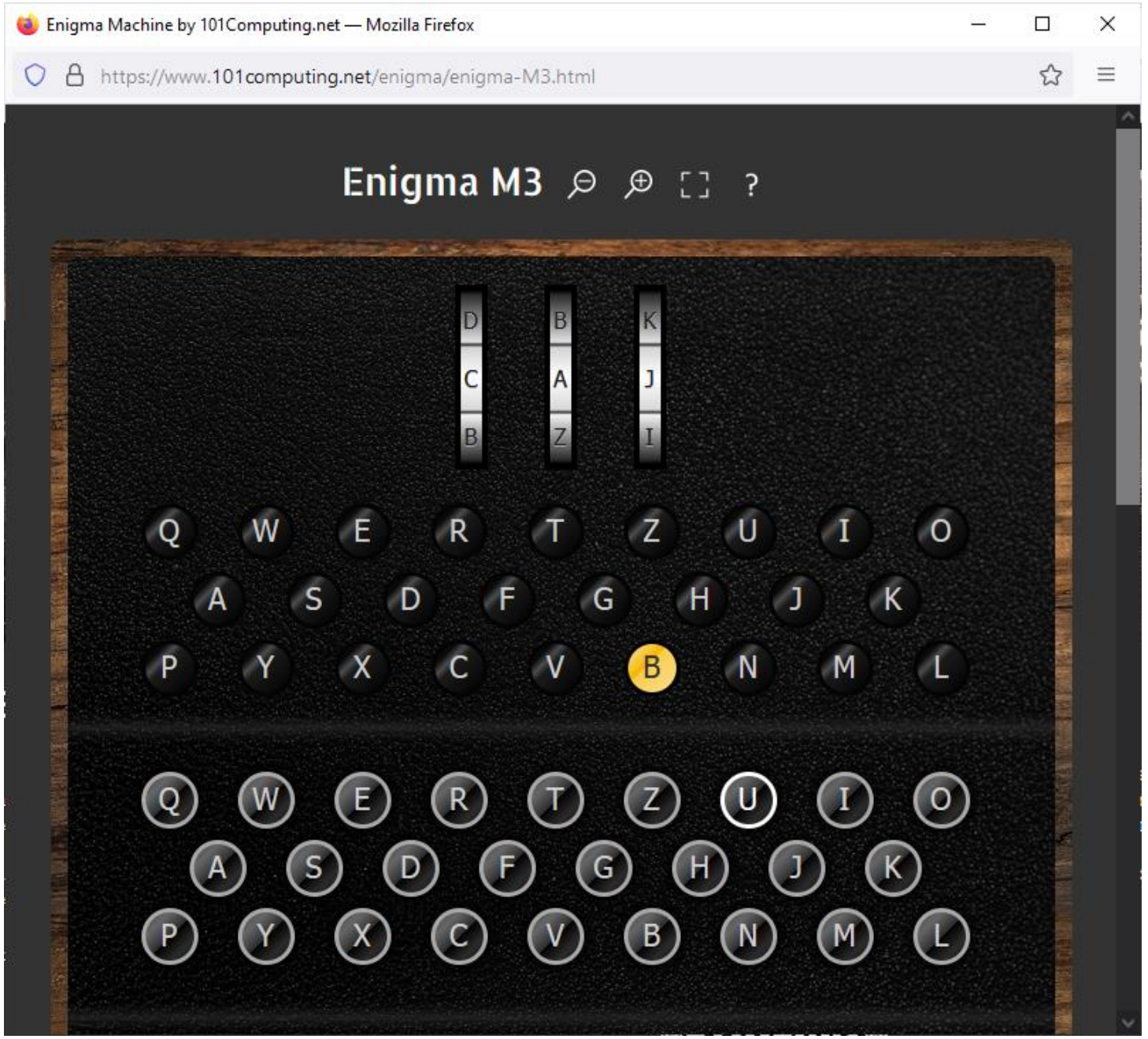
- *Bombe* dirancang oleh Alan Turing.

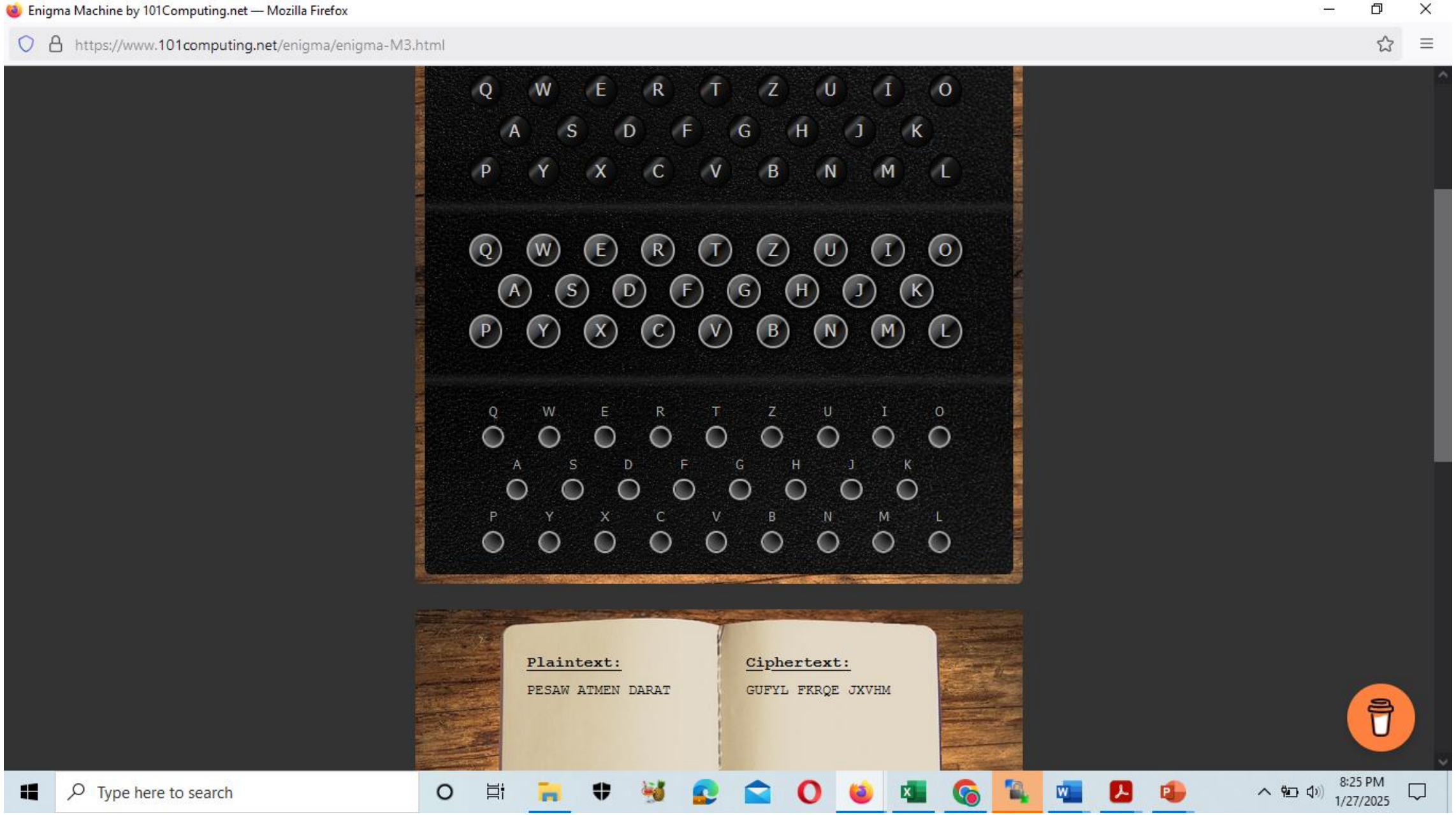


- *Bombe* berhasil memecahkan Enigma Cipher buatan Jerman.

- Keberhasilan memecahkan Enigma Cipher dianggap sebagai faktor yang memperpendek perang dunia kedua menjadi hanya dua tahun.

Coba simulator online Enimga di: https://www.101computing.net/enigma/enigma-instructions.html


Enigma Machine by 101Computing.net — Mozilla Firefox
https://www.101computing.net/enigma/enigma-M3.html
Enigma M3
D C B
B A Z
K J I
Q W E R T Z U I O
A S D F G H J K
P Y X C V B N M L
Q W E R T Z U I O
A S D F G H J K
P Y X C V B N M L


Enigma Machine by 101Computing.net — Mozilla Firefox
https://www.101computing.net/enigma/enigma-M3.html
Q W E R T Z U I O
A S D F G H J K
P Y X C V B N M L
Plaintext:
PESAW ATMEN DARAT
Ciphertext:
GUFYL FKRQE JXVHM
Type here to search
8:25 PM
1/27/2025

Encryption Steps:

Keyboard Input: P
Rotors Position: ABE
Plugboard Encryption: P
Wheel 3 Encryption: W
Wheel 2 Encryption: U
Wheel 1 Encryption: A
Reflector Encryption: Y
Wheel 1 Encryption: O
Wheel 2 Encryption: T
Wheel 3 Encryption: G
Plugboard Encryption: G
Output (Lampboard): G

Keyboard Input: E
Rotors Position: ABM
Plugboard Encryption: E
Wheel 3 Encryption: W
Wheel 2 Encryption: U
Wheel 1 Encryption: A
Reflector Encryption: Y
Wheel 1 Encryption: O
Wheel 2 Encryption: T
Wheel 3 Encryption: Q
Plugboard Encryption: Q
Output (Lampboard): Q

Keyboard Input: S
Rotors Position: ABG
Plugboard Encryption: S
Wheel 3 Encryption: K
Wheel 2 Encryption: G
Wheel 1 Encryption: D
Reflector Encryption: H
Wheel 1 Encryption: P
Wheel 2 Encryption: P
Wheel 3 Encryption: F
Plugboard Encryption: F
Output (Lampboard): F

TAMAT