

LAPORAN PENUGASAN III



Universitas Teknologi Digital

Disusun Oleh :

1. Raisya Puspa Kencana – 20123007
2. Amelia Nazelina Kusmana – 20123028

**PROGRAM STUDI S1 INFORMATIKA UNIVERSITAS
TEKNOLOGI DIGITAL TAHUN AJARAN 2024/2025**

Penugasan

1. Implementasi Vigenere Cipher dan analisis frekuensinya.
2. Tulis laporannya dengan analisis ciphertext.
3. Gunakan CrypTool untuk validasi hasil

Vigenere Cipher

```
 1 import string
 2 from collections import Counter
 3
 4 # Huruf A sampai Z
 5 ALPH = string.ascii_uppercase
 6
 7 # Bersihin teks biar cuma huruf aja dan huruf besar semua
 8 def normalize(s):
 9     return "".join(ch for ch in s.upper() if ch.isalpha())
10
11 # Fungsi buat enkripsi (mengubah teks asli jadi ciphertext)
12 def vigenere_encrypt(plaintext, key):
13     pt = normalize(plaintext)
14     key = normalize(key)
15     ct_chars = []
16     klen = len(key)
17     for i, ch in enumerate(pt):
18         shift = ord(key[i % klen]) - 65
19         ct_chars.append(chr((ord(ch) - 65 + shift) % 26 + 65))
20     return "".join(ct_chars)
21
22 # Fungsi buat dekripsi (mengubah ciphertext jadi teks asli)
23 def vigenere_decrypt(ciphertext, key):
24     ct = normalize(ciphertext)
25     key = normalize(key)
26     pt_chars = []
27     klen = len(key)
28     for i, ch in enumerate(ct):
29         shift = ord(key[i % klen]) - 65
30         pt_chars.append(chr((ord(ch) - 65 - shift) % 26 + 65))
31     return "".join(pt_chars)
32
33 # Fungsi buat hitung frekuensi huruf
34 def frequency_analysis(text):
35     t = normalize(text)
36     total = len(t)
37     cnt = Counter(t)
38     table = [(ch, cnt.get(ch,0), (cnt.get(ch,0)/total*100 if total>0 else 0)) for ch in ALPH]
39     return table
40
41 # Contoh penggunaan (kamu bisa ubah plaintext & key-nya)
42 if __name__ == "__main__":
43     plaintext = "VIGENERE CIPHER IS MORE SECURE THAN CAESAR"
44     key = "KEY"
45
46     ciphertext = vigenere_encrypt(plaintext, key)
47     decrypted = vigenere_decrypt(ciphertext, key)
48     freq_table = frequency_analysis(ciphertext)
49
50     print("Plaintext :", plaintext)
51     print("Key      :", key)
52     print("Ciphertext :", ciphertext)
53     print("Decrypted  :", decrypted)
54     print("\n--- Frequency Table ---")
55     print("Huruf | Jumlah | Persentase")
56     for ch,count,perc in freq_table:
57         if count>0:
58             print(f"{ch:>3} | {count:>2} | {perc:5.2f}%")
```

analisis ciphertext

Plaintext : VIGENERE CIPHER IS MORE SECURE THAN CAESAR

Key : KEY

Ciphertext : FMEORCBIASTFOVGCQMBIQOGSBIRRELMECCEP

Decrypted : VIGENERECIPHERISMORESECURETHANCAESAR

--- Frequency Table ---

Huruf	Jumlah	Persentase
A	1	2.78%
B	3	8.33%
C	4	11.11%
E	4	11.11%
F	2	5.56%
G	2	5.56%
I	3	8.33%
L	1	2.78%
M	3	8.33%
O	3	8.33%
P	1	2.78%
Q	2	5.56%
R	3	8.33%
S	2	5.56%
T	1	2.78%
V	1	2.78%

Implementasi Vigenère Cipher

Implementasi memakai metode Vigenère klasik: setiap huruf plaintext digeser menurut huruf kunci yang diulang sepanjang teks. Program mengimplementasikan fungsi vigenere_encrypt dan vigenere_decrypt serta fungsi frequency_analysis untuk menghitung frekuensi huruf pada ciphertext. Contoh: plaintext VIGENERECIPHERISMORESECURETHANCEASAR dengan key KEY menghasilkan ciphertext FMEORCBIASTFOVGCQMBIQOGSBIRRELMIYCEP. Analisis frekuensi menunjukkan distribusi huruf pada ciphertext (tabel frekuensi disertakan).

Kelemahan: meskipun Vigenère mengacak pergeseran sehingga lebih baik dari Caesar, struktur statistik tetap tersisa jika kunci terlalu pendek atau berulang; teknik seperti Friedman test / Kasiski dapat menemukan panjang kunci dan memungkinkan analisis frekuensi per-subkey.

Validasi Hasil Cryptool

