Bagian A

Buatlah sebuah program kalkulator enkripsi-dekripsi berbasis web (web based) dengan bahasa pemrograman bebas (Javascript/Python/Ruby/Golang/PHP, dll pilih salah satu) dengan antarmuka (GUI) yang mengimplementasikan:

a) Vigenere Cipher standard (26 huruf alfabet)
b) Varian Vigenere Cipher (26 huruf alfabet): Auto-key Vigenere Cipher
c) Extended Vigenere Cipher (256 karakter ASCII)
d) Affine Cipher
e) Playfair Cipher (26 huruf alfabet)
f) Hill Cipher
g) (Bonus) Enigma cipher

Source Code

Constant

Pada file ini, akan diinisialisasi sebuah string yang berisi huruf-huruf besar dalam alfabet serta variabel-variabel konstan untuk keperluan proses testing setiap cipher yang dibangun.

```
import numpy as np
import string

ALPHABET = string.ascii_uppercase

PLAIN_TEXT = "Created By Bintang"
KEY = "#Bintang_138"

AFFINE_KEY_M = 7
AFFINE_KEY_B = 1

HILL_KEY = np.matrix("17 17 5; 21 18 21; 2 2 19")

ENIGMA_STECKERBRETT = {' ' : ' '}
ENGIMA_ALPHA = 5
ENGIMA_BETA = 17
ENGIMA_GAMMA = 24
```

Utils

Pada file ini, terdapat sebuah fungsi untuk memformat sebuah teks random menjadi sebuah teks yang hanya berisi huruf alfabet dan dikonversi menjadi huruf besar.

```
from re import sub
def remove_non_alphabet(text: str) -> str:
    return sub(r'[^A-Z]', '', text.upper())
```

Vignere Cipher

Source code:

```
# region Vigenere
class Vigenere:
 # generate key by repeating the key until len(key) is equals len(text)
 def generate_repeating_key(self, text: str, key: str) -> str:
  return key * (len(text) // len(key)) + key[:len(text) % len(key)]

 def encrypt(self, plain_text: str, key: str) -> str:
  # Make sure it's only 26 alphabet characters
  plain_text = remove_non_alphabet(plain_text)
  key = remove_non_alphabet(key)

  return "".join(
   map(
    lambda c_plain_text, c_key: ALPHABET[(ALPHABET.index(c_plain_text) + ALPHABET.index(c_key)) % 26],
    plain_text,
    self.generate_repeating_key(plain_text, key)
   )
  )

 def decrypt(self, cipher_text: str, key: str) -> str:
  # Make sure it's only 26 alphabet characters
  cipher_text = remove_non_alphabet(cipher_text)
  key = remove_non_alphabet(key)

  return "".join(
   map(
    lambda c_cipher_text, c_key: ALPHABET[(ALPHABET.index(c_cipher_text) + 26 - ALPHABET.index(c_key)) % 26],
    cipher_text,
    self.generate_repeating_key(cipher_text, key)
   )
  )
# endregion Vigenere

print("\n--- Vigenere ---")
print(f"Plain Text: {PLAIN_TEXT}\nKey: {KEY}")
vigenere_cipher_text = Vigenere().encrypt(PLAIN_TEXT, KEY)
print(f"Encrypt result: {vigenere_cipher_text}")
print(f"Decrypt result: {Vigenere().decrypt(vigenere_cipher_text, KEY)}")
```

- **Result**

  --- Vigenere ---

  Plain Text: Created By Bintang

  Key: #Bintang_138

  Encrypt result: DZRTTRJCGOBNGGOO

  Decrypt result: CREATEDBYBINTANG

IF4020 Kriptografi
Semester 2 - 2022/2023
Dosen Pengampu : Dr. Ir. Rinaldi Munir, M.T.

Tugas Kecil 1
Nama : Bintang Fajarianto
NIM : 13519138

Auto-Key Vignere Cipher

Source code:

```
# region AutoKeyVigenere
class AutoKeyVigenere(Vigenere):
  # generate key by filling the key with the text until len(key) is equals len(text)
  def generate_auto_key(self, text: str, key: str) -> str:
    if len(text) > len(key):
      return key + text[:len(text) - len(key)]
    return key

  def encrypt(self, plain_text: str, key: str) -> str:
    # Make sure it's only 26 alphabet characters
    plain_text = remove_non_alphabet(plain_text)
    key = remove_non_alphabet(key)

    return super().encrypt(plain_text, self.generate_auto_key(plain_text, key))

  def decrypt(self, cipher_text: str, key: str) -> str:
    # Make sure it's only 26 alphabet characters
    cipher_text = remove_non_alphabet(cipher_text)
    key = remove_non_alphabet(key)

    list_key = list(key)
    list_plain_text = ""

    for idx in range(len(cipher_text)):
      c_plain_text = ALPHABET[(ALPHABET.index(cipher_text[idx]) + 26 - ALPHABET.index(list_key[idx])) % 26]
      list_plain_text += c_plain_text
      list_key.append(c_plain_text)

    return "".join(list_plain_text)
# endregion AutoKeyVigenere

print("\n--- Auto Key Vigenere ---")
print(f"Plain Text: {PLAIN_TEXT}\nKey: {KEY}")
auto_key_vigenere_cipher_text = AutoKeyVigenere().encrypt(PLAIN_TEXT, KEY)
print(f"Encrypt result: {auto_key_vigenere_cipher_text}")
print(f"Decrypt result: {AutoKeyVigenere().decrypt(auto_key_vigenere_cipher_text, KEY)}")
```

- **Result**

  --- Auto Key Vigenere ---

  Plain Text: Created By Bintang

  Key: #Bintang_138

  Encrypt result: DZRTTRJDPFIGXDOE

  Decrypt result: CREATEDBYBINTANG

Extended Vignere Cipher

Source code:

```
# region ExtendedVigenere
class ExtendedVigenere(Vigenere):
  def encrypt(self, plain_text: str, key: str) -> str:
    return "".join(
      map(
        lambda c_plain_text, c_key: chr((ord(c_plain_text) + ord(c_key)) % 256),
        plain_text,
        super().generate_repeating_key(plain_text, key)
      )
    )

  def decrypt(self, cipher_text: str, key: str) -> str:
    return "".join(
      map(
        lambda c_cipher_text, c_key: chr((ord(c_cipher_text) - ord(c_key)) % 256),
        cipher_text,
        super().generate_repeating_key(cipher_text, key)
      )
    )
# endregion ExtendedVigenere

print("\n--- Extended Vigenere ---")
print(f"Plain Text: {PLAIN_TEXT}\nKey: {KEY}")
extended_vigenere_cipher_text = ExtendedVigenere().encrypt(PLAIN_TEXT, KEY)
print(f"Encrypt result: {extended_vigenere_cipher_text}")
print(f"Decrypt result: {ExtendedVigenere().decrypt(extended_vigenere_cipher_text, KEY)}"
```

- **Result**

  --- Extended Vigenere ---

  Plain Text: Created By Bintang

  Key: #Bintang_138

  Encrypt result: f˝ÎÏèÆÒ¡ªSz°ÝÏâÈ

  Decrypt result: Created By Bintang

Affine Cipher

Source code:

```
# region Affine
class Affine:
 def inverse_mod(self, key_m: int, base: int) -> int:
  for i in range(1, base):
   if ((key_m * i) % base) == 1:
    return i

 def encrypt(self, plain_text: str, key_m: int, key_b: int):
  # Make sure it's only 26 alphabet characters
  plain_text = remove_non_alphabet(plain_text)

  return "".join(
   map(
    lambda c_plain_text: ALPHABET[(ALPHABET.index(c_plain_text) * key_m + key_b) % 26],
    plain_text,
   )
  )

 def decrypt(self, cipher_text: str, key_m: int, key_b: int):
  # Make sure it's only 26 alphabet characters
  cipher_text = remove_non_alphabet(cipher_text)

  return "".join(
   map(
    lambda c_cipher_text: ALPHABET[self.inverse_mod(key_m, len(ALPHABET)) * (ALPHABET.index(c_cipher_text) - key_b) % 26],
    cipher_text,
   )
  )
# endregion Affine
```

```
print("\n--- Affine ---")
print(f"Plain Text: {PLAIN_TEXT}\nKey m: {AFFINE_KEY_M}\nKey b: {AFFINE_KEY_B}")
vigenere_cipher_text = Affine().encrypt(PLAIN_TEXT, AFFINE_KEY_M, AFFINE_KEY_B)
print(f"Encrypt result: {vigenere_cipher_text}")
print(f"Decrypt result: {Affine().decrypt(vigenere_cipher_text, AFFINE_KEY_M, AFFINE_KEY_B)}")
```

- **Result**

  --- Affine ---

  Plain Text: Created By Bintang

  Key m: 7

  Key b: 1

  Encrypt result: PQDBEDWINIFOEBOR

  Decrypt result: CREATEDBYBINTANG

Playfair Cipher

Source code:

```
# region Playfair
class Playfair:
  def generate_bigrams(self, text: str) -> list:
    modified_text = ""

    # separate two consecutive chars with 'X' (uncommon repeated pair)
    for i in range(1, len(text)):
      modified_text += text[i-1]
      if text[i] == text[i-1]:
        modified_text += 'X'    # better approach instead of using list and insert('x')
    modified_text += text[-1]

    # add char 'X' on the last of the modified text if it's is odd
    if (len(modified_text) % 2) == 1:
      modified_text += 'X'

    return findall('..', modified_text)

  def generate_table_key(self, key: str) -> list:
    list_key = "".join(OrderedDict.fromkeys(key.replace("J", "I"))) + "".join([c for c in ALPHABET if (c not in key) and (c != 'J')])
    return [list(text) for text in findall('.....', list_key)]

  def locate_position(self, key, first_ch: str, second_ch: str) -> dict:
```

```
    x1, y1 = np.where(np.array(key) == first_ch)
    x2, y2 = np.where(np.array(key) == second_ch)
    return {'x1': x1[0], 'y1': y1[0], 'x2': x2[0], 'y2': y2[0]}

  def encrypt(self, plain_text: str, key: str) -> str:
    # Make sure it's only 26 alphabet characters
    plain_text = remove_non_alphabet(plain_text)
    key = remove_non_alphabet(key)

    table_key = self.generate_table_key(key)
    cipher_text = ""

    for pair in self.generate_bigrams(plain_text):
      c_pos = self.locate_position(table_key, pair[0], pair[1])

      if (c_pos['x1'] == c_pos['x2']):
        cipher_text += (table_key[c_pos['x1']][(c_pos['y1'] + 1) % 5] + table_key[c_pos['x2']][(c_pos['y2'] + 1) % 5])
      elif (c_pos['y1'] == c_pos['y2']):
        cipher_text += (table_key[(c_pos['x1'] + 1) % 5][c_pos['y1']] + table_key[(c_pos['x2'] + 1) % 5][c_pos['y2']])
      else:
        cipher_text += (table_key[c_pos['x1']][c_pos['y2']] + table_key[c_pos['x2']][c_pos['y1']])
    return cipher_text

  def decrypt(self, cipher_text: str, key: str) -> str:
    # Make sure it's only 26 alphabet characters
    cipher_text = remove_non_alphabet(cipher_text)
    key = remove_non_alphabet(key)

    table_key = self.generate_table_key(key)
    plain_text = ""

    for idx in range(0, len(cipher_text), 2):
      c_pos = self.locate_position(table_key, cipher_text[idx], cipher_text[idx + 1])

      if (c_pos['x1'] == c_pos['x2']):
        plain_text += table_key[c_pos['x1']][(c_pos['y1'] - 1) % 5] + table_key[c_pos['x2']][(c_pos['y2'] - 1) % 5]
      elif (c_pos['y1'] == c_pos['y2']):
        plain_text += table_key[(c_pos['x1'] - 1) % 5][c_pos['y1']] + table_key[(c_pos['x2'] - 1) % 5][c_pos['y2']]
      else:
        plain_text += table_key[c_pos['x1']][c_pos['y2']] + table_key[c_pos['x2']][c_pos['y1']]
```

```
    # remove 'X' (uncommon repeated pair) -> it might remove the original 'X' on plain text
    return plain_text.replace('X','')
# endregion Playfair

print("\n--- Playfair ---")
print(f"Plain Text: {PLAIN_TEXT}\nKey: {KEY}")
playfair_cipher_text = Playfair().encrypt(PLAIN_TEXT, KEY)
print(f"Encrypt result: {playfair_cipher_text}")
print(f"Decrypt result: {Playfair().decrypt(playfair_cipher_text, KEY)}")
```

- **Result**

  --- Playfair ---

  Plain Text: Created By Bintang

  Key: #Bintang_138

  Encrypt result: DQFTEMGNVTNTABBD

  Decrypt result: CREATEDBYBINTANG

Hill Cipher

Source code:

```
# region Hill
class Hill:
  # used for decrypting cipher text
  def inverse_key(self, key: list):
    det = round(np.linalg.det(key)) % 26

    for i in range(1,26):
      if ((det * i) % 26 == 1):
        return (i * np.linalg.det(key) * np.linalg.inv(key)).round() % 26
    return None

  def encrypt(self, plain_text: str, key: list) -> str:
    # Make sure it's only 26 alphabet characters
    plain_text = remove_non_alphabet(plain_text)

    cipher_text = ""

    # add additional 'X' (uncommon repeated pair)
    # if the plain text is not a multiple of the key length
```

```
    while (len(plain_text) % len(key) != 0):
      plain_text += 'X'

    for idx in range(0, len(plain_text), len(key)):
      p = np.array([[ALPHABET.index(plain_text[idx + i])] for i in range(len(key))])
      c = np.array(key).dot(p) % 26

      cipher_text += "".join([ALPHABET[c[i][0]] for i in range(len(key))])

    return cipher_text

  def decrypt(self, cipher_text: str, key: list) -> str:
    # Make sure it's only 26 alphabet characters
    cipher_text = remove_non_alphabet(cipher_text)

    key_inv = self.inverse_key(key)
    plain_text = ""

    if (key_inv.all() == None):
      return 'Cannot decrypt using this key'

    for idx in range(0, len(cipher_text), len(key)):
      p = np.array([[ALPHABET.index(cipher_text[idx + i])] for i in range(len(key))])
      c = np.array(key_inv).dot(p) % 26

      plain_text += "".join([ALPHABET[int(c[i][0])] for i in range(len(key))])

    # remove 'X' (uncommon repeated pair) -> it might remove the original 'X' on plain text
    return plain_text.replace('X','')
# endregion Hill

print("\n--- Hill ---")
print(f"Plain Text: {PLAIN_TEXT}\nKey: {HILL_KEY}")
hill_cipher_text = Hill().encrypt(PLAIN_TEXT, HILL_KEY)
print(f"Encrypt result: {hill_cipher_text}")
print(f"Decrypt result: {Hill().decrypt(hill_cipher_text, HILL_KEY)}")
```

- **Result**

  --- Hill ---

  Plain Text: Created By Bintang

  Key: [[17 17 5]
  [21 18 21]
  [2 2 19]]

  Encrypt result: FQKFKKGNWKWFYWZKJB

  Decrypt result: CREATEDBYBINTANG

Enigma Cipher

Source code:

```
# region Enigma
class Enigma:
  def __init__(self, steckerbrett = None, alpha = None, beta = None, gamma = None):
    self.alphabet = list(ALPHABET)

    # Steckerbrett is a sockets system that connects pairs of letters
    # that are interchanged between them.
    self.steckerbrett = {" " : " "} if (type(steckerbrett) is not dict) else steckerbrett

    if (alpha != None) and (beta != None) and (gamma != None):
      self.alpha = alpha
      self.beta = beta
      self.gamma = gamma

    else:
      # set all rotors to base states
      rotors = [self.alpha, self.beta, self.gamma]
      for rotor in rotors:
        rotor = 0 if (rotor == None) or (type(rotor) is not int) or (type(rotor) is not float) else rotor % 26
      self.alpha, self.beta, self.gamma = rotors

    # set the steckerbrett interchangeable and remove it from the alphabet
    for ch in list(self.steckerbrett.keys()):
      if ch in self.alphabet:
        self.alphabet.remove(ch)
        self.alphabet.remove(self.steckerbrett[ch])
        self.steckerbrett.update({self.steckerbrett[ch]:ch})
```

```
    # set the reflector
    self.reflector = [c for c in reversed(self.alphabet)]

  def permutate(self, rotor: int, inverse: bool = False) -> list:
    new_alphabet = list("".join(self.alphabet))

    if inverse:
      # rotate from first to last
      for _ in range(rotor):
        new_alphabet.append(new_alphabet.pop(0))
    else:
      # rotate from last to first
      for _ in range(rotor):
        new_alphabet.insert(0, new_alphabet.pop(-1))

    return new_alphabet

  def turning_rotor(self) -> None:
    self.alpha += 1
    if self.alpha % len(self.alphabet) == 0:
      self.beta += 1
      self.alpha = 0
    if self.beta % len(self.alphabet) == 0 and self.alpha % len(self.alphabet) != 0 and self.beta >= len(self.alphabet) - 1:
      self.gamma += 1
      self.beta = 1

  def encrypt(self, plain_text: str) -> str:
    # Make sure it's only 26 alphabet characters
    plain_text = remove_non_alphabet(plain_text)

    cipher_text = ""

    for ch in plain_text:
      # check if the letter exist in Steckerbrett
      if ch in self.steckerbrett:
        # encrypted into its pair
        cipher_text += self.steckerbrett[ch]

        # turning the rotors
        self.turning_rotor()
```

```
    # the letter not exist in Steckerbrett
    else:
      # forward
      # encrypted by 1st rotor
      c = self.permutate(self.alpha)[self.alphabet.index(ch)]
      # encrypted by 2nd rotor
      c = self.permutate(self.beta)[self.alphabet.index(c)]
      # encrypted by 3rd rotor
      c = self.permutate(self.gamma)[self.alphabet.index(c)]

      # return the inverse of current letter
      c = self.reflector[self.alphabet.index(c)]

      # backward
      # encrypted by 3rd rotor
      c = self.permutate(self.gamma, True)[self.alphabet.index(c)]
      # encrypted by 2nd rotor
      c = self.permutate(self.beta, True)[self.alphabet.index(c)]
      # encrypted by 1st rotor
      c = self.permutate(self.alpha, True)[self.alphabet.index(c)]

      cipher_text += c

      # turning the rotors
      self.turning_rotor()

    return cipher_text

  def decrypt(self, cipher_text: str) -> str:
    return self.encrypt(cipher_text)
# endregion Enigma


print("\n--- Enigma ---")
print(f"Plain Text: {PLAIN_TEXT}\nSteckerbrett: {ENIGMA_STECKERBRETT}\nalpha: {ENGIMA_ALPHA}\nbeta: {ENGIMA_BETA}\ngamma: {ENGIMA_GAMMA}")
enigma_cipher_text = Enigma(ENIGMA_STECKERBRETT, ENGIMA_ALPHA, ENGIMA_BETA, ENGIMA_GAMMA).encrypt(PLAIN_TEXT)
print(f"Encrypt result: {enigma_cipher_text}")
print(f"Decrypt result: {Enigma(ENIGMA_STECKERBRETT, ENGIMA_ALPHA, ENGIMA_BETA, ENGIMA_GAMMA).decrypt(enigma_cipher_text)}")
```

- **Result**

  --- Enigma ---

  Plain Text: Created By Bintang

  Steckerbrett: {' ': ' '}

  alpha: 5

  beta: 17

  gamma: 24

  Encrypt result: LYNTCTWAFEZWSNCL

  Decrypt result: CREATEDBYBINTANG

Bagian B

1) Teknik Analisis Frekuensi pada Cipher Abjad-Tunggal

Kriptogram

CZWKWFKWUFKNXHLCZWXKNWLFLCXQZWKWCZWCWKEUNSNJPXKNPNJFCWYXJWVXXSLCXCZWBWENJNJWGKWBNIU
NSEWFJNJPNJVWCXKOFRQZNVWXCZWKLAVFNECZWZNLCXKNAJXKQWPNFJLWCCVWEWJCXBUNSWJNLQZWKWCZWJFE
WYWKNUWLZWJAWUNSNJPLQWKWCZWXKNPNJFVYQWVVWKLXBUNSWJFVCWKJFCNUWVRKWAXPJNLWYWCREXVXPNL
CLLHAZFLFJFCXVRVNOWKEFJGXNJCCXCZWXVYJXKLWQXKYUNSFEWFJNJPLWFENVWCZWLGFAWVWBCOWCQWWJCQXK
XQNJPOXFCLNJAXJUXRQZNVWXCZWKLKWBWKCXCZWCZAWJCHKRFJPVXLFIXJGXWEQNYLNCZQZNAZKWBWKLCXLAFJ
YNJFUNFJGNKFCWLFLQNANJPLKWPFKYVWLLPNUWJCZWLAFJYNJFUNFJLYXENJFJAWXBCZWLWFYHKNJPCZNLWFKVRG
WKNXYNCQFLJCVXJPHJCNVAXJCKFLCNJPJFEWLQWKWOWNJPXBBWKWYHGORUFKNXHLXCZWKAHVCHKWLFAKXLLCZ
WGVFJWCCZWFKFOLLVFULFJYORDFJCNJWLBXKWIFEGVWSJWQXBCZWLWKFNYWKLFLKHLXKKZXLKWVFCNJPCXKXQNJ
PQZNVWCZWPWKEFJLVFOWVVWYCZWEFLFLAXEFJJNFLZEWJFVVHYNJPCXCZWNKFLZQXXYOXFCLXCZWKJFCNXJLLHAZ
FLCZWWJPVNLZFJYAWVCLLWCCVWYEWKWVRXJYFJWLZWFCZWJLXKGFPFJLQZNVLCCZWNKNLZKWBWKKWYCXCZWEF
LYHOPFNVFJYBNJJPFNVYFKSFJYBFNKBXKWNPJWKLXKTHNCWLHNCFOVRJXKCZEWJPNUWJCZWAXEEXJAHVCHKFVGKFA
CNAWXBCXKEWJCNJPAXFLCFVLWCCVWEWJCLFJYEXJFLCWKNWLNCQFLJCVXJPHJCNVCZWUNSNJPLBWFKLXEWKWGHC
FCNXJLGKWFYCXFVEXLCFVVAXKJWKLXBWHKXGWFJYEWLXGXCFENFCZWKWNLWUWJWUNYWJAWCZFCCZWUNSNJPLK
WFAZWYOFPZYFYCZWAWJCKWXBCZWNLVFENAWEGNKWFCCZWCNEWCZWUNSNJPFPWFLAXEEXJVRKWBWKKWYCXVFL
CWYBKXECZWWFKVRLCXCZWJXKEFJAXJTHWLCXBWJPVFJYNJCZKXHPZXHCCZNLGWKNXYCZWUNSNJPLHLWYCZWJXK
CZWKJFJYOFVCNALWFLCXCWKKXKNLWJWNPZOXHKNJPSNJPYXELWICWJYNJPCZWNKNJBVHWJAWCZKXHPZAXEOFCFJY
AHVCHKWHJCNVWUWJCHFVVRUNSNJPLAXHVYJXVXJPWKOWEWKWVRYWLAKNOWYFLAXFLCFVKFNYWKLAXJLNYWKC
ZWBFACLCQXUNSNJPSNJPLLQWRJBXKSOWFKYFJYAJHCCZWPKWFCQXHVYFLAWJYCZWWJPVNLZCZKXJWVWNBWKNSLX
JFJWFKVRNAWVFJYWKQXHVYLWCCVWLZXKCNUWYAXVXJNWLNJJXKCZFEWKNAFLAFJYNJFUNFJLQXHVYWUWJLWKUW
FLEWKAWJFKNWLBXKCZWORDFJCNJWWEGNKWNJLZXKCCZWLWQWKWJXEWKWGNKFCWLOHCCZWBXKWBFCZWKLXB
FGFCAZQXKSTHNVCAHVCHKWCZWEXCNUFCNXJLBXKLHAZWIGFJLNXJFKWLHOMWACCXYWOFCWBXKEXYWKJZNLCXK
NFJLCZXHPZCZWKWFKWAVWFKNJAWJCNUWLFLCXQZRCZWGXGHVFCNXJXBLAFJYNJFUNFENPZCZFUWFACWYNJCZWQF
RCZFCCZWRYNYYHKNJPCZNLRWFKGWKNXYXJWKWVFCNUWVRFGGFKWJCKWFLXJNLFLAFKANCRXBKWLXHKAWLCZH
LBXKANJPCZWUNSNJPLCXVXXSBHKCZWKFBNWVYWUWJKXOONJPFJYSNVVNJPAVFLLWLXBGWXGVWOVWLLWYQNCZFE
XKWOXHJCNBHVZXEWVFJYFJXCZWKGXLLNOVWLCNEHVHLNLCZWKHVWXBAZFKVWEFPJWFJYCZWKWVNPNXHLGWKL
WAHCNXJCZFCQWJCZFJYNJZFJYQNCZNCQNCZAZKNLCNFJNJBVHWJAWLWWGNJPWUWKBHKCZWKNJCXYWJEFKSLQWYW
JFJYJXKQFRNCEFSWLVXPNAFVLWJLWCZFCCZWUNSNJPLQWKWVXXSNJPCXGKXCWACCZWNKGFPFJOWVNWBLRLCWEK
WLNLCMHYWXAZKNLCNFJUFVHWLFJYWUWJCFSWKWUWJPWBXKCZXLWLWCCVWEWJCLFVKWFYRVXLCCXFEXJXCZWN
LCNAYWUXCNXJCZNLNLJXCLGWAHVFCNXJCZWNJCKXYHACNXJXBAZKNLCNFJNCRQXHVYAXEWCXYNUNYWJXKQFRBX
KFVEXLCZFVBFAWJCHKRAFHLNJPHJCXVYOVXXYLZWYFJYAHVCHKFVCKFJLBXKEFCNXJNCLZXHVYFVLXOWJXCWYCZF
CYHKNJPCZWUNSNJPFPWLAFJYNJFUNFLAVXLWLCJWNPZOXHKLQWKWWIGWKNWJANJPUFKRNJPVWUWVLXBNJJWKCHK
EXNVCZHLPKFJCNJPCZWUNSNJPLFJFYUFJCFPWQZWJWIGVXNCNJPCZWLWVFJYLBXKQWFVCZLVFUWLXKCWKKNCXKRC
ZWLWFKXHCWLHLWYORCZWUNSNJPLQWKWFVEXLCWJCNKWVRBKWWXBXGGXLNCNXJVWFUNJPCZWKFNYWKLHJNEG
WYWYFLCZWRCKFUWVVWYBKXEXJWYWLCNJFCNXJXBGVHJYWKCXCZWJWICCZNLOKWFSYXQJNJQZFCZFYXJAWOWWJ
FGKXBNCFOVWJWCQXKSXBCKFYWKXHCWLBXKWHKXGWFJSNJPYXELAFJOWZWKFVYWYOFASFLBFKFLCZWAXVVFGLW
XBCZWKXEFJWEGNKWNJCZWCZAWJCHKRFJYVFCWKCXCZWKFGNYCZAWJCHKRWIGFJLNXJXBNLVFENAGZNVXLXGZRCZ
WWJYXBCZWUNSNJPFPWAFJOWGNJJWYYXQJCXFJHEOWKXBBFACXKLBNKLCXBFVVCZWBFVVXHCCZFCXAAHKKWYBXV
VXQNJPCZWAZKNLCNFJNLFCNXJXBLAFJYNJFUNFQXHVYZFUWHJCXVYWBBWACLXJCZWKWPNXJLYXEWLCNAFJYBXKW
NPJGXVNARORCZWCZAWJCHKRYWJEFKSJXKQFRFJYLQWYWJQWKWWBBWACNUWVRAXJCKXVVWYORYNXAWLWLVWPN
CNENLWYORCZWAFCZXVNAAZHKAZFJYZFYBNKEVRWLCFOVNLZWYCZWELWVUWLFLLWGFKFCWSNJPYXELCZNLEWFJC
FJWJXKEXHLAHVCHKFVLZNBCNJCZWGKNXKNCNWLXBLAFJYNJFUNFLVWFYWKLZNGNJCZFCLWJLWCZWUNSNJPLQWKW
JXCYWBWFCWYOHCFKPHFOVREFYWCXOWZFUWNJFEFJJWKCZFCBNCCZWANUNVNCRXBCZWNKTHNASVRCKFJLBXKENJ
PZXEWVFJYLBXKWIFEGVWCZWEWYNWUFVAZHKAZEFYWNCBXKONYYWJCXCFSWBWVVXQAZKNLCNFJLFLLVFUWLPNU
WJCZWBFACCZFCLVFUWCKFYNJPQFLCZWJHEOWKXJWLXHKAWXBGKXBNCBXKCZWUNSNJPLCZNLKWEXUWYFPKWFCY
WFVXBCZWWAXJXENANJAWJCNUWCXCKFUWVFJYKFNYXUWKLWFLCZWJWQVWFYWKLZNGFVLXAZXLWCXKWBXAHLC
ZWNKENVNCFKRFCCWJCNXJBKXECZWSNJPYXELXBCZWQWLCFJYNJLCWFYGFKCFSWNJLHAZAFEGFNPJLFLCZWOFVCNA
QFKLFJYCZWFCCWEGCWYAXJTHWLCXBMWKHLFVWEBKXEZWKWXJXHCNCLWWEWYCZWUNSNJPLQWKWJXVXJPWKFK
WAXPJNLWYBXKAWNJCZWQXKVYCZXHPZCZWNKOKHCFVNCROKFUWKRFJYLCKWJPCZQXHVYVXJPOWKWEWEOWKWY
ORCZXLWQZXZFYXJAWBWVCCZWLZFKGWYPWXBCZWNKOFCCVWFIW

Steps

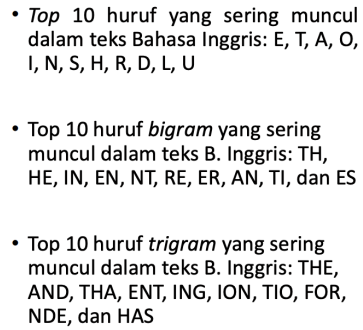
Langkah pertama, kita dapat mencari frekuensi kemunculan huruf pada *cipher text* tersebut. Dengan menggunakan baris kode berikut,

```
from collections import Counter

def text_counter(text: str, n: int) -> dict:
  return Counter(text[i : i + n] for i in range(len(text) - (n-1)))

CIPHER_TEXT_1 = ""

# Huruf paling sering muncul
print(f"\nHuruf paling sering muncul")
print(text_counter(CIPHER_TEXT_1, 1))

# Bigram paling sering muncul
print(f"\nBigram paling sering muncul")
print(text_counter(CIPHER_TEXT_1, 2))

# Trigram paling sering muncul
print(f"\nTrigram paling sering muncul")
print(text_counter(CIPHER_TEXT_1, 3))
```

diperoleh frekuensi kemunculan huruf sebagai berikut.

- Huruf paling banyak muncul **(W)**
  W: 539 - C: 376 - J: 323 - F: 320 - N: 318 - X: 298 - K: 277 - L: 269 - Z: 211 - V: 178
- Bigram paling banyak muncul **(CZ)**
  CZ: 144 - ZW: 113 - NJ: 91 - WK: 86 - FJ: 78 - KW: 70 - WJ: 67 - JP: 64 - XK: 55 - LC: 54
- Trigram paling banyak muncul **(CZW)**
  CZW: 102 - NJP: 51 - FJY: 36 - WKW: 26 - SNJ: 23 - WJC: 23 - ZWK: 22 - UNS: 21 - BXK: 19 - NSN: 17

Berdasarkan **top 10** huruf dalam teks Bahasa Inggris,



Gambar 1. Top 10 huruf dalam teks Bahasa Inggris

maka W berkorespondensi dengan huruf E, CZ berkorespondensi dengan huruf TH, CZW berkorespondensi dengan huruf THE. Sehingga dapat diperoleh pemetaan pertama: W→E, C→T, dan Z→H.

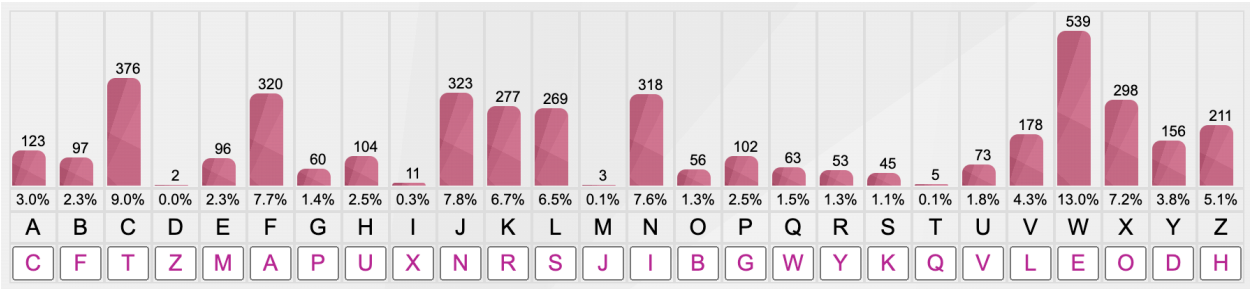
Pada iterasi yang kedua, CZWKWFKW dipetakan menjadi THE*E**E & KW berkorespondensi dengan ER berdasarkan Gambar 1. Maka, pada pemetaan kedua ini diperoleh: K→R dan F→A.

Selanjutnya, akan dianalisis bigram & trigram NJ dan NJP. Berdasarkan Gambar 1, susunan huruf tersebut memungkinkan dipetakan menjadi AN & AND atau IN & ING. Namun, karena F sudah dipetakan menjadi A, maka pada pemetaan ketiga ini diperoleh: N→I dan J→N.

Pada iterasi selanjutnya, CZWXKNWL dipetakan menjadi THE*RIE*. Berdasarkan kata yang ada pada Bahasa Inggris, potongan *cipher text* tersebut dapat diasumsikan sebagai THEORIES sehingga pada pemetaan keempat ini diperoleh: X→O dan L→S.

Setelahnya, kita dapat semakin mudah untuk menebak huruf yang belum dipetakan, seperti LCKWJPCZ dipetakan menjadi STREN*TH, GKNXKNCNWL dipetakan menjadi *RIORITIES, dan CZWKWFKWUFKNXHLCZWXKNWL dipetakan menjadi THEREARE*ARIO*STHEORIES. Maka, pada pemetaan kelima ini diperoleh: P→G, G→P, U→V, dan H→U.

Untuk mempersingkat pembahasan, dengan bantuan Website 101computing untuk melakukan *frequency analysis & subtitution*, diperoleh pemetaan sebagai berikut.



Gambar 2. Hasil Pemetaan

Hasil Dekripsi

THEREAREVARIOUSTHEORIESASTOWHERETHETERMVIKINGORIGINATEDONELOOKSTOTHEFEMININEPREFIXVIKMEANINGI
NLETORBAYWHILEOTHERSCLAIMTHEHISTORICNORWEGIANSETTLEMENTOFVIKENISWHERETHENAMEDERIVESHENCEVIK
INGSWERETHEORIGINALDWELLERSOFVIKENALTERNATIVELYRECOGNISEDETYMOLOGISTSSUCHASANATOLYLIBERMANP
OINTTOTHEOLDNORSEWORDVIKAMEANINGSEAMILETHESPACELEFTBETWEENTWOROWINGBOATSINCONVOYWHILEOTHE
RSREFERTOTHETHCENTURYANGLOSAXONPOEMWIDSITHWHICHREFERSTOSCANDINAVIANPIRATESASWICINGSREGARDLE
SSGIVENTHESCANDINAVIANSDOMINANCEOFTHESEADURINGTHISEARLYPERIODITWASNTLONGUNTILCONTRASTINGNAM
ESWEREBEINGOFFEREDUPBYVARIOUSOTHERCULTURESACROSSTHEPLANETTHEARABSSLAVSANDBYZANTINESFOREXAM

PLEKNEWOFTHESERAIDERSASRUSORRHOSRELATINGTOROWINGWHILETHEGERMANSLABELLEDTHEMASASCOMANNIASH
MENALLUDINGTOTHEIRASHWOODBOATSOTHERNATIONSSUCHASTHEENGLISHANDCELTSSETTLEDMERELYONDANESHEAT
HENSORPAGANSWHILSTTHEIRISHREFERREDTOTHEMASDUBGAILANDFINNGAILDARKANDFAIRFOREIGNERSORQUITESUIT
ABLYNORTHMENGIVENTHECOMMONCULTURALPRACTICEOFTORMENTINGCOASTALSETTLEMENTSANDMONASTERIESITW
ASNTLONGUNTILTHEVIKINGSFEARSOMEREPUTATIONSPREADTOALMOSTALLCORNERSOFEUROPEANDMESOPOTAMIATHE
REISEVENEVIDENCETHATTHEVIKINGSREACHEDBAGHDADTHECENTREOFTHEISLAMICEMPIREATTHETIMETHEVIKINGAGE
ASCOMMONLYREFERREDTOLASTEDFROMTHEEARLYSTOTHENORMANCONQUESTOFENGLANDINTHROUGHOUTTHISPERIO
DTHEVIKINGSUSEDTHENORTHERNANDBALTICSEASTOTERRORISENEIGHBOURINGKINGDOMSEXTENDINGTHEIRINFLUENC
ETHROUGHCOMBATANDCULTUREUNTILEVENTUALLYVIKINGSCOULDNOLONGERBEMERELYDESCRIBEDASCOASTALRAID
ERSCONSIDERTHEFACTSTWOVIKINGKINGSSWEYNFORKBEARDANDCNUTTHEGREATWOULDASCENDTHEENGLISHTHRONE
LEIFERIKSONANEARLYICELANDERWOULDSETTLESHORTIVEDCOLONIESINNORTHAMERICASCANDINAVIANSWOULDEVEN
SERVEASMERCENARIESFORTHEBYZANTINEEMPIREINSHORTTHESEWERENOMEREPIRATESBUTTHEFOREFATHERSOFAPATC
HWORKQUILTCULTURETHEMOTIVATIONSFORSUCHEXPANSIONARESUBJECTTODEBATEFORMODERNHISTORIANSTHOUGHT
HEREARECLEARINCENTIVESASTOWHYTHEPOPULATIONOFSCANDINAVIAMIGHTHAVEACTEDINTHEWAYTHATTHEYDIDDUR
INGTHISYEARPERIODONERELATIVELYAPPARENTREASONISASCARCITYOFRESOURCESTHUSFORCINGTHEVIKINGSTOLOOK
FURTHERAFIELDEVENROBBINGANDKILLINGCLASSESOFPEOPLEBLESSEDWITHAMOREBOUNTIFULHOMELANDANOTHERP
OSSIBLESTIMULUSISTHERULEOFCHARLEMAGNEANDTHERELIGIOUSPERSECUTIONTHATWENTHANDINHANDWITHITWITH
CHRISTIANINFLUENCESEEPINGEVERFURTHERINTODENMARKSWEDENANDNORWAYITMAKESLOGICALSENSETHATTHEVIK
INGSWERELOOKINGTOPROTECTTHEIRPAGANBELIEFSYSTEMRESISTJUDEOCHRISTIANVALUESANDEVENTAKEREVENGEFO
RTHOSESETTLEMENTSALREADYLOSTTOAMONOTHEISTICDEVOTIONTHISISNOTSPECULATIONTHEINTRODUCTIONOFCHRIS
TIANITYWOULDCOMETODIVIDENORWAYFORALMOSTHALFACENTURYCAUSINGUNTOLDBLOODSHEDANDCULTURALTRAN
SFORMATIONITSHOULDALSOBENOTEDTHATDURINGTHEVIKINGAGESCANDINAVIASCLOSESTNEIGHBOURSWEREEXPERIEN
CINGVARYINGLEVELSOFINNERTURMOILTHUSGRANTINGTHEVIKINGSANADVANTAGEWHENEXPLOITINGTHESELANDSFOR
WEALTHSLAVESORTERRITORYTHESEAROUTESUSEDBYTHEVIKINGSWEREALMOSTENTIRELYFREEOFOPPOSITIONLEAVING
THERAIDERSUNIMPEDEDASTHEYTRAVELLEDFROMONEDESTINATIONOFPLUNDERTOTHENEXTTHISBREAKDOWNINWHATH
ADONCEBEENAPROFITABLENETWORKOFTRADEROUTESFOREUROPEANKINGDOMSCANBEHERALDEDBACKASFARASTHEC
OLLAPSEOFTHEROMANEMPIREINTHETHCENTURYANDLATERTOTHERAPIDTHCENTURYEXPANSIONOFISLAMICPHILOSOPHY
THEENDOFTHEVIKINGAGECANBEPINNEDDOWNTOANUMBEROFFACTORSFIRSTOFALLTHEFALLOUTTHATOCCURREDFOLLO
WINGTHECHRISTIANISATIONOFSCANDINAVIAWOULDHAVEUNTOLDEFFECTSONTHEREGIONSDOMESTICANDFOREIGNPOLI
CYBYTHETHCENTURYDENMARKNORWAYANDSWEDENWEREEFFECTIVELYCONTROLLEDBYDIOCESESLEGITIMISEDBYTHE
CATHOLICCHURCHANDHADFIRMLYESTABLISHEDTHEMSELVESASSEPARATEKINGDOMSTHISMEANTANENORMOUSCULTUR
ALSHIFTINTHEPRIORITIESOFSCANDINAVIASLEADERSHIPINTHATSENSETHEVIKINGSWERENOTDEFEATEDBUTARGUABLYM
ADETOBEHAVEINAMANNERTHATFITTHECIVILITYOFTHEIRQUICKLYTRANSFORMINGHOMELANDSFOREXAMPLETHEMEDIE
VALCHURCHMADEITFORBIDDENTOTAKEFELLOWCHRISTIANSASSLAVESGIVENTHEFACTTHATSLAVETRADINGWASTHENUM
BERONESOURCEOFPROFITFORTHEVIKINGSTHISREMOVEDAGREATDEALOFTHEECONOMICINCENTIVETOTRAVELANDRAID
OVERSEASTHENEWLEADERSHIPALSOCHOSETOREFOCUSTHEIRMILITARYATTENTIONFROMTHEKINGDOMSOFTHEWESTAND
INSTEADPARTAKEINSUCHCAMPAIGNSASTHEBALTICWARSANDTHEATTEMPTEDCONQUESTOFJERUSALEMFROMHEREONOU
TITSEEMEDTHEVIKINGSWERENOLONGERARECOGNISEDFORCEINTHEWORLDTHOUGHTHEIRBRUTALITYBRAVERYANDSTR
ENGTHWOULDLONGBEREMEMBEREDBYTHOSEWHOHADONCEFELTTHESHARPEDGEOFTHEIRBATTLEAXE

Setelah dilakukan pencarian: teks tersebut bersumber dari cerita Vikings and Norse Gods in Iceland.

2) Metode Kasiski

Kriptogram

FSIKTSZDRCZEUGPFPOJWXRKCXVPVOQGSNESTECHYYEGKPCNOZCQMJTSFEVYSZEPXEDCCBGAGAHYHQCXRUSOKSTJCA
UUSZURCEYTMJXDKWHZFEZRLETHHSSLMEQWZMCYCLJNOAZSPLHNGFXESIHSSXCVWOUQSTBLMEQWZMCYHNYGAERPP
PROQPYOYIRNIXGBYOGWKSOZPREZOXRZKTRBFWREYPWYMAIKLXVPZGPTIOZPOVSYGAWKAXVPOYRNWEGEBOWYYISH
IEBTQRYXIETXVPAOBQPAZLSCSOQNREYPOYRIYYPAJWOXCYGEMOREBOHNCZEFUVWEMUDGLAVSDFZGRVSJGVCFBJRBF
WREYPWYZNXWQXFTPKGGMOKWHTAGRRHPNEHECHYYEGKAODTUPZIZJYFTBMYAITVPCDWULBJWHSIEMKYEWWMSKS
WIFVWWTIFFDZGBROATSCJUJPEJUWIASXTBPYGRCEVGRVWIUYBEHUZNAETHPWCCLISHIEBTQGQULWYWDCSGBGSDGPT
EVKCNVPNJFZAIFVRWZSGZIZFNJNOGOPJKLDYEZIGFFVPVWETKPPQGSFIEZXICBYMHXPNIGAFKYQSBZLSOIYRGSXKVSNC
XBRHQVXCEVKLBVPNTCWSZFRINATHTCTMPGQXVSOTUPBRACISVOTBGLAJYGEPAPFXNKEQSSJIVPKSIHPFYYOSRKWSLZ
KTRPPNISGWJCAGALSIYRGJFSNKMBQCXARWPNIBZHOMAXDGXHSSLMEGAUJHSSKPHTPOSBLBJGGWKIIOYKGTRWYUYZO
OTLVLEREHPZODRMJGXZLBZGFXDOWWYQOBRRPIEIDSJKNWOJIOEVLMYPKCIRMMZFRITZMBNHOMASBYSAPGVCPMAYE
QNCXBVRCZSRYOKTVHATGSEVOQRVQVXWZBGJFSONVOYYZFRRQSFSCCLNRSLRIHZOHMHXKLXVPHURNPDAQOYDUNHP
WZMCYCLRUIAGVHSOZRUEPZMAPOHMHXIOPZTCTNRSLRIOQHKPGLAKVIAHOMAEYGPRPFGUNWBUVAPRCFVGDZLSYTOJ
YIZCMHSKGRRVWTHPPQGKRADGXWDBUUXRKCRODHUZNPWQIIAKGPQTNKWGFFKZLXDKQORAGRUEPNEGYCZWRXYUQ
SZIZANYOKWHSSKKRVCKRQPCLQNQKYMFTGRYAHPNIFPWYYYWKLSZVZUPREXUYHECHYYEGKMGDOOBUIOGMRLHOKR
ADKRHSSXCJEOGJOCAKPAEIKHHZPGUUSSKRHQWYFVRCZSHSSXGIINZSUPHLGFLPUIOEHUZNKAZWOMWMYAHXKEIEWLS
YJEYLPFHNCVWOAVDCWYCQXDGXHSSLGFLYGRHLZQYCTWXIBEZERUIBOWVTGZFRMJIEFYOZGBRKLEDCWTARWOCLCH
OYAHVOKHTZFBGBPWZMBRHNCEYHKWCQHNCXMJMHCXOYYGLWTOMZILMELWBMBRFKJREOKHVPFLPBQPNIQFFYCGL
AVVWYQKQFAWYQOCFERBFWNSKPJKPGLAXIWDCTCCVKSMGPHNYGLWYFSPBGEEIAJYDZBZFRCONSIWRTMGXARPOYMU
LRXDGXHSSVPVRYKWGTGGDVWDLVCXHNCZENXMORSCYFFKXROMCELNQAJWOXCYGEWWSSGTFMPRAETXCLJKPLLWT
HGZAKYAHOZVCYUHMLFQZXVPFKYEIDGFWEGZFNXWSENPSBCECKTIVPORUNCOLISWGNSAKNEEBOBKTRVOGXWDTOCQ
EHRXVPTUMQVWZMCYGGPREHCEMDRKTBYNKHKTHNMHXPNIFPGZSAXERSBPRGWFEIUWWCOYQVKJKHHZRKJVZAXJCZ
RLMELEYJOEVKPVRPNITTSRBOYPZLSQCUBAIRKVQLAKRBFWGPGZOVNESWILSOGGKBWEXEBOOYIRHSNIFPHNCSSKJJCC
VOKOYPYEAZGOPNHIOXHPRZFNXPNITZCJGFEHXIOOMKYGIJZSPLKGQIINEEBRFEYAHQTOBZKOLTPUHVSLYYFVWLXSATG
KZLWWEMBRGULBJWLMGSGGKBWEXMAXSJGNXARCQZAVJNMJKHHZVOQZSPNIFTTNCPEHRIRLGULBJWLMGSHNCCVETG
SDGCYFHEYEDACOLGIZHIQLIYCUINNYGMOTBUEOHVCVSTRUILXSATGKQUIYXMSOORMGEJJXCWRYYZSOOVHZFALGSPNI
VTUNFVPHYYROSTJLZAXCVPOBWEEETGOXSJMJRSOXVLHKPEMXRIZTUNRAMJMXVPKGRRVKBIFQZUURHPUHFZKTRUIAT
XWCSBGYPWMIHSSVSQHHKXICBKBVRPUEZLYKRUEPOWBZKIYYPAJXCMORYXIPNIBEVKGFPWTHKSSXCFEIUWWCGNCYX
AXMGNORJRHOGQCDWXGFPWTHHSOZQGLANMGECXWBJPUFOWOQCGLWZWGZITGDYAXMUSHODLSQBMGTHZMOEHGOS
JCAANRBKIZEVKABSHGXAZGVFRVAUJHSSRYXIWIGCXDGLVIZHCPPOARVJQRZWPKYMSWWSSGTFOQYEJJ

Steps:

Pertama-tama, kita perlu mengetahui *substring* yang berulang didalam *cipher text* tersebut dan jarak masing-masing *substring* dengan *substring* yang sama setelahnya. Dengan memanfaatkan kode dari B.1 dan juga dengan menggunakan baris kode berikut,

```
for i in range(30, 2, -1):
  print(f"\n----- Duplicated {i}-ch -----")
  d = dict(filter(lambda elm: elm[1] > 1, dict(text_counter(CIPHER_TEXT_2, i)).items()))

  for k in d.keys():
    list_gap = CIPHER_TEXT_2.split(k)
    d[k] = [len(chBetweenItem) + i for chBetweenItem in list_gap[1:len(list_gap)-1]]

  for elem in d.items():
    print(f"Jarak {elem[0]} dengan setelahnya: {', '.join(list(map(str, elem[1])))}")
```

diperoleh *substring* yang memiliki panjang 30 hingga panjang 3 disertai dengan jarak antar *substring* tersebut. *Substring* dengan panjang dua diasumsikan terlalu pendek untuk dijadikan sebagai *key*.

----- Duplicated 30-ch -----
----- Duplicated 29-ch -----
----- Duplicated 28-ch -----
----- Duplicated 27-ch -----
----- Duplicated 26-ch -----
----- Duplicated 25-ch -----
----- Duplicated 24-ch -----
----- Duplicated 23-ch -----
----- Duplicated 22-ch -----
----- Duplicated 21-ch -----

----- Duplicated 20-ch -----
----- Duplicated 19-ch -----
----- Duplicated 18-ch -----
----- Duplicated 17-ch -----
----- Duplicated 16-ch -----
----- Duplicated 15-ch -----
----- Duplicated 14-ch -----
----- Duplicated 13-ch -----
----- Duplicated 12-ch -----
----- Duplicated 11-ch -----

----- Duplicated 10-ch -----
Jarak RBFWREYPWY dengan setelahnya: 140
Jarak GULBJWLMGS dengan setelahnya: 60

----- Duplicated 9-ch -----
Jarak LMEQWZMCY dengan setelahnya: 40
Jarak RBFWREYPW dengan setelahnya: 140
Jarak BFWREYPWY dengan setelahnya: 140
Jarak AJWOXCYGE dengan setelahnya: 1330
Jarak HECHYYEGK dengan setelahnya: 820
Jarak GULBJWLMG dengan setelahnya: 60
Jarak ULBJWLMGS dengan setelahnya: 60

----- Duplicated 8-ch -----
Jarak ECHYYEGK dengan setelahnya: 350, 820
Jarak LMEQWZMC dengan setelahnya: 40
Jarak MEQWZMCY dengan setelahnya: 40
Jarak RBFWREYP dengan setelahnya: 140
Jarak BFWREYPW dengan setelahnya: 140
Jarak FWREYPWY dengan setelahnya: 140
Jarak ISHIEBTQ dengan setelahnya: 240
Jarak AJWOXCYG dengan setelahnya: 1330
Jarak JWOXCYGE dengan setelahnya: 1330
Jarak HECHYYEG dengan setelahnya: 820
Jarak XDGXHSSL dengan setelahnya: 590
Jarak GULBJWLM dengan setelahnya: 60
Jarak ULBJWLMG dengan setelahnya: 60
Jarak LBJWLMGS dengan setelahnya: 60

----- Duplicated 7-ch -----
Jarak ECHYYEG dengan setelahnya: 350, 820
Jarak CHYYEGK dengan setelahnya: 350, 820
Jarak LMEQWZM dengan setelahnya: 40
Jarak MEQWZMC dengan setelahnya: 40
Jarak EQWZMCY dengan setelahnya: 40
Jarak WZMCYCL dengan setelahnya: 850
Jarak RBFWREY dengan setelahnya: 140
Jarak BFWREYP dengan setelahnya: 140
Jarak FWREYPW dengan setelahnya: 140
Jarak WREYPWY dengan setelahnya: 140
Jarak ISHIEBT dengan setelahnya: 240
Jarak SHIEBTQ dengan setelahnya: 240
Jarak AJWOXCY dengan setelahnya: 1330
Jarak JWOXCYG dengan setelahnya: 1330
Jarak WOXCYGE dengan setelahnya: 1330
Jarak HECHYYE dengan setelahnya: 820
Jarak XDGXHSS dengan setelahnya: 590, 240
Jarak DGXHSSL dengan setelahnya: 590
Jarak WWSSGTF dengan setelahnya: 880
Jarak FEIUWWC dengan setelahnya: 550
Jarak GGKBWEX dengan setelahnya: 140
Jarak LXSATGK dengan setelahnya: 140
Jarak GULBJWL dengan setelahnya: 60
Jarak ULBJWLM dengan setelahnya: 60
Jarak LBJWLMG dengan setelahnya: 60
Jarak BJWLMGS dengan setelahnya: 60

----- Duplicated 6-ch -----
Jarak ECHYYE dengan setelahnya: 350, 820
Jarak CHYYEG dengan setelahnya: 350, 820
Jarak HYYEGK dengan setelahnya: 350, 820
Jarak HSSLME dengan setelahnya: 640

Jarak LMEQWZ dengan setelahnya: 40
Jarak MEQWZM dengan setelahnya: 40
Jarak EQWZMC dengan setelahnya: 40
Jarak QWZMCY dengan setelahnya: 40
Jarak WZMCYC dengan setelahnya: 850
Jarak ZMCYCL dengan setelahnya: 850
Jarak RBFWRE dengan setelahnya: 140
Jarak BFWREY dengan setelahnya: 140
Jarak FWREYP dengan setelahnya: 140
Jarak WREYPW dengan setelahnya: 140
Jarak REYPWY dengan setelahnya: 140
Jarak ISHIEB dengan setelahnya: 240
Jarak SHIEBT dengan setelahnya: 240
Jarak HIEBTQ dengan setelahnya: 240
Jarak IYYPAJ dengan setelahnya: 2030
Jarak AJWOXC dengan setelahnya: 1330
Jarak JWOXCY dengan setelahnya: 1330
Jarak WOXCYG dengan setelahnya: 1330
Jarak OXCYGE dengan setelahnya: 1330
Jarak HECHYY dengan setelahnya: 820
Jarak MHXPNI dengan setelahnya: 1200
Jarak XDGXHS dengan setelahnya: 590, 240
Jarak DGXHSS dengan setelahnya: 590, 240
Jarak GXHSSL dengan setelahnya: 590
Jarak AUJHSS dengan setelahnya: 1720
Jarak NRSLRI dengan setelahnya: 70
Jarak WWSSGT dengan setelahnya: 880
Jarak WSSGTF dengan setelahnya: 880
Jarak FEIUWW dengan setelahnya: 550
Jarak EIUWWC dengan setelahnya: 550
Jarak GGKBWE dengan setelahnya: 140
Jarak GKBWEX dengan setelahnya: 140
Jarak LXSATG dengan setelahnya: 140
Jarak XSATGK dengan setelahnya: 140
Jarak GULBJW dengan setelahnya: 60
Jarak ULBJWL dengan setelahnya: 60
Jarak LBJWLM dengan setelahnya: 60
Jarak BJWLMG dengan setelahnya: 60
Jarak JWLMGS dengan setelahnya: 60
Jarak GFPWTH dengan setelahnya: 40

----- Duplicated 5-ch -----
Jarak ECHYY dengan setelahnya: 350, 820
Jarak CHYYE dengan setelahnya: 350, 820
Jarak HYYEG dengan setelahnya: 350, 820
Jarak YYEGK dengan setelahnya: 350, 820
Jarak HSSLM dengan setelahnya: 640
Jarak SSLME dengan setelahnya: 640
Jarak LMEQW dengan setelahnya: 40
Jarak MEQWZ dengan setelahnya: 40
Jarak EQWZM dengan setelahnya: 40
Jarak QWZMC dengan setelahnya: 40
Jarak WZMCY dengan setelahnya: 40, 810, 790
Jarak ZMCYC dengan setelahnya: 850
Jarak MCYCL dengan setelahnya: 850
Jarak HSSXC dengan setelahnya: 1090
Jarak RBFWR dengan setelahnya: 140
Jarak BFWRE dengan setelahnya: 140
Jarak FWREY dengan setelahnya: 140
Jarak WREYP dengan setelahnya: 140
Jarak REYPW dengan setelahnya: 140
Jarak EYPWY dengan setelahnya: 140
Jarak KLXVP dengan setelahnya: 730
Jarak ISHIE dengan setelahnya: 240
Jarak SHIEB dengan setelahnya: 240
Jarak HIEBT dengan setelahnya: 240
Jarak IEBTQ dengan setelahnya: 240
Jarak IYYPA dengan setelahnya: 2030
Jarak YYPAJ dengan setelahnya: 2030
Jarak AJWOX dengan setelahnya: 1330
Jarak JWOXC dengan setelahnya: 1330
Jarak WOXCY dengan setelahnya: 1330
Jarak OXCYG dengan setelahnya: 1330
Jarak XCYGE dengan setelahnya: 1330
Jarak HNCZE dengan setelahnya: 1290
Jarak HECHY dengan setelahnya: 820
Jarak ULBJW dengan setelahnya: 1610, 60
Jarak EHUZN dengan setelahnya: 810
Jarak MHXPN dengan setelahnya: 1200
Jarak HXPNI dengan setelahnya: 1200
Jarak XDGXH dengan setelahnya: 590, 240
Jarak DGXHS dengan setelahnya: 590, 240
Jarak GXHSS dengan setelahnya: 590, 240
Jarak XHSSL dengan setelahnya: 590
Jarak AUJHS dengan setelahnya: 1720
Jarak UJHSS dengan setelahnya: 1720
Jarak VRCZS dengan setelahnya: 380
Jarak NRSLR dengan setelahnya: 70
Jarak RSLRI dengan setelahnya: 70
Jarak OHMHX dengan setelahnya: 50
Jarak KPGLA dengan setelahnya: 500
Jarak KGRRV dengan setelahnya: 1190
Jarak PNIFP dengan setelahnya: 610
Jarak WWSSG dengan setelahnya: 880
Jarak WSSGT dengan setelahnya: 880
Jarak SSGTF dengan setelahnya: 880
Jarak FEIUW dengan setelahnya: 550
Jarak EIUWW dengan setelahnya: 550
Jarak IUWWC dengan setelahnya: 550
Jarak JKHHZ dengan setelahnya: 240
Jarak GGKBW dengan setelahnya: 140
Jarak GKBWE dengan setelahnya: 140

Jarak KBWEX dengan setelahnya: 140
Jarak LXSAT dengan setelahnya: 140
Jarak XSATG dengan setelahnya: 140
Jarak SATGK dengan setelahnya: 140
Jarak GULBJ dengan setelahnya: 60
Jarak LBJWL dengan setelahnya: 60
Jarak BJWLM dengan setelahnya: 60
Jarak JWLMG dengan setelahnya: 60
Jarak WLMGS dengan setelahnya: 60
Jarak GFPWT dengan setelahnya: 40
Jarak FPWTH dengan setelahnya: 40

----- Duplicated 4-ch -----
Jarak XRKC dengan setelahnya: 1070
Jarak ECHY dengan setelahnya: 350, 820
Jarak CHYY dengan setelahnya: 350, 820
Jarak HYYE dengan setelahnya: 350, 820
Jarak YYEG dengan setelahnya: 350, 820
Jarak YEGK dengan setelahnya: 350, 820
Jarak THHS dengan setelahnya: 2280
Jarak HSSL dengan setelahnya: 640, 590
Jarak SSLM dengan setelahnya: 640
Jarak SLME dengan setelahnya: 640
Jarak LMEQ dengan setelahnya: 40
Jarak MEQW dengan setelahnya: 40
Jarak EQWZ dengan setelahnya: 40
Jarak QWZM dengan setelahnya: 40
Jarak WZMC dengan setelahnya: 40, 810, 790
Jarak ZMCY dengan setelahnya: 40, 810, 790
Jarak MCYC dengan setelahnya: 850
Jarak CYCL dengan setelahnya: 850
Jarak ZSPL dengan setelahnya: 1840
Jarak IHSS dengan setelahnya: 2150
Jarak HSSX dengan setelahnya: 1090, 40
Jarak SSXC dengan setelahnya: 1090, 1120
Jarak CVWO dengan setelahnya: 1180
Jarak HNYG dengan setelahnya: 1370
Jarak OYIR dengan setelahnya: 1725
Jarak ZKTR dengan setelahnya: 501, 1569
Jarak RBFW dengan setelahnya: 140, 1160, 370
Jarak BFWR dengan setelahnya: 140
Jarak FWRE dengan setelahnya: 140
Jarak WREY dengan setelahnya: 140
Jarak REYP dengan setelahnya: 78, 62
Jarak EYPW dengan setelahnya: 140
Jarak YPWY dengan setelahnya: 140
Jarak KLXV dengan setelahnya: 730
Jarak LXVP dengan setelahnya: 730
Jarak POYR dengan setelahnya: 48
Jarak ISHI dengan setelahnya: 240
Jarak SHIE dengan setelahnya: 240
Jarak HIEB dengan setelahnya: 240
Jarak IEBT dengan setelahnya: 240
Jarak EBTQ dengan setelahnya: 240
Jarak RYXI dengan setelahnya: 2070, 150
Jarak IYYP dengan setelahnya: 2030
Jarak YYPA dengan setelahnya: 2030
Jarak YPAJ dengan setelahnya: 2030
Jarak AJWO dengan setelahnya: 1330
Jarak JWOX dengan setelahnya: 1330
Jarak WOXC dengan setelahnya: 1330
Jarak OXCY dengan setelahnya: 1330
Jarak XCYG dengan setelahnya: 1330
Jarak CYGE dengan setelahnya: 1330
Jarak HNCZ dengan setelahnya: 1290
Jarak NCZE dengan setelahnya: 1290
Jarak GLAV dengan setelahnya: 1160
Jarak OKWH dengan setelahnya: 780
Jarak HECH dengan setelahnya: 820
Jarak ULBJ dengan setelahnya: 1610, 60
Jarak LBJW dengan setelahnya: 1610, 60
Jarak ZGBR dengan setelahnya: 940
Jarak EHUZ dengan setelahnya: 810
Jarak HUZN dengan setelahnya: 610, 200
Jarak GSDG dengan setelahnya: 1584
Jarak PPQG dengan setelahnya: 500
Jarak XICB dengan setelahnya: 1720
Jarak MHXP dengan setelahnya: 1200
Jarak HXPN dengan setelahnya: 1200
Jarak XPNI dengan setelahnya: 1200, 160
Jarak IYRG dengan setelahnya: 118
Jarak NCXB dengan setelahnya: 270
Jarak ZFRI dengan setelahnya: 220
Jarak GJFS dengan setelahnya: 190
Jarak PNIB dengan setelahnya: 1600
Jarak HOMA dengan setelahnya: 120, 160
Jarak XDGX dengan setelahnya: 590, 240
Jarak DGXH dengan setelahnya: 590, 240
Jarak GXHS dengan setelahnya: 590, 240
Jarak XHSS dengan setelahnya: 590, 240
Jarak AUJH dengan setelahnya: 1720
Jarak UJHS dengan setelahnya: 1720
Jarak JHSS dengan setelahnya: 1720
Jarak HSSK dengan setelahnya: 390
Jarak WYQO dengan setelahnya: 677
Jarak VRCZ dengan setelahnya: 380
Jarak RCZS dengan setelahnya: 380
Jarak NRSL dengan setelahnya: 70
Jarak RSLR dengan setelahnya: 70
Jarak SLRI dengan setelahnya: 70
Jarak OHMH dengan setelahnya: 50

Jarak HMHX dengan setelahnya: 50
Jarak PWZM dengan setelahnya: 450
Jarak RUIA dengan setelahnya: 1300
Jarak HSOZ dengan setelahnya: 1410
Jarak RUEP dengan setelahnya: 140, 1190
Jarak HXIO dengan setelahnya: 962
Jarak KPGL dengan setelahnya: 500
Jarak PGLA dengan setelahnya: 500
Jarak KGRR dengan setelahnya: 1190
Jarak GRRV dengan setelahnya: 1190
Jarak KRAD dengan setelahnya: 150
Jarak GXWD dengan setelahnya: 660
Jarak PNIF dengan setelahnya: 610, 280
Jarak NIFP dengan setelahnya: 610, 120
Jarak KHHZ dengan setelahnya: 570, 240
Jarak LGFL dengan setelahnya: 60
Jarak YGLW dengan setelahnya: 90
Jarak LMEL dengan setelahnya: 380
Jarak HSSV dengan setelahnya: 710
Jarak WWSS dengan setelahnya: 880
Jarak WSSG dengan setelahnya: 880
Jarak SSGT dengan setelahnya: 880
Jarak SGTF dengan setelahnya: 880
Jarak ZFNX dengan setelahnya: 250
Jarak NEEB dengan setelahnya: 250
Jarak FEIU dengan setelahnya: 550
Jarak EIUW dengan setelahnya: 550
Jarak IUWW dengan setelahnya: 550
Jarak UWWC dengan setelahnya: 550
Jarak JKHH dengan setelahnya: 240
Jarak PNIT dengan setelahnya: 100
Jarak GGKB dengan setelahnya: 140
Jarak GKBW dengan setelahnya: 140
Jarak KBWE dengan setelahnya: 140
Jarak BWEX dengan setelahnya: 140
Jarak LXSA dengan setelahnya: 140
Jarak XSAT dengan setelahnya: 140
Jarak SATG dengan setelahnya: 140
Jarak ATGK dengan setelahnya: 140
Jarak GULB dengan setelahnya: 60
Jarak BJWL dengan setelahnya: 60
Jarak JWLM dengan setelahnya: 60
Jarak WLMG dengan setelahnya: 60
Jarak LMGS dengan setelahnya: 60
Jarak SPNI dengan setelahnya: 120
Jarak TRUI dengan setelahnya: 130
Jarak GFPW dengan setelahnya: 40
Jarak FPWT dengan setelahnya: 40
Jarak PWTH dengan setelahnya: 40

----- Duplicated 3-ch -----
Jarak RCZ dengan setelahnya: 883, 380
Jarak CZE dengan setelahnya: 310, 1290
Jarak XRK dengan setelahnya: 1070
Jarak RKC dengan setelahnya: 1070
Jarak XVP dengan setelahnya: 200, 20, 30, 680, 730, 70, 500
Jarak VPV dengan setelahnya: 543, 1020
Jarak VOQ dengan setelahnya: 880, 1160
Jarak QGS dengan setelahnya: 548
Jarak NES dengan setelahnya: 1857
Jarak ECH dengan setelahnya: 350, 820
Jarak CHY dengan setelahnya: 350, 820
Jarak HYY dengan setelahnya: 350, 820, 994
Jarak YYE dengan setelahnya: 350, 820
Jarak YEG dengan setelahnya: 350, 820
Jarak EGK dengan setelahnya: 350, 820
Jarak EDC dengan setelahnya: 1330
Jarak AGA dengan setelahnya: 650
Jarak QCX dengan setelahnya: 660
Jarak STJ dengan setelahnya: 2123
Jarak JCA dengan setelahnya: 632, 1738
Jarak UUS dengan setelahnya: 1170
Jarak RCE dengan setelahnya: 380
Jarak CEY dengan setelahnya: 1334
Jarak XDK dengan setelahnya: 1020
Jarak KWH dengan setelahnya: 270, 780
Jarak HZF dengan setelahnya: 2080
Jarak ZRL dengan setelahnya: 1727
Jarak ETH dengan setelahnya: 380
Jarak THH dengan setelahnya: 2280
Jarak HHS dengan setelahnya: 2280
Jarak HSS dengan setelahnya: 30, 610, 10, 390, 80, 40, 70, 240, 710, 190
Jarak SSL dengan setelahnya: 640, 590
Jarak SLM dengan setelahnya: 640
Jarak LME dengan setelahnya: 40, 600, 700, 380
Jarak MEQ dengan setelahnya: 40
Jarak EQW dengan setelahnya: 40
Jarak QWZ dengan setelahnya: 40
Jarak WZM dengan setelahnya: 40, 810, 450, 340
Jarak ZMC dengan setelahnya: 40, 810, 790
Jarak MCY dengan setelahnya: 40, 810, 790
Jarak CYC dengan setelahnya: 850
Jarak YCL dengan setelahnya: 850
Jarak CLJ dengan setelahnya: 1528
Jarak JNO dengan setelahnya: 422
Jarak ZSP dengan setelahnya: 1840, 97
Jarak SPL dengan setelahnya: 1840
Jarak GFX dengan setelahnya: 680
Jarak SIH dengan setelahnya: 550

Jarak IHS dengan setelahnya: 2150
Jarak SSX dengan setelahnya: 1090, 40, 1080
Jarak SXC dengan setelahnya: 1090, 1120
Jarak XCV dengan setelahnya: 2065
Jarak CVW dengan setelahnya: 1180
Jarak VWO dengan setelahnya: 1180
Jarak UQS dengan setelahnya: 990
Jarak HNY dengan setelahnya: 1370
Jarak NYG dengan setelahnya: 1370, 595
Jarak YGA dengan setelahnya: 69
Jarak RPP dengan setelahnya: 537
Jarak OYI dengan setelahnya: 1725
Jarak YIR dengan setelahnya: 1725
Jarak GWK dengan setelahnya: 587
Jarak SOZ dengan setelahnya: 790, 1410
Jarak OZP dengan setelahnya: 35
Jarak PRE dengan setelahnya: 1000, 570
Jarak ZKT dengan setelahnya: 501, 1569
Jarak KTR dengan setelahnya: 501, 1030, 539
Jarak RBF dengan setelahnya: 140, 1160, 370
Jarak BFW dengan setelahnya: 140, 1160, 370
Jarak FWR dengan setelahnya: 140
Jarak WRE dengan setelahnya: 140
Jarak REY dengan setelahnya: 78, 62
Jarak EYP dengan setelahnya: 78, 62
Jarak YPW dengan setelahnya: 140, 1935
Jarak PWY dengan setelahnya: 140, 830
Jarak KLX dengan setelahnya: 730
Jarak LXV dengan setelahnya: 730
Jarak GPT dengan setelahnya: 295
Jarak VPO dengan setelahnya: 1470, 500
Jarak POY dengan setelahnya: 48, 1280
Jarak OYR dengan setelahnya: 48
Jarak WEG dengan setelahnya: 1444
Jarak EBO dengan setelahnya: 60, 1420, 170
Jarak BOW dengan setelahnya: 1117
Jarak WYY dengan setelahnya: 930
Jarak ISH dengan setelahnya: 240
Jarak SHI dengan setelahnya: 240
Jarak HIE dengan setelahnya: 240
Jarak IEB dengan setelahnya: 240
Jarak EBT dengan setelahnya: 240
Jarak BTQ dengan setelahnya: 240
Jarak RYX dengan setelahnya: 2070, 150
Jarak YXI dengan setelahnya: 2070, 150
Jarak ETX dengan setelahnya: 1380
Jarak BQP dengan setelahnya: 1201
Jarak ZLS dengan setelahnya: 320, 450, 810
Jarak IYY dengan setelahnya: 2030
Jarak YYP dengan setelahnya: 2030
Jarak YPA dengan setelahnya: 2030
Jarak PAJ dengan setelahnya: 2030
Jarak AJW dengan setelahnya: 1330
Jarak JWO dengan setelahnya: 1330
Jarak WOX dengan setelahnya: 1330
Jarak OXC dengan setelahnya: 1330
Jarak XCY dengan setelahnya: 1330
Jarak CYG dengan setelahnya: 1330, 128
Jarak YGE dengan setelahnya: 366, 964
Jarak MOR dengan setelahnya: 1303, 722
Jarak HNC dengan setelahnya: 1010, 100, 10, 170, 310, 180
Jarak NCZ dengan setelahnya: 1290
Jarak VWE dengan setelahnya: 246
Jarak WEM dengan setelahnya: 1697
Jarak DGL dengan setelahnya: 2168
Jarak GLA dengan setelahnya: 340, 350, 470, 30, 880
Jarak LAV dengan setelahnya: 1160
Jarak GRV dengan setelahnya: 140
Jarak SJG dengan setelahnya: 1706
Jarak GVC dengan setelahnya: 534
Jarak NXW dengan setelahnya: 1340
Jarak PKG dengan setelahnya: 1889
Jarak GMO dengan setelahnya: 1765
Jarak OKW dengan setelahnya: 780
Jarak AGR dengan setelahnya: 750
Jarak GRR dengan setelahnya: 690, 1190
Jarak RHP dengan setelahnya: 1890
Jarak HPN dengan setelahnya: 800
Jarak PNE dengan setelahnya: 750
Jarak HEC dengan setelahnya: 820
Jarak TUP dengan setelahnya: 260
Jarak ZIZ dengan setelahnya: 146, 600
Jarak MYA dengan setelahnya: 900
Jarak CDW dengan setelahnya: 1970
Jarak ULB dengan setelahnya: 1610, 60
Jarak LBJ dengan setelahnya: 357, 1253, 60
Jarak BJW dengan setelahnya: 1610, 60
Jarak WHS dengan setelahnya: 732
Jarak MKY dengan setelahnya: 1540
Jarak KYE dengan setelahnya: 1260
Jarak EWW dengan setelahnya: 1210
Jarak SWI dengan setelahnya: 1455
Jarak IFV dengan setelahnya: 100
Jarak FVW dengan setelahnya: 1570
Jarak ZGB dengan setelahnya: 940
Jarak GBR dengan setelahnya: 940
Jarak YGR dengan setelahnya: 880
Jarak CEV dengan setelahnya: 150
Jarak RVW dengan setelahnya: 591
Jarak EHU dengan setelahnya: 810
Jarak HUZ dengan setelahnya: 610, 200
Jarak UZN dengan setelahnya: 610, 200
Jarak AET dengan setelahnya: 1160

Jarak THP dengan setelahnya: 580
Jarak HPW dengan setelahnya: 476
Jarak CCL dengan setelahnya: 439
Jarak LIS dengan setelahnya: 1224
Jarak LWY dengan setelahnya: 1030
Jarak WDC dengan setelahnya: 1011
Jarak GSD dengan setelahnya: 1584
Jarak SDG dengan setelahnya: 1584
Jarak EVK dengan setelahnya: 100, 1220, 500, 120
Jarak VPN dengan setelahnya: 100
Jarak FVR dengan setelahnya: 730
Jarak GZI dengan setelahnya: 1880
Jarak ZFN dengan setelahnya: 1150, 250
Jarak GOP dengan setelahnya: 1383
Jarak PJK dengan setelahnya: 960
Jarak GFF dengan setelahnya: 550
Jarak FVP dengan setelahnya: 1632
Jarak PPQ dengan setelahnya: 500
Jarak PQG dengan setelahnya: 500
Jarak XIC dengan setelahnya: 1720
Jarak ICB dengan setelahnya: 1720
Jarak MHX dengan setelahnya: 360, 50, 790
Jarak HXP dengan setelahnya: 1200
Jarak XPN dengan setelahnya: 1200, 160
Jarak PNI dengan setelahnya: 120, 30, 440, 300, 310, 60, 100, 120, 120, 150
Jarak FKY dengan setelahnya: 1090
Jarak LSO dengan setelahnya: 1290
Jarak IYR dengan setelahnya: 118
Jarak YRG dengan setelahnya: 118
Jarak NCX dengan setelahnya: 270, 552
Jarak CXB dengan setelahnya: 270
Jarak BRH dengan setelahnya: 806
Jarak RHQ dengan setelahnya: 644
Jarak QVX dengan setelahnya: 290
Jarak ZFR dengan setelahnya: 220, 70, 450, 180
Jarak FRI dengan setelahnya: 220
Jarak TCT dengan setelahnya: 360
Jarak OTB dengan setelahnya: 1470
Jarak AJY dengan setelahnya: 880
Jarak IVP dengan setelahnya: 1025
Jarak VPK dengan setelahnya: 1565
Jarak GJF dengan setelahnya: 190
Jarak JFS dengan setelahnya: 190
Jarak XAR dengan setelahnya: 833, 480
Jarak ARW dengan setelahnya: 660
Jarak NIB dengan setelahnya: 1600
Jarak HOM dengan setelahnya: 120, 160
Jarak OMA dengan setelahnya: 120, 160
Jarak MAX dengan setelahnya: 1295
Jarak XDG dengan setelahnya: 590, 240, 915
Jarak DGX dengan setelahnya: 330, 260, 240
Jarak GXH dengan setelahnya: 590, 240
Jarak XHS dengan setelahnya: 590, 240
Jarak AUJ dengan setelahnya: 1720
Jarak UJH dengan setelahnya: 1720
Jarak JHS dengan setelahnya: 1720
Jarak SSK dengan setelahnya: 390, 105, 659
Jarak SKP dengan setelahnya: 747
Jarak REH dengan setelahnya: 967
Jarak RMJ dengan setelahnya: 570
Jarak WYQ dengan setelahnya: 670, 7
Jarak YQO dengan setelahnya: 677
Jarak EID dengan setelahnya: 856
Jarak IOE dengan setelahnya: 450
Jarak OEV dengan setelahnya: 1000
Jarak ITZ dengan setelahnya: 1093
Jarak ZMB dengan setelahnya: 560
Jarak BVR dengan setelahnya: 1420
Jarak VRC dengan setelahnya: 380
Jarak CZS dengan setelahnya: 380
Jarak SRY dengan setelahnya: 1593
Jarak YOK dengan setelahnya: 255
Jarak ATG dengan setelahnya: 1113, 140
Jarak TGS dengan setelahnya: 1200
Jarak OYY dengan setelahnya: 522
Jarak YYZ dengan setelahnya: 1252
Jarak NRS dengan setelahnya: 70
Jarak RSL dengan setelahnya: 70
Jarak SLR dengan setelahnya: 70
Jarak LRI dengan setelahnya: 70
Jarak OHM dengan setelahnya: 50
Jarak HMH dengan setelahnya: 50
Jarak HXK dengan setelahnya: 361
Jarak VPH dengan setelahnya: 1245
Jarak PWZ dengan setelahnya: 450
Jarak RUI dengan setelahnya: 390, 780, 130
Jarak UIA dengan setelahnya: 1300
Jarak HSO dengan setelahnya: 1410
Jarak RUE dengan setelahnya: 140, 1190
Jarak UEP dengan setelahnya: 140, 1190
Jarak HXI dengan setelahnya: 962
Jarak XIO dengan setelahnya: 962
Jarak HKP dengan setelahnya: 1220
Jarak KPG dengan setelahnya: 500
Jarak PGL dengan setelahnya: 500
Jarak LAK dengan setelahnya: 855
Jarak AHO dengan setelahnya: 644
Jarak GPR dengan setelahnya: 735
Jarak LSY dengan setelahnya: 264
Jarak KGR dengan setelahnya: 1190
Jarak RRV dengan setelahnya: 1190
Jarak WTH dengan setelahnya: 590, 690, 40
Jarak KRA dengan setelahnya: 150

Jarak RAD dengan setelahnya: 150
Jarak GXW dengan setelahnya: 660
Jarak XWD dengan setelahnya: 660
Jarak QII dengan setelahnya: 876
Jarak KWG dengan setelahnya: 480
Jarak FFK dengan setelahnya: 504
Jarak KZL dengan setelahnya: 900
Jarak KYM dengan setelahnya: 1345
Jarak YAH dengan setelahnya: 130, 99, 261, 320
Jarak NIF dengan setelahnya: 610, 120, 160
Jarak IFP dengan setelahnya: 610, 120
Jarak FPW dengan setelahnya: 1165, 40
Jarak UIO dengan setelahnya: 73
Jarak KRH dengan setelahnya: 30
Jarak RHS dengan setelahnya: 676
Jarak KHH dengan setelahnya: 570, 240
Jarak HHZ dengan setelahnya: 570, 240
Jarak WYF dengan setelahnya: 275
Jarak YFV dengan setelahnya: 740
Jarak IIN dengan setelahnya: 700
Jarak LGF dengan setelahnya: 60
Jarak GFL dengan setelahnya: 60
Jarak FLP dengan setelahnya: 187
Jarak OEH dengan setelahnya: 1155
Jarak DCW dengan setelahnya: 60
Jarak IBE dengan setelahnya: 980
Jarak GZF dengan setelahnya: 320
Jarak VOK dengan setelahnya: 516
Jarak OKH dengan setelahnya: 60
Jarak MBR dengan setelahnya: 40, 560
Jarak MJM dengan setelahnya: 810
Jarak YGL dengan setelahnya: 90
Jarak GLW dengan setelahnya: 90, 880
Jarak LWT dengan setelahnya: 210
Jarak MEL dengan setelahnya: 380
Jarak BRF dengan setelahnya: 520
Jarak VPF dengan setelahnya: 210
Jarak CGL dengan setelahnya: 930
Jarak CFE dengan setelahnya: 853
Jarak JKP dengan setelahnya: 140
Jarak XIW dengan setelahnya: 967
Jarak CCV dengan setelahnya: 396, 174
Jarak PHN dengan setelahnya: 380
Jarak GXA dengan setelahnya: 903
Jarak SSV dengan setelahnya: 710
Jarak PVR dengan setelahnya: 260
Jarak CYF dengan setelahnya: 490
Jarak WWS dengan setelahnya: 880
Jarak WSS dengan setelahnya: 880
Jarak SSG dengan setelahnya: 880
Jarak SGT dengan setelahnya: 880
Jarak GTF dengan setelahnya: 880
Jarak FQZ dengan setelahnya: 584
Jarak FNX dengan setelahnya: 250
Jarak COL dengan setelahnya: 396
Jarak NEE dengan setelahnya: 250
Jarak EEB dengan setelahnya: 250
Jarak EHR dengan setelahnya: 330
Jarak PGZ dengan setelahnya: 88
Jarak FEI dengan setelahnya: 550
Jarak EIU dengan setelahnya: 550
Jarak IUW dengan setelahnya: 550
Jarak UWW dengan setelahnya: 550
Jarak WWC dengan setelahnya: 550
Jarak JKH dengan setelahnya: 240
Jarak ZAX dengan setelahnya: 380
Jarak VRP dengan setelahnya: 460
Jarak NIT dengan setelahnya: 100
Jarak OYP dengan setelahnya: 70
Jarak LSQ dengan setelahnya: 576
Jarak GGK dengan setelahnya: 140
Jarak GKB dengan setelahnya: 140
Jarak KBW dengan setelahnya: 140
Jarak BWE dengan setelahnya: 140, 178
Jarak WEX dengan setelahnya: 140
Jarak AZG dengan setelahnya: 540
Jarak BZK dengan setelahnya: 330
Jarak PUH dengan setelahnya: 271
Jarak LXS dengan setelahnya: 140
Jarak XSA dengan setelahnya: 140
Jarak SAT dengan setelahnya: 140
Jarak TGK dengan setelahnya: 140
Jarak GUL dengan setelahnya: 60
Jarak JWL dengan setelahnya: 60
Jarak WLM dengan setelahnya: 60
Jarak LMG dengan setelahnya: 60
Jarak MGS dengan setelahnya: 60
Jarak XSJ dengan setelahnya: 180
Jarak SPN dengan setelahnya: 120
Jarak ETG dengan setelahnya: 120
Jarak IZH dengan setelahnya: 380
Jarak TRU dengan setelahnya: 130
Jarak SOO dengan setelahnya: 16
Jarak MGE dengan setelahnya: 235
Jarak EJJ dengan setelahnya: 360
Jarak JXC dengan setelahnya: 160
Jarak TUN dengan setelahnya: 50
Jarak GFP dengan setelahnya: 40
Jarak PWT dengan setelahnya: 40
Jarak AXM dengan setelahnya: 60

Dari kumpulan *substring* berulang beserta jaraknya tersebut, dapat kita lihat bahwa seluruhnya merupakan kelipatan 10 (FPB). Maka, kunci kemungkinan memiliki panjang 10 karakter.

Selanjutnya, kita dapat mengetahui trigram yang paling sering muncul dalam cipher text berikut dengan memanfaatkan kode pada B.1 sehingga diperoleh

- Trigram paling banyak muncul
  HSS: 11, PNI: 11, XVP: 8, HNC: 7, GLA: 6

Berdasarkan **Gambar 1.** Top 10 huruf dalam teks Bahasa Inggris, kita juga dapat mengetahui trigram berbahasa inggris paling sering muncul adalah THE. Sehingga kita dapat berasumsi bahwa kelima trigram di atas adalah THE.

Selanjutnya, kita kelompokkan cipher text tersebut setiap 10 karakter. Bagian yang mengandung kelima trigram diatas akan dianalisis untuk mengetahui kunci yang digunakan.

```
trigrams = text_counter(CIPHER_TEXT_2, 3).most_common(5)
grouped = findall('.' * 10, CIPHER_TEXT_2)

trigrams = {trigram[0]:[str10 for str10 in grouped if trigram[0] in str10] for trigram in trigrams}

for trigram in trigrams:
  print(f"--- {trigram} ---")
  print("\n".join(trigrams[trigram]))
```

— HSS —
cipher ← key ← plain (hasil dekripsi utuh)

- RLETHHSSLM ← *****OLO** ← *****THE** (EHINDTHEFO)
- FXESIHSSXC ← *****OLO** ← *****THE** (STIMETHERE)
- AXDGXHSSLM ← *****OLO** ← *****THE** (NTHATTHEFO)
- EGAUJHSSKP ← *****OLO** ← *****THE** (RCEOFTHEER)
- NYOKWHSSKK ← *****OLO** ← *****THE** (AUSESTHEEM)
- RADKRHSSXC ← *****OLO** ← *****THE** (EWHENTHERE)
- VRCZSHSSXG ← *****OLO** ← *****THE** (INGTOTHERI)
- QXDGXHSSLG ← *****OLO** ← *****THE** (DTHATTHEFI)
- RXDGXHSSVP ← *****OLO** ← *****THE** (ETHATTHEPR)
- YPWMIHSSVS ← *****OLO** ← *****THE** (LLAGETHEPU)
- RVAUJHSSRY ← *****OLO** ← *****THE** (EREOFTHELA)

— PNI —
cipher ← key ← plain (hasil dekripsi utuh)

- HXPNIGAFKY ← **WGE***** ← **THE***** (UTTHESPREA)
- RPPNISGWJC ← **WGE***** ← **THE***** (ELTHEEVIDE)
- RWPNIBZHOM ← **WGE***** ← **THE***** (ESTHENOTIO)
- AHPNIFPWYY ← **WGE***** ← **THE***** (NDTHEREISA)
- BQPNIQFFYC ← **WGE***** ← **THE***** (OMTHECURSE)
- HXPNIFPGZS ← **WGE***** ← **THE***** (UTTHERESTU)
- VRPNITTSRB ← **WGE***** ← **THE***** (INTHEFIELD)

- NXPNITZCJG ← **WGE***** ← **THE***** (ATTHEFOODI)
- ZSPNIFTTNC ← **WGE***** ← **THE***** (MOTHERIFHE)
- GSPNIVTUNF ← **WGE***** ← **THE***** (TOTHEHIGHH)
- XIPNIBEVKG ← **WGE***** ← **THE***** (KETHENTHEI)

— XVP —
cipher ← key ← plain (hasil dekripsi utuh)
- XRKCXVPVOQ ← ****EOL*** ← ****THE*** (KNOWTHEHIS)
- AIKLXVPZGP ← ****EOL*** ← ****THE*** (NEOFTHELAR)
- AWKAXVPOYR ← ****EOL*** ← ****THE*** (NSOUTHEAST)
- XIETXVPAOB ← ****EOL*** ← ****THE*** (KEINTHEMID)
- HXKLXVPHUR ← ****EOL*** ← ****THE*** (UTOFTHETOT)
- LFQZXVPFKY ← ****EOL*** ← ****THE*** (YBUTTHEREA)
- QEHRXVPTUM ← ****EOL*** ← ****THE*** (DALLTHEFOO)
- AMJMXVPKGR ← ****EOL*** ← ****THE*** (NINGTHEWAT)

— HNC —
cipher ← key ← plain (hasil dekripsi utuh)
- EMOREBOHNC ← *******OGY ← *******THE (RISLANDTHE)
- YJEYLPFHNC ← *******OGY ← *******THE (LFISHBUTHE)
- BPWZMBRHNC ← *******OGY ← *******THE (OLATINGTHE)
- EYHKWCQHNC ← *******OGY ← *******THE (RULESOFTHE)
- VWDLVCXHNC ← *******OGY ← *******THE (ISHFROMTHE)
- RHSNIFPHNC ← *******OGY ← *******THE (EDWHERETHE)
- BJWLMGSHNC ← *******OGY ← *******THE (OFAFISHTHE)

— GLA —
cipher ← key ← plain (hasil dekripsi utuh)
- GLAVSDFZGR ← NEW******* ← THE******* (THEPOPULAT)
- GLAJYGEPAP ← NEW******* ← THE******* (THEDUSTBUR)
- GLAKVIAHOM ← NEW******* ← THE******* (THEERUPTIO)
- GLAVVWYQKQ ← NEW******* ← THE******* (THEPRINCES)
- GLAXIWDCTC ← NEW******* ← THE******* (THEREISONE)
- GLANMGECXW ← NEW******* ← THE******* (THEHISTORY)

Saat potongan *key* tersebut digabungkan, akan terbentuk sebuah kata NEWGEOLOGY. Ini berpotensi sebagai *key* yang benar. Untuk mengujinya, kita dapat mendekripsi potongan *cipher text* diatas menjadi sebuah potongan *plain text* yang utuh. Dapat dilihat pada bagian yang di dalam kurung.

Hasil dari dekripsi potongan *cipher text* tersebut memiliki penggalan kata-kata dalam bahasa inggris sehingga *key* ini merupakan kunci yang tepat dan kita dapat mendekripsi keseluruhan *cipher text* tersebut.

Hasil Dekripsi

SOMEPEOPLEMAYALREADYKNOWTHEHISTORYOFTOBALAKELOCATEDINNORTHSUMATRAPROVINCEBUTFORTHOSEOFYO
UWHODONOTKNOWTHESTORYBEHINDTHEFORMATIONOFLAKETOBATHISTIMETHEREISSOMEINFORMATIONTHATWILLBED
ISCUSSEDCURIOUSASWHATREADITDOWNTOBALAKEISONEOFTHELARGESTLAKESINSOUTHEASTASIAANDISAVOLCANICL
AKEINTHEMIDDLETHEREISANISLANDCALLEDSAMOSIRISLANDTHEMAJORITYOFTHEPOPULATIONAROUNDTOBALAKEISBA

TAKTRIBEITISESTIMATEDTHATTOBALAKEWASFORMEDDURINGANEXPLOSIONOFABOUTYEARSAGOWHICHISANERUPTION
SUPERVOLCANOSUPERVOLCANOTHATISMOUNTTOBAWINDBLOWNVOLCANICASHHASSPREADTOHALFTHEEARTHFROMCH
INATOSOUTHAFRICAEVENQUITESURPRISINGBECAUSEITTURNSOUTTHESPREADOFTHEDUSTTOBERECORDEDUPTOTHENO
RTHPOLETHEERUPTIONOCCURREDFORONEWEEKANDTHEDUSTBURSTREACHEDKILOMETERSABOVESEALEVELTHEEVIDE
NCEFOUNDALSOREINFORCESTHENOTIONTHATTHEFORCEOFTHEERUPTIONANDITSOCEANWAVESCOULDANNIHILATELIFEI
NATLANTISTHISINCIDENTCAUSEDMASSDEATHFOLLOWEDBYTHEEXTINCTIONOFSOMESPECIESACCORDINGTODNAEVIDEN
CETHISERUPTIONALSOSHRANKTHENUMBEROFPEOPLETOABOUTOFTHETOTALHUMANPOPULATIONOFTHEEARTHATTHATT
IMEABOUTMILLIONPEOPLEAFTERTHEERUPTIONACALDERAWASFORMEDWHICHTHENFILLEDWITHWATERANDBECAMEWH
ATISNOWKNOWNASTOBALAKEUPWARDPRESSUREBYTHEMAGMATHATHASNOTYETCOMEOUTCAUSESTHEEMERGENCEOFS
AMOSIRISLANDTHEREISALSOFOLKLOREABOUTTOBALAKEISSAIDHESAIDATIMEWHENTHEREWASAFARMERNAMEDTOBA
WHOWENTFISHINGTOTHERIVERTOGETFISHTOEATTOBAGETSABIGANDBEAUTIFULFISHBUTHEISSURPRISEDTHATTHEFISHC
ANTALKAPPARENTLYTHEFISHISTHEINCARNATIONOFAPRINCESSWHOWASCURSEDFORVIOLATINGTHERULESOFTHEKINGD
OMASATHANKYOUFORHAVINGRELEASEDHERFROMTHECURSETHEPRINCESSWASMARRYTOBAHOWEVERTHEREISONEPRO
MISETHATHASBEENAGREEDUPONTHEYSHOULDNOTTELLANYONETHATTHEPRINCESSISAFISHFROMTHEMARRIAGEWASBO
RNABOYNAMEDSAMOSIRSAMOSIRGREWINTOAVERYHANDSOMEANDSTRONGBOYBUTTHEREAREHABITSTHATAMAZEEVE
RYONEHEALWAYSFEELSHUNGRYANDNEVERSATISFIEDALLTHEFOODRATIONSAREALWAYSDEVOUREDWITHOUTTHERESTU
NTILONEDAYSAMOSIRASSIGNEDTODELIVERFOODFORHISFATHERINTHEFIELDBUTTHEFOODNEVERCAMETOBAALSOAPPR
OACHEDSAMOSIRANDASKEDWHERETHEFOODFORHIMBUTSAMOSIRADMITTEDTHATTHEFOODISALREADYEATENTOBAWA
SVERYANGRYANDUNKNOWINGLYBREAKSHISPROMISEBYSAYINGSONOFAFISHSAMOSIRIMMEDIATELYCOMPLAINEDTOHIS
MOTHERIFHECALLEDASONOFAFISHTHEPRINCESSWASDISAPPOINTEDBECAUSEHERHUSBANDHASBROKENTHEPROMISESH
ECRIEDALOTANDTOLDSAMOSIRTORUNTOTHEHIGHHILLSUDDENLYVERYHEAVYRAINCAMEDOWNWITHATERRIBLELIGHTN
INGTHEWATEROVERFLOWEDTODROWNTHEENTIREVILLAGETHEPUDDLETURNEDINTOALAKETHATISNOWCALLEDTOBALA
KETHENTHEISLANDWHERESAMOSIRSHELTERISCALLEDSAMOSIRISLANDTHATSTHEHISTORYOFTOBALAKETHATSSOUNIQU
ERIGHTIFYOUVISITTOBALAKEYOUCANFEELTHECOOLATMOSPHEREOFTHELAKEACCOMPANIEDBYBEAUTIFULVIEWSOFSA
MOSIRISLAND

IF4020 Kriptografi
Semester 2 - 2022/2023
Dosen Pengampu : Dr. Ir. Rinaldi Munir, M.T.

Tugas Kecil 1
Nama : Bintang Fajarianto
NIM : 13519138

3) Kriptanalisis *Playfair Cipher*

Kriptogram

QUKAROQULALPKHBUSHPLIWIDCSCYGRBAUXSHBUSHAGCFHZQCQBWUZCBKECIVDGFQDGFAEALASHBPKNPOBLHZFXF
MBCFBMEALALXDUGWUZHXDFQFTLUSHKNLVCSANSHXDUGWUVCMOCLCSENMLKFHEQUVFUGZDGDMBZSCZEMZHXDF
QFTIDPWPCGRDQRUQCBLCZGROWVCRVBLHZUQZOSHXKDKFAILKBKGFQKBXDBLBLFBKZCHHAFTLUIBKZCHUPQMQCTH
PWWOEAIVDTQPBUSHQBWUFTLUSHBKIDPWPCCHXKABNVROQUBLCZLGBAQBWUFTHOSHBLWCHLRVUMSHBPAHCYWCDI
BLLAGLCLCKLGDIEMZDLRACPZQBWUEMBKLCZEDMWOFTLCZEMLKFEMBLCZPWWOQCBLCZGROWYCTKSAMPQUCEBAN
ULEBMSHLELCDGRVAMCHCTHLBAUXDSVCMEOZLGBAEMQBWUFTLCPBCHEMTHPWWOFTILKBLEPKVCMEZEMLKFEMHE
QUVFUGZSCZEMBLKBAEPUAKLMHAIDLRAMHZUQXYZDLRACUGZDFBBLAKLGDIVCZEPUBUSHONKZFBIGCHCEBATKZFKT
LEIOEMZHLUSHIPFQUMPCUMLGOMAHECUQBLKBXDEAZSCZEMETACRUGKTDUWVLBTRUXKWCLCUGLENDLBVFUGZSAK
WCDIUQUTACNULEALFNQUKZAHZHZSWCDIOBKXEMZSKBGREHCTNVZSHSCLCVQUHCNKTDDMPCGRIYHDEMMEUGZDFB
ZSHZUMWOGRKNQUUMKGALUGDMPCCHLASHBPAHKGRGALUGMEGIDMPCGRIYHDEMMHPWPCGRIYHDEMMPBUSHVCIV
DVDSGRBLHPMLUGHPBLAEKLBKMBQCLMMASHDMPCGRIYHDEMMEBQZCBKLUDSCWONKZFBKTRVBLAEKLBKMBQCLM
MASHBQSHVCPZMERLQEHKGCPUZSAFBFVULERVBLAHPWPCGRIYHDEMPZMQPCABSHUNRUSERUQBWUZDLRCZEMINWU
GRDNMKUGINCZOPBKFTLOPUHEACRUGKTDFTOUSHBQSHVCPZZHZAFNACBPMDHOSHVCMEZEACRUGKTDEMBLCHFVFTL
OPUHEACRUGKTDBLFTBLFSBCFBSHBPAHECUGQUKAZDFBBLFSBCFBEMFAILKBDMPCGRIYHDEMMHPWPCGRIYHDEMMHP
WPCGRIYHDEMMPBUSHVCIVDVDSGRBLHPMLUGHPBLAEKLBKMBQCLMDGALBQZCBKLUDSTCNUECPKKBBLFTLMDVDSK
THSKBAHCLNYBCFBTRAVPUHEACRUGKTDMBGDDGPZQBWUZDLRACPHZCFBBLFBBLFSBCFBVCPZZSWDFALCUGURNBCH
ZFIVDMVCUGINKZFBXDBLFBWZKNPOBLLADSRNZDFBAZKBZSAHPWPCCHLAQBWUZDLRHSKBSHKTBLACZAIGBGKBTNA
CMBZSHSLACHBLFHPKHMPOCDLEUGFBWZFTIOKPHCINCZLGHLKBFTILGHSHKPBGKBSHKTAMPIACZCCHMPBLLAZSHSCL
UQSKCKCHFVPURGHLBMSHKCIVHALCDBBUMOFTFTHOKLLUCFABSHEDGDLCVCXGBUMUSHCQBKMBDXGDGTMDQUHFBI
LTHSKBGRQVPHHLRAGDDLKBFTHUKPFQSATEKFSHBIACNUEMLAPCFBBLHSRBBMVLUQEHZFLCUGUMPCSHTKVCZNRUZD
FBSHLEIWIUCVBGHPFRGKACUGVCMEUSRUUMXPKBKAQBWUZDLRHZGRUQKABUSHKTBMSHEHLFBUWUKZKSFTHOKLLC
LIACBIRBBMVLTGACZASAQVDGEADXGDITRBCVFVGRUMBLCZGDGCPUHZALUGBFZAUKPOHSPOACLFGRFTLCPZZSCZEM
MPEHCTRLCNMASHDBBUMOFTLEIOLRWQBGKBFTHOKLIURBCVFVBLFBWQROPWPUEDGDLUSHXDDIEHEMBMXDIVHMPW
QULEZAHCWMRNCHWQRUDSNDLBSHFTGWABLRZHASCHEMGKEMUMBFLEHUKZFBUPIVHAGCUPSLHPSHKTBMGRKALRV
CQBMDQUHFNBKZCHCNGAPULGVAPCGDDLKBFTLCMEWQRBHNMCKPBORVBLFBKCSANRRYRGFQNZSHMBETEHIGALUGD
CUMRULRZSACNUFTLUPWPCLRUMBLFSBCFBEMXKPWQCBLLABLAKLGDIVCOHACLUSHDWPCLELUPMLMRVBLHPFTLUSH
XDKFCHONKZFBLAKHALUGBLWCHLBMSHLEHCLAFBBLHSACNUQCQBWUZDLRHERUUQVUSHQPPUEDGDOUEMLMMLDM
BFGRBLFHBKLACQBKMBFTHOMUKBKPKGCHBLLAZSAHPWPCCHLAAZKBVGACCQEAZANXUNKZACUGIGGUMUFRWUZDL
AUMOWBLFBZSHEFHFTUMNRTFUGTKUQZABPLEKXMBQUHZFTLUIBWCHLBLPUBQFTLUSHXDDIEHEMBMTKRQEMBLKBTR
ULGISHNBSDKTOMSHTKTFMHQULFZSKBEAABUWTDIHWCBUGDLRVCMEMVPMBAZARGFQBLFBZSHZCVLMLRAMAKMUA
LHFFTOUFBRLBUPDRVUMBKMBSHBPAHEAZANXBLGRHGBUSHDWRVALHFVCPZMOLMHTRUBFLAKBFTODHAFTKBBLFTKC
BTABMDLUSHBKVCMEALTHCSUYLGEDVCMEZDLRAKMWKBFTHOAZFBDGRVAMKSMLBAMEUGQUACEDMILAACDGRVAM
HZQPQUOPBKMWKBBLKBCZKBHZAZFBDGRVAMAFRGZAMEUGCQKIMEUGZHZATGLRAMASNDMELFGRMBBLCZLMHMUM
WOZHZATGLRAMHECHPZPHHCNZPHHCFTODMWKBCHXDBLQBWUZDLRAKWCBUKBMKBFMHLRAMHZUQXYZDLRCZEMM
PDBFQNXFTLULGBPNVBLLAZSACBAMPSAZAHFALTHHZFTHUBKCLKNPOACEDMILAACUGIUFTFCIUPQQCXDBFWCUMEAC
KLSNDCQBPSHIGGUMUXDZSUMALFALMGAUPOPBFQUGVBLFBEMZSCZEMMPRLBKHOSHMLGISHBPAHCGWUUQUQBFQCB
LAHLRAMHECHZERUUQBFQCBLASOBLRCHSHUPBFVCMBPWMDQULFUQBLKBUMPMDSNUTRALAMEMBMSHCQRBLULERV
BLHZAZFBBLQUCZEMKPHLALUGMPZSKBASKLBFIGRGBMKLLCXKFVKBFTOUFBRVUMMHKZKSZDFBLRVCMSRDBALCKIBL
CSYUKBMPANSHDWRVALHFKCSATEKFBLFZZCBKZHZAUGVCNKOZDSOCLFUGHCMTCHGRBLASWCBUKLBQCQUPBKBLFZD
GRVBLKHBPAZFBKCUGHAMEUGEMLRVCQBWUDGHLUVLRLGBUSHDWRVALHFBLFZMEVULEBMPOCSBFLEHUKZFBBLKBA
EPUAZKBVNLGFQVCQBMUGRBLHZLVCSUIKBGRZDDKLBSHZGCLHAANLRBLAEPKMBMEGIZDSDEMEHFBPHRVRLFQQUQV
EDKLNUSHZHDWUGZDSDEMEHFBACRLFQQCAQXQMLBABLABLGUIZDSDEMEHFBALIPFQMPSAQCQVTNACLUSHCQRNBLZ
DSDEMEHFBACRLFQQVZAHFALTHHZBLAEGKBLZDSDEMEHFBBOGRLQNGPUARSHPHVRDZDSLEZEKZFBDCRVRLFQXDBLQ
VFRGKABSHZHZABUDZDSLEZEKZFBIUGRIPFQWCUMAQVQNDBLFBBLAEPKMBDMPCCHUPRBLCUGMEVULERVBLKBCZEM
LMLRAMHSMDCSBGHPEHLCUGSHMEGIZDSDEMEHFBHFRVRLFQGRUMQVEHLUSHMUSHBPKCSAROFTHCCKDELMHAOWR
UMPSAUSGWLEZEKZFBGVGRHGTQZNCHUMSURUBLHZAZFBBLZDFBSHDMPCCHLALGZECHMEVDLRAMHZUQXYZDLRCZD
SVCMHXGFQEMHAKSBLKBKHTFUGSHKCIVHALUSHMUSHBPZDWDLVKBRBRGFQROFTLUSHWFKPCHWCUVLRLEMBQUHPL
SFBALUGTNWCHLRVBLKHBPAZFBDGRVAMAKFTIDHZFTIDABBLAHGHLBECIVDMTGLRAMHZUQXYZDLRCSIZSHLTFBMPIZ
SHLTFBOEKPCHBLFZZDLALMMWAHWFLGDIFTLUSHZXKBHZQDMLLULERVBLFZIGGUMUVCFASHNRSRRUHABLKBMHKZK
SQUGRBLHECHFTLUSHZHZABULIVCNKMHKSUVLRIZLGOEPOMKRGQUMPBFDSRNXDBLACEDFTHOQDMUBLUPNDLBSHRG
HLUVSHRGUVEMOPNBGRHDLRAMHZUQXYZDLRACUWFAFTOUEMKNGHLBFBCHZDFBZSHZCVBLHZAZFBOUKTEQTRRVBL
FZZCBKKNHKUGGVFTLCPECHSHUXLEBLKBTFSAVCMONFLVKHPZQBWUZDLRHZSHFTOZKBCHDSZDLTFZWCDWSAUMWC
BIWCZHMEGIBLCHVCNKMBSHBQSHUMDIBLASBLLASHBPBAQOMUSHBICLXDZSCHUMLEZASHKIGDHAHURUBLLABLHSR
BBMVLBGCLMQKTCHSHTIGKACUGBLLAZSHSCLNRFGPUBLCZDSHALCWNBUGDLAMLMBZSHSCLCQRBLUSHPKOUAHDGF

QBLCHVCNKPZMEGIGKWQCUGDMACFAKKTCSCQRNDSBOAKPWPCVLFABLHECHOZEMSZCZFTICRGALUGGKWQCUGDIAK
ZKSAZKBVNLGFQBFLAFTODHAFTWQCDFTMBFVXDBLBOFTGWWCZSLMDVFTNAPUUQBLGRQNHZMEGIQBWUZDLRCZLRL
EIVMVZGACNUFTHOSHMBFVCHXDBLBLASZSAFKSRVBLHSWCZHGRPUHCNKPUBLASGRBLASPURGFQMBSHZXFBRVUMBL
ASWCBUKLBQFTIDPWPCCHCQUKPQSKTKUGDQDIGRBLKZAZFBGRHQBLFZKCSATEKFFTLUSHBUSHPKOBXRSKBICLUMEH
BKCLVNSHHGTIVCMEDMBFBLCZDSHALCZQMUSHDQWOOUKTEQVCNBCHSHTKUGMEGIEHVCBKQCWQRNMBKSOPBLKBZ
SCZGDHMPWQUFRWUBLLAWQTCBKSHBKEHOBBKUMHAUBPWQUAKPOKHBTRUBLFSBCFBEHDGAZGRLNLEBMSHLEHCLA
FBQBWUZDLRHZSHKTUAWCDIUQUTRUBLGRARLEBMSHVCMELFGRBLAEPKMBFTLOPMUTACRUGKTDQCLMMLUGZSCZFB
RVUMSHTIPWPCGRIYHDEMMEUGMEGIDMPCGRIYHDEMMHPWPCGRIYHDEMMPBUSHVCIVDVDSGRBLHPMLUGHPBLAEKL
BKMBQCLMMLUGBLFHMLPZMEBQZCBKOWGSBCFBBLWCKTNAKLBKMBQCLMDGZHHERUMWKBBLHSGDHMZDKBABSHZ
HZAUGVCNKOHZCIVHMUQXYZDLRKHPZMBGDMLDGZAFTOUAHMLUGUQBFHPMPECPKEMZSABSHBQSHVCMEMBWCUGC
HCQBPSHFRCZBLLABLAHPWPCGRIYHDEMQBAZKBMQPCAELMZHGVFTHOSHFRCZBLLABLHSRBBMVLBGCLEHUNFVCHSH
TKUGBLLADGRVAMHZUQXYZDLRCZEMMBGDMLDGZAFTHOPMSHBLWCHLALUGBLWCHLALLFGRDSZOSHPLHLRAGDDLK
BCQBPMDQULFPZZSCZEMUQRVBLAEKLBKMBGRBLCZDSHAMLUGFBWZHABLKBLEZAUQBKMBFTOUSHBQSHLELCAMEMR
VBLWCHLUMEQPOABLGFQUMOWZSKSKLBUCHSHNKCKACUGIUHZZHILKBZHGADGFAFTMDIOCHHAUXPOFTFTGUPWQUA
KWCDILEZAFRWUBGKBGRBLHPIGQHRDZHZSCZFBUMZANUSHZHZATQWCBUKLBQUMBLHZAZFBOUKTEQFTICUQKFCHLA
BLCHPWPUFTODKPCHXKABNVBLGRHQUMZHIVMTKBWFSHKMZANXEDACMILRAMHZUQXYZDLRAHPWPCCHWCUMANSH
XDUGWUFTODLMHAOWRUDQWOLCVQNQPWWOUWVLUYLEBLLTFZWCUMZHIVDHPWWOBLGRHQXKABNVBLGRHQZSAC
BQZCBKHOALVGCKHZQCLMDAMDWCBPFTHOSHORIVHAOWRUQUCZLGEDVCZETKGICHKNPOZFIVDMXUCHFTHUDTHZGD
IPVLVGABBLCZPUBLWQDIUWVLRGBUDSNDLBQBWUZDLRACUGZSCBBTMDBALUSHOWPUFTHUWCHLRVBLKSBKRVNVML
KAOELGDIMEGIBLCSDIUWVLUYLEALKNGHLBWQTDPWPCDWSADXGDIVMLKAWQROUPSKTINGPUQUKAQBWUZDLRHSCL
UQOBMQDKDPBTRUMBPWWOEHCQBKSHTKUGHABLKBZHGAEHMLKAOULRLBSHBFVDCKHZTRRVBLCSDIUWVLTGCKCH
MPONDKAIVFUGMPRLHLAMVNLEBMUQXYZDLRAHPMBLKBTNACBLFTGCAHDIWUTFPHCGACGUWULKVASHOPMBEHCTR
LCNHMKLLUSHONKZFBUMSHBPAHECUGTKTFVCNFMUDMFQCEBAUXKTHOGRBLKZAZFBGRLNSHZHZAUGVCNKOELVHSP
OHERUDSZODSKFCHBLFZZCBKZDFBBLFZMEVUSHPKHCKTDGRVAMHZUQXYZDLRAHXGFQQUBLFHUPRBLCUGBLLAZSFB
KHBLKBMBPKBKGUPUOPZALCUGZHASCHUMEHHCCLBLFZDGANCHSHNRMKUGEMBLFZMEVULEBMSHVCMHCKCHUMMU
GHLBGVBLFZDCRVBLAHCKHZBLFBZSHEFHFTUMTNACBLACDGRVAMACUGCEBATKZDGDAKLVABMDGKACLFGRZDFBBLC
HVCNKZSACUIZDLALEILKMSKBFLUSHZQKLLUSHMDIOCHHAUXPOFTVCQBPWQUKZAHZHBLFTBLCZDKFAGOBCFBALFAK
CBKFTIDABUQQUAKPOKHUYSHUYACBKUQUVLRGZWCTRRVBLCZDKFAOUPOFTZDFBZSHSCLBKCKSHILPOCZFBRLFGUPB
UQCBLFHMLPZMEUGEMFAIDPWPCGRIYHDEMMHPWPCGRIYHDEMMPBUSHVCIVDVDSGRBLHPMLUGHPBLAEKLBKMBQCL
MMLUGLRFTOZKBCHEMEHCQBKSDONKZFBBLWCKTNAKLBKMBQCLMDGZHHERUMWKBBLHSGDHMZDKBABSHZHZAUG
VCNKOHZCIVHMUQXYZDLRKHPZMBGDMLDGZAFTOUAHMLUGUQBFHPMPECPKEMZSCZSHBQSHSHKTLULEALIVDLKBTH
PWWONRZSCHUMSHBIACNUXDBLEAKTCQBPSHMEXOMLGRGVBLLADGRVAMHZUQXYZDLRCZEMFLKLTFDGZATRUBWUZ
SHZKLIGXDIVMALGRFQCMPSABLGRLNLEBMLEIVIMRUFTFBLUQWWCFTHUVNSHSHIMQCXDUADETKANZDDKSZSHRBHC
BPUMWOZSASCLACQPHPQUWCOEPOTUSHBQSHIGQHRDZHILKBZHGAFTLUPWPCBLHZLESKQCFTMUSHUPDIUWVLBQPMS
HZCBUMWKBBLHZAZFBOPRBALGRMBMUSHZHZAUGVCNKPEUQKFCHLABLCHPWPUFTODKPCHDQWOBLGRHQUMZHIVDA
SHKMEDACMILRAMHZUQXYZDLRAHPWPCCHWCALUGMEGIDQCVFVDKFTUQAMABFTWQBFDWSAGRHPROQPZHWQCDFT
DMPCMEGIBLCSDIUWVLTFUGORIVHALUSHQPHPQUWCOEPOUSRUFTILAHIGURXPRMHAEMCHBLFHPKHMWUAHMALEBMS
HHABLKBZHGAEHEHDNGDCHFTOWSKBFLUSHOWPUZDFBBLFZLELOCLACTENIKLBUSHMDOUPOFTMEGIUQXDXDIVDAPO
NZWCXKPQKBGVCQUPUFKSPWPUDGRVAMHZUQXYZDLRHSCLUQOBMQDKDPTFUGHARVBLCSDIUWVLUGMCPZZSKSHAE
MCHTRBLKTIDZGCLZSKSRUBLAKPORHBGKBLEPKBLFTBLKSPDMPRGRGRVUMPCAEEAAUFTLUSHHGTIEAIVDIWUBQFBZH
HAPZQZWCMKTKDQUQENACRUZQKLLUSHXDKFCHUWVLFNWCKTCHQUAEPUUQVCUGZSCZFBALVCZNRUCQNURBLAAHX
GUVEMLMOPMBAZFBGRDYSHBUSHZHZAUGVCNKOELVHSPOCZSHBUSHZQCVQBWUZDLRAHXGFQEMGKHCCLROQUBLFH
UPRBLUSHVFUMUFHZBOSKAUCHBLHZBAQOMUSHTKUGBLFZDMPCCHFTGCWCUGBLKSPDMPBFODPOEZKCUKAHZGCLBL
FZALFARGUMRUZDFBQBWUZDLRAKLVABSDKBZHGAFTLUMDLUSHOVLEBLCLLEXRSKBFLUSHBUSHZXKTBFILKBQUKAOP
BKUMEHROQUSHNITCUIFTLUPWPQFBBLCHPWPUUMUQQUABSHONKZFBLADSSAZCBUGRKNQUUMANSHHDEMMEUGMEG
IDMPCGRIYHDEMMHPWPCGRIYHDEMMPBUSHVCIVDVDSGRBLHPMLUGHPBLAEKLBKMBQCLMMASHRGALBQZCBKLCZEA
EPUCSGSBCFBBLWCKTNAKLBKMBQCLMDGZHHERUMWKBBLHSGDHMZDKBABSHZHZAUGVCNKOHZCIVHMUQXYZDLRK
HPZMBGDMLDGZAFTOUAHMLUGUQBFHPMPECPKEMZSCZSHBQSHSHKTLUSHHDEMPZMQACARDBPZZSABBKSTHALCUGZ
SPWPCXDBLTKHFQBWUZDLRHZLEIVDIHKZSAKKPCHAZFBGKLRDWMBMOASVGGKABSHBKROQUZSCZFBRLBUMCONLRH
ZAKBKAMQUAHXNPWPOZDKBFBPWQUKZAZKBKCSAFTLUSHBKZSASCLACZANXQPHPQUWCMEXRMICSRUPHHCLRDMPC
CHXKABNVZDLRCZLRLEBKODSHAFMPBLLAAZKBWQBFZDPMCVLRDMFQCHCQKPUTRUZDSCZATKBAKMHKKAQCLROTM
BOBBKGVIGCZSHBUSHKMMICZEMBKCLZQSHMKGRBAILKBECKAFTIWIUHZZHILKBZHGAROEMCKWCBUNXUWVLTFUGMP
ZSCZFBUMZANUSHZHZATQWCBUKLBQUMBLHZAZFBOUKTEQZSAFUQKFCHLABLCHPWPUQBWUZDLRKSRUSHBIACOWRU
QCBLCZGROWVCUGMEGIDKFTUQAMABFTWQBFGRBLHZAZFBOUKTEQLEZACQNTGIHCTQKHRLMEIVMASHMESAUMSAFT
OZKBCHBLCZPOFTHPLEIVHMPWQUHFUNGIQCQVKMMIHZBLKBKHXDIVDHLXFZWCQUFBPMKLHOUQXYZDLRKHLCBKUQ
RVALFAFTVNLGFQKTFZWCCETKGIQCQPHPQUMEGIBLCSDIUWVLTGPWPCDXGDDARUBLACXRMIKHBUUWXPAKHZWQBCL
ABLKBCHEDKZAFFTIGXDIVHALABLCZLGBABLACXRMICZEMMPDCRYRGFQGVVLHCBLLAQUGVBLKBCHEDKZAFVCMQPD
MPBFHOUQXYZDLRAHQUHFGCPUBLAEGRACXRMIACUGZDFBZSHZCVBLLABLCZPOFTLAKSKTUMKGBMSHDWTDIUHZHP

UBMDQUHFTKUGMBBKUASHOWRUSHBIFTLCUGUMPCBLKSPDMPUQBOLEGATRBLKTIDZGCLZSACRHUMKGRLBGKBOPRU
LBLEBQSHEAIVDIWUUGACLUSHBUSHONKZFBDMPCCHLASHUXLRLEIUACGCTDDMPCFTIDCTHLCHLMWCLCUGMEGIZDLR
ACPZQBWUBKLCZEDMWOTHCKFCZHUSFNUWWOBLHPRLSABLCHVCNKOEFTUQUVCFFZWCROFLKLTFUGZDFBZSACPECH
QCBLAHPWPCGRIYHDEMMEBLPOAHPWPCGRIYHDEMMHPWPCGRIYHDEMMPBUSHVCIVDVDSGRBLHPMLUGHPBLAEKLBK
MBQCLMDGALBQZCBKLCAMEMUMGSBCFBGRBLHPMLUGBLWCKTNAKLBKMBQCLMMASHBGKBFBXLWCZSACNULEIUHZ
BMQCKTEMFTFBXLWCZSACNUKCBGLTKBHZRVBLCHVCNKOZSHBUSHWFLVHSPOKHBUSHAZFBGRQGWCUGQBWUZDLRA
HXGFQROQUBLFHUPRBHOSHTNACBLCHUQDMFQKBFTOUEMHCCLBLFZDGANCHSHNRMIPWPCCHUMZHCZSHBLKBBLFZD
WTDGCGRLCFNBLGRIQPDMPUQBORBMLKAILKBDWSTCHSHBIKLUXEMSHILKBXDBLVCBATKUGXDBFTRRLUVEMLMDMGF
WCZHBLKSPWPUEDGDOUEMHCCLFTIUASKLBFIWHCCLBLFZMLGISHNRQPTFRHKBFTLCIVHMAZFBQCBLASMEUNWCUGLR
FTOUKSNAPUSHTKUGZCMRBLBKCHFVMHQULNSHBUSHZXKBFHPDFQUMTRNXSHNTRUZSHZRLIVIVDMPCCHEMGKZSCZK
BAHLXGRLFUGMBGDDLCLSHKXBKRVNVBKODSHAFMBSHZQKLOUAKWCDIUQUTRNZGKBGRBLCHKTFIUPRBLCUGBLFZLE
LCUNFTMQKTFBUAQCKGUQGQMLPZMOCLHZMULEBMSHDWTDHUHZKZFBKNPOLMHMGIHZFTLUSHVGKLILKBGRLRFTOU
UPBASHNBLVCBQPRGRLFQMDHCTGABBABPFTLULEBMSHVCMEIPFQOHCTHLBANUSHBUSHXQRUBLAKQCKGUQRUROQU
BLASWCBUKLTFUGQUCSANSHVLDVFVPZMBFVCHNZLRFTOULAEDCHLRFTHUPKHOKCSAUMMCUGZCMRCQBPUQXYZDLR
AEPKMBFTWUMASHTFTKZATFUGMLMBCLMWACUGUQZOUQXYZDLRAHFVLMQUHDQULNLPKHBUSHDWKYKGTFUGZSCH
GIUQUALEFQHERUDMPCCHEMGKZSCZKBACMHKZKSCQBPSHVCMEOZLGBAEMQBWUEMBKLCZEDMWOFTILKBLEPKVCM
EZEMLKFEMHEQUXGBLKMSKBFILWUAZKBBLLACFGRHQMPUFLVKHBUMUSHCQBKMBFTOUFBRVUMBLCHVCNKZSWCZH
UMMQFBLUSHRGHLBLHZCVBLAKQCKGUQBUSHOPRBALGRFTLUSHEHCTRLCNHMUQXYZDLRCZLRLGRGALUGBKCLZDLA
VCOZKPRVBABRQPRGRLFQMDHCTGABBABPBLFBSHMEGIUMBLCHVCNKMHABSALEZABLAKQCKGRGXDIVDHLXFZWCZD
LAAZKBWQCUFTNAPULRTRRVBLCHVCNKMEBQZCBKOUCZGDGTMUMKNUXDBLLRCQTKIVMASHDQDIGRBLCZPUDIBLFBS
HMEGIHAAVHSLTKHALMEHGANCQKPKCRYUQAMLXCZLRDSRUZHKHFQQBWUZDLRKHXDIVDLQUWCTKUGXKHXHSKBEM
QVHCKTHZRMPMZHPZPHQUEMSHMQPCKHBULGOZFVBLFHPWWOOUKTEQUMPCXPNVROQULGVLUGLFZALGVASHDWKY
KGTFUGUQVUSHIPFQPZMPBGCLLRKCKRKPCHCVFVNZLGPZZHTXFTBMQUBLKHBPDSTDHCBPFTIGBLKMSKBFLULERVBLF
ZMBTOTHCHMWKBLABKHZUROQFTOULRLBSHZSWDARSHQPHPQUWCMQHKKAQCKMMICZLGEDQBWUZDLRHSCLRHRVB
AUQKYEKLVCSRUQCSHNUIBMCALUGEHCQBKDMFQZSCSSKBFILKBFZHZDGANCHROBLAHGIQCBLAKQCKGBQLAROFTOU
EMQUKAOPBKLMLXCSIZSHLTFBOZSHBKLVHPSHKUHKLUSHIPFQPZMPFGTDDMKGQWMEGIWQTCBKXDBLSAFTLUMDILKB
ZDLALEILKMSKBFLCUGMEGIGDMWFZWCOPBKBLFTAZKBVNLGFQGRBLCZPUDIDWSAXDBLSAUMQVECBLKBMQLMCKFZ
WCZSLMHTASZXGKACUGQBWUZDLRCZEMXDIVDGFQFTOUFBUVLRIZLGVLUGBLKHUXCHIGFQVCZSAHOULRLHBKLAZSW
UFTHOMIFBOWRNTRBMUQXYZDLRHZXDKFCHMBKSOPBLKBVCMEHMSUGIHCBUMUSHEAEMUVSHBQSHLELCKRTKZFILK
BZHGAGREHCTRLCNDADMBLHZZSCZFBUTAEPUABSHDMPCGRIYHDEMMEUGMEGIDMPCGRIYHDEMMHPWPCGRIYHDEMM
PBUSHVCIVDVDSGRBLHPMLUGHPBLAEKLBKMBQCLMMASHHDEMMEBQZCBKOWGSBCFBQCLMDLKBABSHECPKHZALNU
BLWCTRRVBLFZWCFQONKZFBHPECPKKBNZECTKPHUNWUBLFBBLCZDKFAOUPOFTRUBABKLCDCBPACUGVCPZMPXUABE
DCHMPRUBATIZXBKUASHLULEBMSHFRCZUQUVLERVUMOWLAKGBPBMSHUWTDGUMUDQUMBLCZCHIGFQLALMHTRUZS
HSCLUQSKCKACUGOTMBDQUMZHABSHWQRBQYBCFBFTOUSHBQSHZCBUGRZSAFBFZOUQXYZDLRACUGZSHZUMWOMBG
DDVLRIBFLACUGEAKTFTODWCDIUQBMRLNTRUPKQUMHPXSKBPLELCITACGWEHFBORURQPBUSHKGBKFTLUSHZXKBHE
UPNDLBGRXDBLUMFQMEUGZHUTAEPUHSKBBLFBZSCZEMCQUKCHUMORUMBUSHBKILMUZSSCMEUGLCUFCBBUGDZSCH
UPXRSKIWOWYUHCCLNZINLPPLCKSUFTOUGDSHMV

Steps:

Langkah pertama, kita dapat mencari frekuensi kemunculan huruf pada *cipher text* tersebut. Dengan menggunakan baris kode berikut,

```
from collections import Counter

def text_counter(text: str, n: int) -> dict:
  return Counter(text[i : i + n] for i in range(len(text) - (n-1)))

CIPHER_TEXT_3 = ""

# Bigram paling sering muncul
print(f"\nBigram paling sering muncul")
print(text_counter(CIPHER_TEXT_3, 2))
```

diperoleh frekuensi kemunculan bigram sebagai berikut.

'BL': 296, 'SH': 268, 'UG': 142, 'FT': 142, 'KB': 125, 'FB': 122, 'HZ': 117, 'CH': 117, 'GR': 116, 'LA': 111, 'EM': 105, 'LR': 102, 'ZD': 101, 'LU': 99, 'BK': 95, 'US': 94, 'LC': 94, 'UM': 94, 'ZS': 92, 'CZ': 92, 'QU': 88, 'MB': 85, 'LF': 84, 'ME': 83, 'UQ': 82, 'AC': 82, 'VC': 79, 'HP': 77, 'DL': 76, 'LK': 73, 'HB': 72, 'MP': 71, 'CL': 69, 'LE': 69, 'BU': 67, 'WC': 67, 'AL': 65, 'QB': 65, 'PC': 63, 'ZA': 63, 'BF': 63, 'CB': 62, 'LB': 62, 'AB': 62, 'RU': 61, 'KL': 60, 'CU': 57, 'QC': 56, 'ZH': 56, 'RV': 56, 'HC': 56, 'PW': 54, 'HL': 54, 'ML': 53, 'ZF': 52, 'DM': 52, 'FQ': 51, 'RB': 50, 'BA': 50, 'SA': 50, 'MH': 50, 'IV': 49, 'BM': 49, 'KT': 49, 'BQ': 49, 'LG': 48, 'HD': 48, 'WU': 47, 'LH': 47, 'DI': 47, 'PU': 47, 'BB': 46, 'HM': 46, 'DG': 45, 'AH': 45, 'LM': 45, 'HA': 44, 'GD': 44, 'MM': 44, 'SC': 43, 'KZ': 43, 'HS': 42, 'HU': 41, 'PB': 41, 'DS': 41, 'AZ': 41, 'OU': 41, 'KH': 40, 'AS': 40, 'XD': 40, 'WP': 40, 'RL': 40, 'BP': 39, 'HE': 39, 'VB': 39, 'AM': 39, 'EH': 39, 'KM': 39, 'CQ': 38, 'BW': 38, 'TL': 38, 'AK': 38, 'BG': 38, 'TH': 37, 'RA': 37, 'GI': 37, 'MU': 37, 'VD': 36, 'CK': 36, 'HF': 36, 'UL': 35, 'CM': 35, 'HO': 35, 'DE': 35, 'CN': 35, 'FZ': 35, 'UZ': 34, 'PZ': 34, 'PO': 33, 'HK': 33, 'ZE': 33, 'UP': 33, 'UB': 33, 'SK': 33, 'UK': 32, 'KA': 32, 'AE': 32, 'MS': 32, 'TK': 32, 'VL': 32, 'ZC': 31, 'LL': 31, 'KS': 31, 'TR': 31, 'GA': 31, 'BS': 31, 'TO': 31, 'TF': 31, 'BC': 30, 'GZ': 30, 'CG': 30, 'AF': 30, 'GB': 30, 'MD': 30, 'CS': 29, 'NK': 29, 'GK': 29, 'RI': 29, 'PK': 28, 'IL': 28, 'UA': 28, 'HV': 28, 'KC': 28, 'WQ': 28, 'CF': 27, 'KF': 27, 'UF': 27, 'RG': 27, 'ZK': 27, 'KP': 27, 'QX': 26, 'IU': 26, 'AU': 25, 'GF': 25, 'TI': 25, 'DK': 25, 'MQ': 25, 'CC': 25, 'EB': 25, 'MA': 25, 'PM': 25, 'RH': 25, 'GC': 24, 'EA': 24, 'HX': 24, 'DB': 24, 'UW': 24, 'EG': 24, 'GU': 24, 'DF': 23, 'BZ': 23, 'QP': 23, 'UH': 23, 'GH': 23, 'GV': 23, 'RO': 22, 'KG': 22, 'CP': 22, 'XY': 22, 'YZ': 22, 'IG': 22, 'IY': 22, 'YH': 22, 'ZM': 22, 'EC': 21, 'MZ': 21, 'WO': 21, 'ZL': 21, 'ED': 21, 'MW': 21, 'FV': 21, 'FA': 20, 'AN': 20, 'FU': 20, 'HH': 20, 'NU': 20, 'KU': 20, 'OP': 20, 'LT': 20, 'ID': 19, 'FH': 19, 'CT': 19, 'VA': 19, 'DW': 19, 'CI': 18, 'LV': 18, 'DQ': 18, 'OW': 18, 'QK': 18, 'CD': 18, 'HI': 18, 'TD': 18, 'EK': 18, 'VH': 18, 'BO': 18, 'HG': 18, 'PR': 18, 'UV': 17, 'QR': 17, 'CR': 17, 'OF': 17, 'EP': 17, 'MK': 17, 'GM': 17, 'PH': 17, 'QV': 17, 'FL': 17, 'HT': 17, 'ZU': 16, 'QZ': 16, 'HQ': 16, 'EU': 16, 'QS': 16, 'RC': 16, 'DH': 16, 'ZQ': 15, 'VF': 15, 'ZG': 15, 'XK': 15, 'VU': 15, 'OZ': 15, 'AT': 15, 'CV': 15, 'UD': 15, 'ER': 15, 'SB': 15, 'SR': 15, 'KO': 15, 'UX': 14, 'OB': 14, 'AQ': 14, 'LW': 14, 'EL': 14, 'EZ': 14, 'DD': 14, 'MI': 14, 'ZZ': 14, 'NA': 14, 'TG': 14, 'SD': 14, 'DC': 13, 'MO': 13, 'RD': 13, 'WV': 13, 'OE': 13, 'UE': 13, 'ON': 13, 'EI': 13, 'PF': 13, 'TN': 13, 'DV': 13, 'SG': 13, 'XG': 13, 'ZW': 13, 'AR': 12, 'OQ': 12, 'KN': 12, 'NL': 12, 'IB': 12, 'PQ': 12, 'UC': 12, 'MC': 12, 'BI': 12, 'BT': 12, 'UU': 12, 'TE': 12, 'FG': 12, 'PS': 12, 'HN': 12, 'HR': 12, 'AI': 11, 'WW': 11, 'PA': 11, 'ND': 11, 'RK': 11, 'NR': 11, 'BE': 11, 'LN': 11, 'UR': 11, 'GT': 11, 'FR': 11, 'SP': 11, 'PD': 11, 'OD': 11, 'IZ': 11, 'DA': 11, 'IW': 10, 'QF': 10, 'OC': 10, 'EQ': 10, 'OS': 10, 'QM': 10, 'GL': 10, 'TA': 10, 'CA': 10, 'GQ': 10, 'BD': 10, 'TC': 10, 'RN': 10, 'NZ': 10, 'CX': 10, 'TM': 10, 'HW': 10, 'FC': 10, 'AA': 10, 'BR': 10, 'WK': 10, 'RM': 10, 'VN': 10, 'LP': 9, 'LI': 9, 'QD': 9, 'GW': 9, 'NS': 9, 'DP': 9, 'NV': 9, 'ZP': 9, 'IP': 9, 'OM': 9, 'UT': 9, 'SW': 9, 'MF': 9, 'NB': 9, 'FI': 9, 'MV': 9, 'AD': 9, 'GE': 9, 'RF': 9, 'PE': 9, 'MR': 9, 'MG': 9, 'BH': 9, 'WR': 9, 'DZ': 9, 'AO': 9, 'IC': 9, 'AG': 8, 'CE': 8, 'KW': 8, 'UN': 8, 'NC': 8, 'TB': 8, 'FS': 8, 'OK': 8, 'AP': 8, 'OA': 8, 'RR': 8, 'QN': 8, 'NX': 8, 'TU': 8, 'QQ': 8, 'SZ': 8, 'ZB': 8, 'QW': 8, 'ZX': 8, 'VM': 8, 'PL': 7, 'LX': 7, 'BX': 7, 'UI': 7, 'KI': 7, 'MT': 7, 'BV': 7, 'QE': 7, 'KK': 7, 'SL': 7, 'IA': 7, 'RQ': 7, 'VG': 7, 'OH': 7, 'NG': 7, 'OR': 7, 'CO': 7, 'KR': 7, 'IH': 7, 'SU': 7, 'LZ': 7, 'XR': 7, 'KQ': 7, 'KE': 6, 'NP': 6, 'XF': 6, 'FM': 6, 'ZO': 6, 'KD': 6, 'VR': 6, 'IO': 6, 'GO': 6, 'FN': 6, 'LQ': 6, 'IN': 6, 'GP': 6, 'XP': 6, 'SN': 6, 'RZ': 6, 'LD': 6, 'GG': 6, 'LS': 6, 'VE': 6, 'RS': 6, 'GS': 6, 'WF': 6, 'WS': 6, 'QO': 6, 'DU': 5, 'NM': 5, 'TQ': 5, 'BN': 5, 'IE': 5, 'KV': 5, 'FK': 5, 'ET': 5, 'CW': 5, 'NW': 5, 'LO': 5, 'VP': 5, 'DX': 5, 'MX': 5, 'OL': 5, 'WT': 5, 'UY': 5, 'PN': 5, 'VK': 5, 'EV': 5, 'QG': 5, 'QT': 5, 'IQ': 5, 'GX': 5, 'OT': 5, 'EN': 4, 'NQ': 4, 'KX': 4, 'VZ': 4, 'VQ': 4, 'ST': 4, 'AV': 4, 'WZ': 4, 'PI': 4, 'SF': 4, 'RW': 4, 'WM': 4, 'DO': 4, 'XU': 4, 'EF': 4, 'FF': 4, 'IM': 4, 'SO': 4, 'IK': 4, 'IF': 4, 'NH': 4, 'XL': 4, 'NF': 4, 'TW': 4, 'QH': 4, 'QA': 4, 'QL': 4, 'NT': 4, 'WI': 3, 'XS': 3, 'FE': 3, 'SV': 3, 'XE': 3, 'OG': 3, 'WD': 3, 'ZN': 3, 'IT': 3, 'WA': 3, 'RY': 3, 'RT': 3, 'IS': 3, 'XB': 3, 'YL': 3, 'SM': 3, 'YU': 3, 'XQ': 3, 'FD': 3, 'AW': 3, 'EX': 3, 'VI': 3, 'SS': 3, 'ZR': 3, 'KY': 3, 'XC': 3, 'PX': 3, 'CY': 2, 'SE': 2, 'DT': 2, 'WY': 2, 'EO': 2, 'RE': 2, 'DN': 2, 'YB': 2, 'YR': 2, 'PP': 2, 'UO': 2, 'FW': 2, 'WL': 2, 'SI': 2, 'ZI': 2, 'VS': 2, 'OO': 2, 'WG': 2, 'FP': 2, 'TV': 2, 'YS': 2, 'ES': 2, 'RX': 2, 'NI': 2, 'PT': 2, 'YK': 2, 'YG': 1, 'FX': 1, 'YW': 1, 'YC': 1, 'NY': 1, 'IR': 1, 'EW': 1, 'WB': 1, 'MN': 1, 'XM': 1, 'DR': 1, 'SY': 1, 'VT': 1, 'SQ': 1, 'WN': 1, 'VX': 1, 'RP': 1, 'PV': 1, 'ZT': 1, 'YA': 1, 'TZ': 1, 'XO': 1, 'EE': 1, 'DY': 1, 'FO': 1, 'OV': 1, 'XN': 1, 'WX': 1, 'VV': 1, 'WH': 1, 'XH': 1, 'TX': 1, 'YE': 1, 'AX': 1, 'QY': 1, 'WE': 1

Langkah kedua, berdasarkan **Gambar 1.** Top 10 huruf dalam teks Bahasa Inggris, BL berkorespondensi dengan TH dan SH berkorespondensi dengan HE. Selanjutnya saya tidak memiliki *clue* sama sekali 😕

Hasil Dekripsi

-

4) Kriptanalisis *Hill Cipher* dengan *known-plaintext attack*

Kriptogram

TFJOXUPOUXYTTRDSXQMONIYPEUFJDQUBGIMOCJQTNBEHCZEKROVBNTWLMVXMOWZLUCHOXYGSKBQGUAOBQZKIXYJ
IETSWVXHVKCUAOTOFYIZAKJGXKAWGQTRVFDZAJNQDUIWZCMYWNFIUPYMCZXIAKYUCQIAZPIQMGAMGUAKKKHMWK
DUXQDUAAKYOWEHLJPWYFKXSARBLLHGAJKTQNTRTPWSCIZASCGSLKVDHTUZSWBNBTJGYYUPQMFSYZAUTOQCDNGQ
MFSRLRTUWEMKADIVYLTJKFHLKJUWTSSHMHJFGTRIBYIDAHQEPMPIQCROWDYRYZNSPNOJHQVKKTOCBPNFAJNLYJZNVB
AYJWRGMCHJPWBDHHTPOXSIJVQWDMSIGMTRVEVXDILKVAYTNUNJXEZLAPGYETRVZNVHSVWLGICDXQFOALDVPASUSY
XPFHUWTILUQHTJQVGWFSPAEKBRBNIINYKHNTNUKJVDHVLXQKUZNVQXUOZZOJZYNPIVYSVFVTZMMUUPWTGHRIOWCB
KZYAGUMRCKHIQZSIGISPGBXPYXMOAWGAGHQVUWTEIGPBMOMBWIOPQEVKMRQATNBMILHHLVUXGMOUWTZCLBKGWI
JHFRNGOSCMUHDWHBB

Steps:

Diketahui bahwa *plain text* dimulai dengan “HELLOCAPTAINHADDOCK” dan *cipher text* TFJOXUPOUXYTTRDSXQM. Dengan *known plain text* berikut, dapat diambil 9 huruf pertama sesuai jumlah *key*, yaitu *matrix* 3 x 3.

- *Plain text*
  - HEL → (7, 4, 11)
  - LOC → (11, 14, 2)
  - APT → (0, 15, 19)
- *Cipher text*
  - TFJ → (19, 5, 9)
  - OXU → (14, 23, 20)
  - POU → (15, 14, 20)

Untuk menemukan kunci dari Hill Cipher ini, dapat menggunakan persamaan berikut.

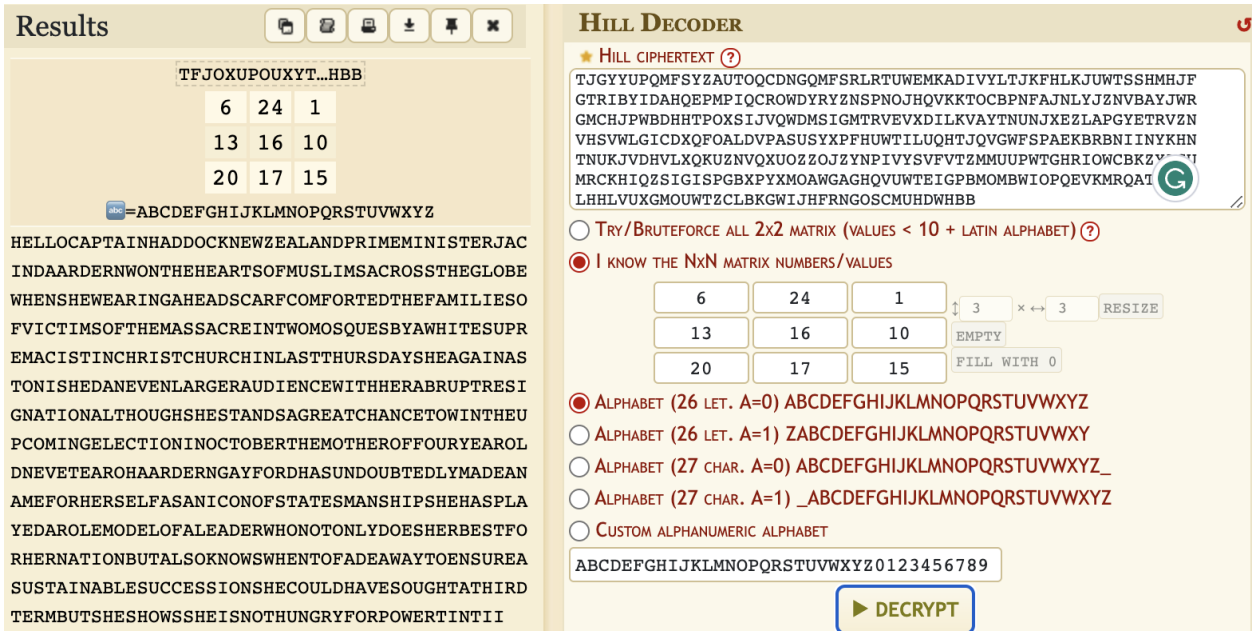
$$\mathbf{K = CP^{-1} \bmod 26}$$

Dari hasil perhitungan berikut, diperoleh:
K = ((656, 414, 443), (637, 432, 374), (644, 485, 483)) mod 26
K = ((6, 24, 1), (13, 16, 10), (20, 17, 15))

Dengan bantuan link https://www.dcode.fr/hillcipher, hasil dekripsi dari *cipher text* tersebut dengan key yang telah kita temukan adalah sebagai berikut.

IF4020 Kriptografi
Semester 2 - 2022/2023
Dosen Pengampu : Dr. Ir. Rinaldi Munir, M.T.

Tugas Kecil 1
Nama : Bintang Fajarianto
NIM : 13519138



Hasil Dekripsi

HELLOCAPTAINHADDOCKNEWZEALANDPRIMEMINISTERJACINDAARDERNWONTHEHEARTSOFMUSLIMSACROSSTHEGLOB
EWHENSHEWEARINGAHEADSCARFCOMFORTEDTHEFAMILIESOFVICTIMSOFTHEMASSACREINTWOMOSQUESBYAWHITESU
PREMACISTINCHRISTCHURCHINLASTTHURSDAYSHEAGAINASTONISHEDANEVENLARGERAUDIENCEWITHHERABRUPTRES
IGNATIONALTHOUGHSHESTANDSAGREATCHANCETOWINTHEUPCOMINGELECTIONINOCTOBERTHEMOTHEROFFOURYEAR
OLDNEVETEAROHAARDERNGAYFORDHASUNDOUBTEDLYMADEANAMEFORHERSELFASANICONOFSTATESMANSHIPSHEHA
SPLAYEDAROLEMODELOFALEADERWHONOTONLYDOESHERBESTFORHERNATIONBUTALSOKNOWSWHENTOFADEAWAYTO
ENSUREASUSTAINABLESUCCESSIONSHECOULDHAVESOUGHTATHIRDTERMBUTSHESHOWSSHEISNOTHUNGRYFORPOWERT
INTII

LAMPIRAN

Repository

https://github.com/bintangfrnz/Old-Cryptography

Spesifikasi - Bagian A

No	Spek	Berhasil (✔️)	Kurang Berhasil (✔️)	Keterangan
1	Vigenere Standard	✔️		
2	Auto-Key Vigenere Cipher	✔️		
3	Extended Vigenere Cipher	✔️		
4	Affine Cipher	✔️		
5	Playfair Cipher	✔️		
6	Hill Cipher	✔️		
7	(Bonus) Enigma Cipher	✔️		

Spesifikasi - Bagian B

No	Kriptanalisis	Berhasil (✔️)	Kurang Berhasil (✔️)	Keterangan
1	Kriptanalisis Cipher Abjad-Tunggal	✔️		
2	Metode Kasiski	✔️		
3	Kriptanalisis Playfair Cipher		✔️	
4	Kriptanalisis Hill Cipher	✔️		