Κρυπτογραφία

9ο Εξάμηνο 2023 – 2024

Assignment 1 – Solutions

Ζαρίφης Στέλιος – el20435
Email: el20435@mail.ntua.gr

Contents

Άσκηση 1

Εξηγούμε τις συναρτήσεις που χρησιμοποιήσαμε:

1. `load_ngrams`: Η συνάρτηση `load_ngrams` φορτώνει τις πιθανότητες των `ngram` από αρχείο. Τα `ngrams` είναι ακολουθίες `n` χαρακτήρων. Υπολογίζει τις πιθανότητες καταγραφής με βάση τη συχνότητα κάθε `ngram`. Τα δεδομένα που προκύπτουν περιλαμβάνουν ένα λεξικό ngrams, το μήκος των `ngrams` (L) και μια κατώτατη τιμή, η οποία είναι ένα κατώφλι για τη βαθμολογία των `ngrams` που δεν βρίσκονται στο λεξικό.

2. `score_text`: Η συνάρτηση `score_text` αποδίδει μια βαθμολογία στο κείμενο με βάση στατιστικά στοιχεία `ngram`. Λαμβάνει ως είσοδο το κείμενο προς βαθμολόγηση, ένα λεξικό `ngrams`, το μήκος των `ngrams` (L) και την κατώτατη τιμή. Η συνάρτηση επεξεργάζεται το κείμενο διαιρώντας το σε `ngrams` και υπολογίζει τη βαθμολογία λαμβάνοντας υπόψη τις λογαριθμικές πιθανότητες αυτών των `ngrams`. Εάν ένα ngram δεν βρεθεί στο λεξικό, χρησιμοποιείται η κατώτατη τιμή (για να αποφευχθεί το $\log(0) = NaN$). Το αποτέλεσμα είναι η συνολική βαθμολογία του κείμενο εισόδου.

$$\text{Score}(\text{text}, \text{ngrams}, L, \text{floor}) = \sum_{i=1}^{(\text{length}(\text{text})-L+1)} \left(\log_{10}\left(\text{ngrams}[\text{text}[i : i+L]] + \text{floor}\right)\right)$$

3. `remove_periods_and_spaces`: Αυτή η συνάρτηση εντοπίζει τις θέσεις των τελειών, κομμάτων κενών μέσα στο κείμενο εισόδου και τις αφαιρεί. Κατά την επεξεργασία του κειμένου, καταγράφει τις θέσεις αυτές σε ξεχωριστά λεξικά. Οπότε παράγει ένα κείμενο μόνο με κεφαλαίους, αγγλικούς χαρακτήρες. Η συνάρτηση επιστρέφει το τροποποιημένο κείμενο μαζί με τα λεξικά που περιέχουν πληροφορίες για τις θέσεις των σημείων στίξης.

4. `insert_spaces_and_periods`: Η συνάρτηση επαναφέρει τα κενά και τις περιόδους στο κείμενο στις αρχικές τους θέσεις. Αντιστρέφει δηλαδή τη δράση της προηγούμενης συνάρτησης, όταν πλέον το κείμενο έχει αποκρυπτογραφηθεί.

5. `split_text_to_tokens`: Αυτή η συνάρτηση διαχωρίζει το κείμενο εισόδου αναγνωρίζοντας λέξεις, περιόδους, κόμματα και κενά. Χρησιμοποιείται στην εξαγωγή tokens από το κείμενο.

6. `reconstruct_from_tokens`: Αυτή η συνάρτηση ανακατασκευάζει το κείμενο από την έξοδο της `split_text_to_tokens`.

7. Ο κύριος κώδικας αρχικοποιεί τυχαία το πρώτο κλειδι. Ο αλγόριθμος ανακατεύει επαναληπτικά το κλειδί και αξιολογεί τη βαθμολογία του αποκρυπτογραφημένου κειμένου χρησιμοποιώντας την `score_text`. Στη συνέχεια, αποδέχεται τροποποιήσεις στο κλειδί που βελτιώνουν το score. Το καλύτερο κλειδί και η βαθμολογία ενημερώνονται. Η διαδικασία συνεχίζεται έως ότου δεν υπάρξει βελτίωση της καλύτερης βαθμολογίας για κάποιες επαναλήψεις, οπότε και θεωρούμε ότι επιτύχαμε βέλτιστη λύση.

```python
# Import necessary libraries
from pycipher import SimpleSubstitution as SimpleSub
import random
import re
import string
from math import log10

# Function to load ngrams and calculate log probabilities
def load_ngrams(ngramfile, sep=' '):
    ngrams = {}

    # Read ngrams and their counts from a file
    for line in open(ngramfile):
        key, count = line.split(sep)
        ngrams[key] = int(count)

    L = len(key)
    N = sum(ngrams.values())

    # Calculate log probabilities
    for key in ngrams.keys():
        ngrams[key] = log10(float(ngrams[key]) / N)

    floor = log10(0.01 / N)
```

```
    return ngrams, L, floor

# Function to compute the score of text
def score_text(text, ngrams, L, floor):
    score = 0
    ngrams_get = ngrams.__getitem__

    # Calculate the score based on ngrams
    for i in range(len(text) - L + 1):
        ngram = text[i:i+L]
        if ngram in ngrams:
            score += ngrams_get(ngram)
        else:
            score += floor

    return score

# Function to remove periods and spaces from text
def remove_periods_and_spaces(text):
    period_positions = {}
    text_without_periods = []

    for i, char in enumerate(text):
        if char == '.':
            period_positions[i] = '.'
        else:
            text_without_periods.append(char)

    space_positions = {}
    text_without_spaces = []

    for i, char in enumerate(text_without_periods):
        if char == ' ':
            space_positions[i] = ' '
        else:
            text_without_spaces.append(char)

    return ''.join(text_without_spaces), space_positions, period_positions

# Function to insert spaces and periods back into text
def insert_spaces_and_periods(text_without_spaces, space_positions, period_positions):
    text_characters = list(text_without_spaces)

    for pos, char in sorted(space_positions.items()):
        text_characters.insert(pos, char)

    text_with_spaces = ''.join(text_characters)
    text_characters = list(text_with_spaces)

    for pos, char in sorted(period_positions.items()):
        text_characters.insert(pos, char)

    text_with_spaces_and_periods = ''.join(text_characters)
    return text_with_spaces_and_periods

# Function to split text into tokens
def split_text_to_tokens(text):
    word_pattern = r'\b\w+\b'
```

```
    period_pattern = r'\.'
    comma_pattern = r','
    space_pattern = r' '

    # Define a combined regular expression pattern
    combined_pattern = f"({word_pattern})|({period_pattern})|({comma_pattern})|({
        space_pattern})"

    # Find all matches based on the combined pattern
    tokens = re.findall(combined_pattern, text)

    # Filter out empty strings
    tokens = [token for token in sum(tokens, ()) if token]

    raw_text = []

    # Remove punctuation and spaces
    for token in tokens:
        if token != ' ' and token != '.' and token != ',':
            raw_text.append(token)

    return ''.join(raw_text), tokens

# Function to reconstruct text from tokens
def reconstruct_from_tokens(raw_text, tokens):
    reconstructed_list = []
    raw_text = list(raw_text)
    j = 0

    for i in range(len(tokens)):
        if tokens[i] in [' ', '.', ',']:
            reconstructed_list.append(tokens[i])
        else:
            reconstructed_list.append(raw_text[j:j+len(tokens[i])])
            j += len(tokens[i])

    reconstructed = ""

    # Concatenate the reconstructed list into a string
    for elem in reconstructed_list:
        if isinstance(elem, list):
            reconstructed += ''.join(elem)
        else:
            reconstructed += elem

    return reconstructed

# Load ngrams and parameters
ngrams, L, floor = load_ngrams('english_quadgrams.txt')

# Cipher text
cipher_text = "KVU HQBINKWALU DNBAURG BWO AU YUGHRCAUY ARCUPLO WG KVU RUWL DNBAURG ZVQGU
    UTIRUGGCQDG WG W YUHCBWL WRU HWLHNLWALU AO PCDCKU BUWDG. WLKVQNJV KVU GNAEUHK QP
   KVCG IWIUR CG QGKUDGCALO KVU HQBINKWALU DNBAURG. CK CG WLBQGK UFNWLLO UWGO KQ YUPCDU
    WDY CDXUGKCJWKU HQBINKWALU PNDHKCQDG QP WD CDKUJRWL XWRCWALU QR W RUWL QR
   HQBINKWALU XWRCWALU, HQBINKWALU IRUYCHWKUG, WDY GQ PQRKV. KVU PNDYWBUDKWL IRQALUBG
   CDXQLXUY WRU, VQZUXUR, KVU GWBU CD UWHV HWGU, WDY C VWXU HVQGUD KVU HQBINKWALU
   DNBAURG PQR UTILCHCK KRUWKBUDK WG CDXQLXCDJ KVU LUWGK HNBARQNG KUHVDCFNU. C VQIU
   GVQRKLO KQ JCXU WD WHHQNDK QP KVU RULWKCQDG QP KVU HQBINKWALU DNBAURG, PNDHKCQDG,
```

```
    WDY GQ PQRKV KQ QDU WDQKVUR. KVCG ZCLL CDHLNYU W YUXULQIBUDK QP KVU KVUQRO QP
    PNDHKCQDG QP W RUWL XWRCWALU UTIRUGGUY CD KURBG QP HQBINKWALU DNBAURG. WHHQRYCDJ KQ
    BO YUPCDCKCQD, W DNBAUR CG HQBINKWALU CP CKG YUHCBWL HWD AU ZRCKKUD YQZD AO W
    BWHVCDU. C JCXU GQBU WRJNBUDKG ZCKV KVU CDKUDKCQD QP GVQZCDJ KVWK KVU HQBINKWALU
    DNBAURG CDHLNYU WLL DNBAURG ZVCHV HQNLY DWKNRWLLO AU RUJWRYUY WG HQBINKWALU. CD
    IWRKCHNLWR, C GVQZ KVWK HURKWCD LWRJU HLWGGUG QP DNBAURG WRU HQBINKWALU. KVUO
    CDHLNYU, PQR CDGKWDHU, KVU RUWL IWRKG QP WLL WLJUARWCH DNBAURG, KVU RUWL IWRKG QP
    KVU MURQG QP KVU AUGGUL PNDHKCQDG, KVU DNBAURG IC, U, UKH. KVU HQBINKWALU DNBAURG YQ
     DQK, VQZUXUR, CDHLNYU WLL YUPCDWALU DNBAURG, WDY WD UTWBILU CG JCXUD QP W YUPCDWALU
     DNBAUR ZVCHV CG DQK HQBINKWALU. WLKVQNJV KVU HLWGG QP HQBINKWALU DNBAURG CG GQ
    JRUWK, WDY CD BWDO ZWOG GCBCLWR KQ KVU HLWGG QP RUWL DNBAURG, CK CG DUXURKVULUGG
    UDNBURWALU. C UTWBCDU HURKWCD WRJNBUDKG ZVCHV ZQNLY GUUB KQ IRQXU KVU HQDKRWRO. AO
    KVU HQRRUHK WIILCHWKCQD QP QDU QP KVUGU WRJNBUDKG, HQDHLNGCQDG WRU RUWHVUY ZVCHV WRU
     GNIURPCHCWLLO GCBCLWR KQ KVQGU QP JQYUL. KVUGU RUGNLKG VWXU XWLNWALU WIILCHWKCQDG.
    CD IWRKCHNLWR, CK CG GVQZD KVWK KVU VCLAURKCWD UDKGHVUCYNDJGIRQALUB HWD VWXU DQ
    GQLNKCQD"

# Remove periods and spaces
raw_cipher_text, tokens = split_text_to_tokens(cipher_text)
ctext = re.sub('[^A-Z]', '', raw_cipher_text.upper())

# Initialize variables
letters = list(string.ascii_uppercase)
maxkey = list(''.join(random.sample(letters, len(letters))))
maxscore = -99e9
parentscore, parentkey = maxscore, maxkey[:]
i = 0
best_score_combo = 0
best_score = -99e9

while 1:
    i = i + 1
    random.shuffle(parentkey)
    deciphered = SimpleSub(parentkey).decipher(ctext)
    parentscore = score_text(deciphered, ngrams, L, floor)
    count = 0

    while count < 1000:
        a = random.randint(0, 25)
        b = random.randint(0, 25)
        child = parentkey[:]
        child[a], child[b] = child[b], child[a]
        deciphered = SimpleSub(child).decipher(ctext)
        score = score_text(deciphered, ngrams, L, floor)

        if score > parentscore:
            parentscore = score
            parentkey = child[:]
            count = 0
            count = count + 1

    if parentscore > maxscore:
        maxscore, maxkey = parentscore, parentkey[:]
        print('\nbest score so far:', maxscore, 'on iteration', i)
        ss = SimpleSub(maxkey)
        print('    best key: ' + ''.join(maxkey))
        # print('    plaintext: ' + ss.decipher(ctext))

    if best_score != maxscore:
```

```
        best_score = maxscore
        best_score_combo = 0
    else:
        best_score_combo += 1

    if best_score_combo > 1:
        break

raw_decrypted_text = ss.decipher(ctext)
decrypted_text = reconstruct_from_tokens(raw_decrypted_text, tokens)
print("=== DECRYPTED TEXT ===\n")
print(decrypted_text)
```

Και το αποτέλεσμα φαίνεται εδώ:

```
best score so far: -6242.47053444735 on iteration 1
best key: WAHYUPJVCESLBDQIFRGKNXZTOM

best score so far: -6240.388453280439 on iteration 3
best key: WAHYUPJVCEMLBDQIFRGKNXZTOS
=== DECRYPTED TEXT ===

THE COMPUTABLE NUMBERS MAY BE DESCRIBED BRIEFLY AS THE REAL NUMBERS WHOSE EXPRESSIONS AS
    A DECIMAL ARE CALCULABLE BY FINITE MEANS. ALTHOUGH THE SUBJECT OF THIS PAPER IS
   OSTENSIBLY THE COMPUTABLE NUMBERS. IT IS ALMOST EQUALLY EASY TO DEFINE AND
   INVESTIGATE COMPUTABLE FUNCTIONS OF AN INTEGRAL VARIABLE OR A REAL OR COMPUTABLE
   VARIABLE, COMPUTABLE PREDICATES, AND SO FORTH. THE FUNDAMENTAL PROBLEMS INVOLVED ARE
   , HOWEVER, THE SAME IN EACH CASE, AND I HAVE CHOSEN THE COMPUTABLE NUMBERS FOR
   EXPLICIT TREATMENT AS INVOLVING THE LEAST CUMBROUS TECHNIQUE. I HOPE SHORTLY TO GIVE
    AN ACCOUNT OF THE RELATIONS OF THE COMPUTABLE NUMBERS, FUNCTIONS, AND SO FORTH TO
   ONE ANOTHER. THIS WILL INCLUDE A DEVELOPMENT OF THE THEORY OF FUNCTIONS OF A REAL
   VARIABLE EXPRESSED IN TERMS OF COMPUTABLE NUMBERS. ACCORDING TO MY DEFINITION, A
   NUMBER IS COMPUTABLE IF ITS DECIMAL CAN BE WRITTEN DOWN BY A MACHINE. I GIVE SOME
   ARGUMENTS WITH THE INTENTION OF SHOWING THAT THE COMPUTABLE NUMBERS INCLUDE ALL
   NUMBERS WHICH COULD NATURALLY BE REGARDED AS COMPUTABLE. IN PARTICULAR, I SHOW THAT
   CERTAIN LARGE CLASSES OF NUMBERS ARE COMPUTABLE. THEY INCLUDE, FOR INSTANCE, THE
   REAL PARTS OF ALL ALGEBRAIC NUMBERS, THE REAL PARTS OF THE ZEROS OF THE BESSEL
   FUNCTIONS, THE NUMBERS PI, E, ETC. THE COMPUTABLE NUMBERS DO NOT, HOWEVER, INCLUDE
   ALL DEFINABLE NUMBERS, AND AN EXAMPLE IS GIVEN OF A DEFINABLE NUMBER WHICH IS NOT
   COMPUTABLE. ALTHOUGH THE CLASS OF COMPUTABLE NUMBERS IS SO GREAT, AND IN MANY WAYS
   SIMILAR TO THE CLASS OF REAL NUMBERS, IT IS NEVERTHELESS ENUMERABLE. I EXAMINE
   CERTAIN ARGUMENTS WHICH WOULD SEEM TO PROVE THE CONTRARY. BY THE CORRECT APPLICATION
    OF ONE OF THESE ARGUMENTS, CONCLUSIONS ARE REACHED WHICH ARE SUPERFICIALLY SIMILAR
   TO THOSE OF GODEL. THESE RESULTS HAVE VALUABLE APPLICATIONS. IN PARTICULAR, IT IS
   SHOWN THAT THE HILBERTIAN ENTSCHEIDUNGSPROBLEM CAN HAVE NO SOLUTION.
```

Πηγή: [1, Practical Cryptography]

Άσκηση 2

Έστω το affine cipher: $c = Enc((a,b),m) \equiv am+b \pmod{26}$ και έστω δύο γνωστά μηνύματα m_1, m_2 με γνωστά c_1, c_2 αντίστοιχα.

Ερώτημα 1

Για τα ζεύγη $(m_1, c_1), (m_2, c_2)$ έχουμε:

$$\begin{cases} c_1 \equiv am_1 + b \pmod{26} \\ c_2 \equiv am_2 + b \pmod{26} \end{cases}$$

Αφαιρώντας τις σχέσεις κατά μέλη παίρνουμε:

$$c_1 - c_2 \equiv a(m_1 - m_2) + b \pmod{26} \Rightarrow a \equiv (c_1 - c_2)(m_1 - m_2)^{-1} \pmod{26}$$

Όπου, $(m_1 - m_2)^{-1}$ ο αντίστροφος του $(m_1 - m_2) \pmod{26}$. Σε αυτό το σημείο, έχουμε τον περιορισμό ότι πρέπει να υπάρχει ο αντίστροφος και αυτό γίνεται αν και μόνον αν gcd$((m_1 - m_2), 26) = 1$

Δείξαμε, λοιπόν, ότι αν επιλέξουμε τα $m_1 - m_2$ ώστε $gcd((m_1 - m_2), 26) = 1$, μπορούμε να υπολογίσουμε το a χρησιμοποιώντας τον εκτεταμένο αλγόριθμο του Ευκλείδη $a \equiv (c_1 - c_2)(m_1 - m_2)^{-1} \pmod{26}$.

Με γνωστό το a, επιλέγουμε τη μία ισοδυναμία, έστω $c_1 \equiv am_1 + b \pmod{26}$ και υπολογίζουμε $b \equiv c_1 - am_1 \pmod{26}$

Άρα, "σπάσαμε" το κρυπτοσύστημα, αφού υπολογίσαμε το κλειδί $k = (a, b)$

Ερώτημα 2

Έστω τα κλειδιά $k_1 = (a_1, b_1)$ και $k_2 = (a_2, b_2)$. Τότε, για τη διπλή κρυπτογράφηση έχουμε:

$$Enc(k, m) = Enc(k_2, Enc(k_1, m)) = a_2(a_1 m + b_1) + b_2 \equiv a_1 a_2 m + b_1 a_2 + b_2 \pmod{26}$$

Αν θέσουμε $k = (a, b) = (a_1 a_2, b_1 a_2 + b_2)$ εύκολα βλέπουμε ότι το πλήθος των κλειδιών είναι ίδιο με την αρχική περίπτωση: όλα τα ζεύγη $(a, b) \in \mathbb{Z}_{26} \times \mathbb{Z}_{26}$.

Δηλαδή ο χώρος κλειδιών δεν επηρεάζεται. Άλλωστε, ο αφινικός μετασχηματισμός ενός αφινικού μετασχηματισμού είναι αφινικός μετασχηματισμός, οπότε δεν αλλάζει ούτε ο χώρος των κλειδιών, ούτε η ασφάλεια του συστήματος.

Άσκηση 3

Ακολουθεί σύντομη εξήγηση του κώδικα:

Alphabet Mapping

Το λεξικό `letter_mappings` αντιστοιχίζει κάθε γράμμα του αλφαβήτου στην αριθμητική τιμή του.

Text Grouping

Η συνάρτηση `split_text_into_groups` διαιρεί ένα δεδομένο κείμενο σε ομάδες με βάση το εκάστοτε μήκος κλειδιού, όπως συμβαίνει στην στην αποκρυπτογράφηση Vigenère.

Index of Coincidence

Η συνάρτηση `calculate_index_of_coincidence` υπολογίζει το δείκτη σύμπτωσης για μια λίστα χαρακτήρων. Αυτό το μέτρο χρησιμοποιείται για να ανακαλύψουμε αν μια συμβολοσειρά πρόκειται για τυχαίους χαρακτήρες ή για αγγλικό κείμενο. Για παράδειγμα, ένα (όχι μικρό) αγγλικό κείμενο έχει $\mathbb{E}[IC] \approx 0.065$ ενώ ένα τελείως τυχαίο κείμενο έχει $\mathbb{E}[IC] \approx 0.038$.

Caesar Decryption

Η συνάρτηση `caesar_decrypt` εκτελεί αποκρυπτογράφηση Caesar σε μια ομάδα χαρακτήρων. Περιλαμβάνει μια ανάλυση συχνοτήτων για τον προσδιορισμό της πιο πιθανής μετατόπισης, λαμβάνοντας υπόψη τις συχνότητες των αγγλικών γραμμάτων.

Vigenère Decryption

Η κύρια συνάρτηση `vigenere_decrypt` εφαρμόζει την αποκρυπτογράφηση. Δοκιμάζει επαναληπτικά διαφορετικά μήκη κλειδιών, υπολογίζει το μέσο δείκτη σύμπτωσης και εκτελεί αποκρυπτογράφηση Caesar σε κάθε ομάδα για να βρει το κλειδί για την κρυπτογράφηση Vigenère.

```python
import math

# Dictionary mapping letters to their corresponding positions in the alphabet
letter_mappings = {chr(i): i - ord('A') + 1 for i in range(ord('A'), ord('Z') + 1)}

def split_text_into_groups(key, text):
    # Function to split text into groups based on a specified key length
    n = len(text)
    groups = [[] for _ in range(key)]
    for i in range(n):
        groups[i % key].append(text[i])

    return groups

def calculate_index_of_coincidence(text_list):
    # Function to calculate the index of coincidence for a given list of characters
    letter_freqs = {}
    n = len(text_list)

    # Count the frequency of each letter in the text
    for letter in text_list:
        letter_freqs[letter] = letter_freqs.get(letter, 0) + 1

    # Calculate the index of coincidence using the formula
    ic = sum((freq / n) * ((freq - 1) / (n - 1)) for freq in letter_freqs.values())
    return ic

def caesar_decrypt(group):
    # Function to perform Caesar decryption on a group of characters
    letter_freqs = {}
```

```
    english_freqs = {'A': 8.5,  'B': 2.07, 'C': 4.54, 'D': 3.38, 'E': 11.16, 'F': 1.81, 'G': 2.47, 'H': 3,
        'I': 7.54, 'J': 0.2,  'K': 1.1,  'L': 5.49, 'M': 3.01,  'N': 6.65, 'O': 7.16, 'P': 3.17,
        'Q': 0.2,  'R': 7.58, 'S': 5.74, 'T': 6.95, 'U': 3.63,  'V': 1.01, 'W': 1.29, 'X': 0.29,
        'Y': 1.78, 'Z': 0.27}

    # Count the frequency of each letter in the group
    for letter in group:
        if letter in letter_freqs:
            letter_freqs[letter] += 1
        else:
            letter_freqs[letter] = 1

    def shift(letter_freqs, n):
        # Helper function to shift the frequency of letters by a certain amount
        shifted_freqs = {}
        for letter, frequency in letter_freqs.items():
            # Convert the letter to uppercase and calculate the shifted position
            shifted_letter = chr(((ord(letter) - ord('A') + n) % 26) + ord('A'))
            # Update the frequency in the shifted dictionary
            shifted_freqs[shifted_letter] = frequency
        return shifted_freqs

    min_entropy = 2 ** 10
    n = 0

    # Find the best Caesar shift by minimizing entropy
    for i in range(1, 26):
        shifted_by_n_freqs = shift(letter_freqs, i)
        current_entropy = 0

        # Calculate the entropy for the shifted frequencies
        for letter in shifted_by_n_freqs:
            current_entropy += (shifted_by_n_freqs[letter] / len(group)) * math.log10(
                english_freqs[letter])

        current_entropy *= -1
        if current_entropy < min_entropy:
            min_entropy = current_entropy
            n = i

    c = group[0]
    p = chr(((ord(group[0]) - ord('A') + n) % 26) + ord('A'))

    # Calculate the key character for the Caesar decryption
    key_char_map = (letter_mappings[c] - letter_mappings[p]) % 26
    key_char = chr(ord('A') + key_char_map)

    return key_char

def vigenere_decrypt(cipher):
    # Main function for Vigenere decryption
    input_text = list(cipher)

    # Filter out non-alphabetic characters
    input_text_letters = [char for char in input_text if char.isalpha()]

```

```
    key_length = 2
    n = len(input_text_letters)
    results = []
    ic_counter = 0

    # Iterate until 10 possible keys are found
    while ic_counter < 10:
        decrypted_text = []
        mean_ic = 0

        # Split text into groups based on the candidate key length
        groups = split_text_into_groups(key_length, input_text_letters)

        # Calculate the mean index of coincidence for all groups
        for group in groups:
            if len(group) > 1:
                mean_ic += calculate_index_of_coincidence(group)

        # Find mean index of coincidence
        mean_ic /= key_length

        # Check if mean index of coincidence is above a threshold
        if mean_ic >= 0.06:
            vigenere_key = ""

            # Decrypt each group separately with Caesar and find the key
            for group in groups:
                vigenere_key += caesar_decrypt(group)

            key_counter = 0

            # Decrypt the text with the key
            for i in range(0, len(input_text)):
                if key_counter == len(vigenere_key):
                    key_counter = 0
                if input_text[i].isalpha():
                    decrypted_text.append(chr(
                    (letter_mappings[input_text[i]] - letter_mappings[vigenere_key[
                        key_counter]]) % 26 + ord('A')))
                    key_counter += 1
                else:
                    decrypted_text.append(input_text[i])

            decrypted_text_str = ''.join(decrypted_text)

            # Store the result in a dictionary
            result = {'vigenere_key': vigenere_key, 'decrypted_text': decrypted_text_str
                , 'mean_ic': mean_ic}
            results.append(result)
            ic_counter += 1

        key_length += 1

        # Break if the key length is equal to the length of the input text
        if key_length == n:
            break

        # Sort results in descending order of mean_ic
        results = sorted(results, key=lambda x: x['mean_ic'], reverse=True)
```

```
143
144         # Print the top 10 results
145         for i, result in enumerate(results, 1):
146             print(f"Decrypted Text {i}")
147             print("Vigenere Key:", result['vigenere_key'])
148             print("Decrypted Text:")
149             print(result['decrypted_text'])
150             print("Mean Index of Coincidence:", result['mean_ic'])
151             print("------------------------------------------------------------")
152
153
154 cipher = "ZL CXCEB IHRDF YR VYC QKWQR YJ C ICKHZXASSP ZL RKMSAYKTRNWR. HKL NIXVJDIAHUD
    SH TFTTD GPQMVRJ WTFGDKVG YYH YFHLN MV WPDF HKL NIUZEC EWPPDEVZMCL CI TOGJRLXVOO
    JYQRLRXGU DUN FTFSVAH WOO GQJR DY VLNR KTRBT VFBWDSIIYEAWF KOZKTCH WCZU DS YYCGX
    HKLI GCE ZT NGHK SR ULAW VCPTOVEZYA TDSSSGCKGDGG DZ BIOFRT VOVO NMUGCCLSUZ KRF
    TMBIIWLB XGIKXGOOZ. SR VLPC, LIFO KTRCGRTHLVXW EICPMS D UOIF WMG GSZ AITGJ MU
    VFBWDSIIYEAWF ZIWVVKH PVLJR QKEGBBNH ARI PVATLGLAI SH JCRNFH ROC FZQIKWEBDMQE
    AWTBQLVW CEB HNDSSI XJV CFNWYHVIPK MU T KUPDXGE QXZBDAEVG. RR IAS VHWI VZKT,
    MVHVBIVZAPE RHCOPQGKTGHV PX MPWMGFOWPYR VYCDKM DUN GQDNJMSU ZMMGEAT LVRD ZVQDGHX CI
    WBSXZBXGU SYYZCSJN LSFBBI EIWEMCVFCXGDQ, RAOQNSRI KFXL OQJSIPK YGM WQAY E UTGTGQH.
    ARI FVTTECSTORV FD RHASBDIT TMCMFRSVIF TMBFIQPMEVZMC GSWDYVMJ NGHALZOW GWDDKHOLCW
    CEB XGSAWORUZTT VCQAKGV SCIPSHU ZIQGJT HF FVWTWKCGL CQ VZTQJGIX GLKOW QW RWX KRYVH,
    TVNATQLUQ QQJR BTWO HXH ORLN XLFBBWKFLH PWWO DINVADFAXUSGCKGDGG. IVB QCEW
    PIDOPMEVZMCL HKLCI EFLITQWZ WYUK ZT FOGL CIELPT TUDPXWV SMIA SDCOWFIMEIWQN KRF KFT
    BBMLMXKFL DY WOSOKKKGBTHH TOWURETL. OW WBIUVLI, ACZLFIT, KFT LCOBDMQE MU LSFBBMVP
    NGHPOLWW NREH PSOS LIJZLS HHKLB ETVYH HT FVWQWEGRTHLVXW VVAWGCOVQC. EFLIXASVBETP
    AGRDWVQVCGFN BG XUKFNV RD FSHA DLG ICFNWULWIPKQ, XG HKHD MVJ SHX KRBVH KDNDLS VBML
    UVTTKS LUMSPMCCBSQJOW QE RWX GBZDIO LQTKG, DZ DS GCGBBBDAO QCEW DY HKL LIPVDXMG RM
    DINVNGHQHZCMPX. RWX PHZD OPFUC VFBWDSIIYEAWF WBSDCCB BG WOKX QW NGBJDJI: TTVTTGHLUQ
    XJV SCTIWOYVKQCS XLWYKGVZMC HT LUPSTDYIBCQ MBSO TMBFIQPMEVZMCL CYLB EP ZLHXQXYO
    GJRLCXZ. LU YVFVP IH IVL MVAGRDZFDWRC VF CCLIUL ZVKMYRR, VRDOZGI, GI BG FBBVGERAR
    BHJOWURPN YCU ARI EFKBNBLJKXKEE ETFWPOW VF QWTFH H UIA NFXVV LZ URQNL IH BR VXI GCQT
    . MVLZ SW FFLT UM VLXHKEE IAS NLI MP RBKTBFL YZGI QDFS VLMYTV AWTBQLV WWTF PL
    DUPFEVV ADNFLLB ST ICVBGWLBIF DYXE. O SYSZCKC RHBYLBWCKGDG PHAGIGE RLH DHVZPG NGIA
    BR WBMQI YRJIDPXXCEAT BG D JYQOFL DVQXYBIPTC XG PXZSRGJQ, WHKHCOV, CEB XM WV
    BXVGRJXLHLJ DS GONTVH LUSXKRJ QNGLUOWU TMCMOFAC XQ SC EHGWWYRGU JDGU HUYYIY DDK YHFC
     XQ SC IKOQZWMVKCS UM VVWI RYWHBQDS WICEQ. IAS FVCX CEB SXZDF SQRFQTW PB ARMU BCN
    WWVABMDLRXHB SYYFNVK XL O PHTST SYGKWHY DS VYC IKOQZPIT FD QNGLUOWU TMBFIQPMEVZMCL
    HR SKVIV RTESSYYGGJQXGU QLDAQIIH."
155 vigenere_decrypt(cipher)
```

Και το αποτέλεσμα φαίνεται εδώ:

```
Decrypted Text 1
Vigenere Key: DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTO
Decrypted Text:
WE STAND TODAY ON THE BRINK OF A REVOLUTION IN CRYPTOGRAPHY. THE DEVELOPMENT OF CHEAP
    DIGITAL HARDWARE HAS FREED IT FROM THE DESIGN LIMITATIONS OF MECHANICAL COMPUTING
    AND BROUGHT THE COST OF HIGH GRADE CRYPTOGRAPHIC DEVICES DOWN TO WHERE THEY CAN BE
    USED IN SUCH COMMERCIAL APPLICATIONS AS REMOTE CASH DISPENSERS AND COMPUTER
    TERMINALS. IN TURN, SUCH APPLICATIONS CREATE A NEED FOR NEW TYPES OF CRYPTOGRAPHIC
    SYSTEMS WHICH MINIMIZE THE NECESSITY OF SECURE KEY DISTRIBUTION CHANNELS AND SUPPLY
    THE EQUIVALENT OF A WRITTEN SIGNATURE. AT THE SAME TIME, THEORETICAL DEVELOPMENTS IN
     INFORMATION THEORY AND COMPUTER SCIENCE SHOW PROMISE OF PROVIDING PROVABLY SECURE
    CRYPTOSYSTEMS, CHANGING THIS ANCIENT ART INTO A SCIENCE. THE DEVELOPMENT OF COMPUTER
     CONTROLLED COMMUNICATION NETWORKS PROMISES EFFORTLESS AND INEXPENSIVE CONTACT
    BETWEEN PEOPLE OR COMPUTERS ON OPPOSITE SIDES OF THE WORLD, REPLACING MOST MAIL AND
    MANY EXCURSIONS WITH TELECOMMUNICATIONS. FOR MANY APPLICATIONS THESE CONTACTS MUST
    BE MADE SECURE AGAINST BOTH EAVESDROPPING AND THE INJECTION OF ILLEGITIMATE MESSAGES
    . AT PRESENT, HOWEVER, THE SOLUTION OF SECURITY PROBLEMS LAGS WELL BEHIND OTHER
```

```
    AREAS OF COMMUNICATIONS TECHNOLOGY. CONTEMPORARY CRYPTOGRAPHY IS UNABLE TO MEET THE
    REQUIREMENTS, IN THAT ITS USE WOULD IMPOSE SUCH SEVERE INCONVENIENCES ON THE SYSTEM
    USERS, AS TO ELIMINATE MANY OF THE BENEFITS OF TELEPROCESSING. THE BEST KNOWN
    CRYPTOGRAPHIC PROBLEM IS THAT OF PRIVACY: PREVENTING THE UNAUTHORIZED EXTRACTION OF
    INFORMATION FROM COMMUNICATIONS OVER AN INSECURE CHANNEL. IN ORDER TO USE
    CRYPTOGRAPHY TO ENSURE PRIVACY, HOWEVER, IT IS CURRENTLY NECESSARY FOR THE
    COMMUNICATING PARTIES TO SHARE A KEY WHICH IS KNOWN TO NO ONE ELSE. THIS IS DONE BY
    SENDING THE KEY IN ADVANCE OVER SOME SECURE CHANNEL SUCH AS PRIVATE COURIER OR
    REGISTERED MAIL. A PRIVATE CONVERSATION BETWEEN TWO PEOPLE WITH NO PRIOR
    ACQUAINTANCE IS A COMMON OCCURRENCE IN BUSINESS, HOWEVER, AND IT IS UNREALISTIC TO
    EXPECT INITIAL BUSINESS CONTACTS TO BE POSTPONED LONG ENOUGH FOR KEYS TO BE
    TRANSMITTED BY SOME PHYSICAL MEANS. THE COST AND DELAY IMPOSED BY THIS KEY
    DISTRIBUTION PROBLEM IS A MAJOR BARRIER TO THE TRANSFER OF BUSINESS COMMUNICATIONS
    TO LARGE TELEPROCESSING NETWORKS.
Mean Index of Coincidence: 0.06787102892366051
=========
Decrypted Text 2
Vigenere Key:
    DHKECRYPTODHKECRYPTODHKRCRYPTODHKECRYPTHDHKECRYPTODHKICRYPTODHKECRYPTODHKECRYPTOOHKECRLPTODH

Decrypted Text:
WE STAND TODAY ON THE BRINK OS A REVOLUTION IN CRFPTOGRAPHY. THE DAVELOPMENT OF CHEAP
    DIGITAL HARSWARE HNS FREED IT FROM THE DESIGN LIMITATIONS OF MEPHANICAL COMPUTINN
    AND BROUGHT THE YOST OF HIGH GRADE CRYPTOGRAPHIR DEVICRS DOWN TO WHERE THEY CAN BE
    USED IN SUCH COMMEECIAL APPLICATIONZ AS REMOTE CASH DESPENSERS AND COMPUTER
    TERMINAAS. IN TUEN, SUCH APPLICATIONS CREATE A NEED FOR NEW TYCES OF CRYPTOGRAPHPC
    SYSTEMS WHICH IINIMIZE THE NECESSITY OF SECURT KEY DIFTRIBUTION CHANNELS AND SUPPLY
    THE EQUIVALRNT OF A WRITTEN SIGUATURE. AT THE SAMA TIME, THEORETICAL DEVELOPMENTH IN
     INFBRMATION THEORY AND COMPUTER SCIENCE SHOW PEOMISE OF PROVIDINN PROVABLY SECURA
    CRYPTOSYSTEMS, CHANGING THIS ACCIENT NRT INTO A SCIENCE. THE DEVELOPMENT OF COMPUTRR
     CONTROLLED COMMBNICATION NETWONKS PROMISES EFFORTLESS AND INEMPENSIIE CONTACT
    BETWEEN PEOPLE OR COMPUTERS ON OPCOSITE SIDES OF THE DORLD, REPLACING IOST MAIL AND
    MANY EXCURSIONS WIIH TELEPOMMUNICATIONS. FOR MANY APPLICATIONS THESR CONTACTS MUST
    BE MHDE SECURE AGAINOT BOTH EAVESDROPPING AND THE INYECTIOA OF ILLEGITIMATE MESSAGES
    . AT PRESENT, HOWEVRR, THE SOLUTION OF SLCURITY PROBLEMO LAGS WELL BEHIND OTHER
    AREAS OF ROMMUNVCATIONS TECHNOLOGY. CONTEMPORARY CRYPTOGEAPHY IS UNABLE TO MLET THE
    REQUIREMANTS, IN THAT ITS USE WOULD IMPOSE HUCH SEIERE INCONVENIENCES ON THE SYSTEM
    USERS, AS TB ELIMINATE MANY OF AHE BENEFITS OF TALEPROCESSING. THE BEST KNOWN
    CRNPTOGRNPHIC PROBLEM IS THAT OF PRIVACY: PREVENTING GHE UNAUTHORIZED EETRACTION OF
    INFKRMATION FROM COMMUNICATIONS OKER AN IASECURE CHANNEL. IN ORDER TO USE
    CRYPTOGRAPHL TO ENSURE PRIVACY, OOWEVER, IT IS CURNENTLY NECESSARY FOR THE
    COMMUNXCATINT PARTIES TO SHARE A KEY WHICH IS KNOWN TO NO ONR ELSE. THIS IS DONE BF
    SENDING THE KEY EN ADVANCE OVER SOME SECURE CHANCEL SUCU AS PRIVATE COURIER OR
    REGISTERED MAIL. A PRIIATE CONVERSATION IETWEEN TWO PEOPHE WITH NO PRIOR
    ACQUAINTANCE IS P COMMOA OCCURRENCE IN BUSINESS, HOWEVER, AND IT IS UNEEALISTIC TO
    EXPECA INITIAL BUSINEOS CONTACTS TO BE POSTPONED LONG TNOUGH SOR KEYS TO BE
    TRANSMITTED BY SOME PHYSICAL MRANS. THE COST AND DESAY IMPOSED BY THES KEY
    DISTRIBUTION PROBLEM IS A BAJOR BNRRIER TO THE TRANSFER OF BUSINESS COMMUNICNTIONS
    TO LARGE TELLPROCESSING NETSORKS.
Mean Index of Coincidence: 0.06774681802545572
=========
Decrypted Text 3
Vigenere Key:
    DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRLPTPDHKECLYPTODHKECRYPTODHKECRYPTO
Decrypted Text:
WE STAND TODAY ON THE BRINK OF A REVOLUTION IN CRYPTOGRACHY. SHE DEVKLOPMENT OF CHEAP
    DIGITAL HARDWARE HAS FREED IT FROM THE DESIGN LIMITATIONS OF MECHAAICZL COMPATING
    AND BROUGHT THE COST OF HIGH GRADE CRYPTOGRAPHIC DEVICES DOWN TO WHERE THEY CAN OE
    URED IN SACH COMMERCIAL APPLICATIONS AS REMOTE CASH DISPENSERS AND COMPUTER
```

TERMINALS. IN TUEN, STCH APPRICATIONS CREATE A NEED FOR NEW TYPES OF CRYPTOGRAPHIC SYSTEMS WHICH MINIMIZE THE NEPESRITY OF YECURE KEY DISTRIBUTION CHANNELS AND SUPPLY THE EQUIVALENT OF A WRITTEN SIGNATURE. AG THD SAME TOME, THEORETICAL DEVELOPMENTS IN INFORMATION THEORY AND COMPUTER SCIENCE SHOW PROMVSE NF PROVODING PROVABLY SECURE CRYPTOSYSTEMS, CHANGING THIS ANCIENT ART INTO A SCIENCE. THE DEIELNPMENT UF COMPUTER CONTROLLED COMMUNICATION NETWORKS PROMISES EFFORTLESS AND INEXPENSIIE CNNTACT HETWEEN PEOPLE OR COMPUTERS ON OPPOSITE SIDES OF THE WORLD, REPLACING MOST MAIL AND MAAY EWCURSIUNS WITH TELECOMMUNICATIONS. FOR MANY APPLICATIONS THESE CONTACTS MUST BE MADE SECUEE AFAINST HOTH EAVESDROPPING AND THE INJECTION OF ILLEGITIMATE MESSAGES . AT PRESENT, HOWEVER, TUE SNLUTIOT OF SECURITY PROBLEMS LAGS WELL BEHIND OTHER AREAS OF COMMUNICATIONS TECHNOLOGY. COATELPORARE CRYPTOGRAPHY IS UNABLE TO MEET THE REQUIREMENTS, IN THAT ITS USE WOULD IMPOSE SUCH SEIERD INCONBENIENCES ON THE SYSTEM USERS, AS TO ELIMINATE MANY OF THE BENEFITS OF TELEPROCESSING. GHE AEST KNUWN CRYPTOGRAPHIC PROBLEM IS THAT OF PRIVACY: PREVENTING THE UNAUTHORIZED EXTRACTIBN OE INFORSATION FROM COMMUNICATIONS OVER AN INSECURE CHANNEL. IN ORDER TO USE CRYPTOGRAPHY TO RNSTRE PRIBACY, HOWEVER, IT IS CURRENTLY NECESSARY FOR THE COMMUNICATING PARTIES TO SHARE A KEY WUICG IS KNOCN TO NO ONE ELSE. THIS IS DONE BY SENDING THE KEY IN ADVANCE OVER SOME SECURE CHANNEL SUCU AS ORIVATK COURIER OR REGISTERED MAIL. A PRIVATE CONVERSATION BETWEEN TWO PEOPLE WITH NO PRIOR NCQTAINTATCE IS A COMMON OCCURRENCE IN BUSINESS, HOWEVER, AND IT IS UNREALISTIC TO EXPECT INITIAY BURINESS IONTACTS TO BE POSTPONED LONG ENOUGH FOR KEYS TO BE TRANSMITTED BY SOME PHYSICAL MEANF. THD COST ATD DELAY IMPOSED BY THIS KEY DISTRIBUTION PROBLEM IS A MAJOR BARRIER TO THE TRANSFER OS BURINESS IOMMUNICATIONS TO LARGE TELEPROCESSING NETWORKS.

Mean Index of Coincidence: 0.06707015810276683

=========

Decrypted Text 4

Vigenere Key:

DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYQVODHKECRYPTODHZECRYPTODHKECRYPTO

Decrypted Text:

WE STAND TODAY ON THE BRINK OF A REVOLUTION IN CRYPTOGRAPHY. THE DEVELNNMENT OF CHEAP DIRITAL HARDWARE HAS FREED IT FROM THE DESIGN LIMITATIONS OF MECHANICAL COMPUTING AND BROUFFT THE COST OF HIGS GRADE CRYPTOGRAPHIC DEVICES DOWN TO WHERE THEY CAN BE USED IN SUCH COMMERCIAL APPLICATINLS AS REMOTE CASH OISPENSERS AND COMPUTER TERMINALS. IN TURN, SUCH APPLICATIONS CREATE A NEED FOR NEW TYPES OE ARYPTOGRAPHIC SJSTEMS WHICH MINIMIZE THE NECESSITY OF SECURE KEY DISTRIBUTION CHANNELS AND SUPPLY THE EPSIVALENT OF A WRIETEN SIGNATURE. AT THE SAME TIME, THEORETICAL DEVELOPMENTS IN INFORMATION THEORY AND COMPTRER SCIENCE SHOW AROMISE OF PROVIDING PROVABLY SECURE CRYPTOSYSTEMS, CHANGING THIS ANCIENT ART INTO A SCIDLCE. THE DEVELOPMPNT OF COMPUTER CONTROLLED COMMUNICATION NETWORKS PROMISES EFFORTLESS AND INEXPENSIVD AONTACT BETWEEN AEOPLE OR COMPUTERS ON OPPOSITE SIDES OF THE WORLD, REPLACING MOST MAIL AND MANY EXCURSIONR UITH TELECOMMUNTCATIONS. FOR MANY APPLICATIONS THESE CONTACTS MUST BE MADE SECURE AGAINST BOTH EAVESDROONING AND THE INJENTION OF ILLEGITIMATE MESSAGES . AT PRESENT, HOWEVER, THE SOLUTION OF SECURITY PROBLEMS LAGR UELL BEHIND OTHEC AREAS OF COMMUNICATIONS TECHNOLOGY. CONTEMPORARY CRYPTOGRAPHY IS UNABLE TO MEET THE REPSIREMENTS, IN THAE ITS USE WOULD IMPOSE SUCH SEVERE INCONVENIENCES ON THE SYSTEM USERS, AS TO ELIMINATE MANY ND THE BENEFITS OF EELEPROCESSING. THE BEST KNOWN CRYPTOGRAPHIC PROBLEM IS THAT OF PRIVACY: PREVENTING THE UMYUTHORIZED EXTRLCTION OF INFORMATION FROM COMMUNICATIONS OVER AN INSECURE CHANNEL. IN ORDER TO USE CRYPTNERAPHY TO ENSURE ARIVACY, HOWEVER, IT IS CURRENTLY NECESSARY FOR THE COMMUNICATING PARTIES TO SHARE A KEY WHHAH IS KNOWN TO NO OYE ELSE. THIS IS DONE BY SENDING THE KEY IN ADVANCE OVER SOME SECURE CHANNEL SUCH AS PRIVATE CNSRIER OR REGISTECED MAIL. A PRIVATE CONVERSATION BETWEEN TWO PEOPLE WITH NO PRIOR ACQUAINTANCE IS A COMMON NACURRENCE IN BUSTNESS, HOWEVER, AND IT IS UNREALISTIC TO EXPECT INITIAL BUSINESS CONTACTS TO BE POSTPONED LNLG ENOUGH FOR KEYD TO BE TRANSMITTED BY SOME PHYSICAL MEANS. THE COST AND DELAY IMPOSED BY THIS KEY DISTRIBUTHMN PROBLEM IS A MAUOR BARRIER TO THE TRANSFER OF BUSINESS COMMUNICATIONS TO LARGE TELEPROCESSING NETWORKR.

Mean Index of Coincidence: 0.06641604010025062
=========
Decrypted Text 5
Vigenere Key: DHKECRYPTO
Decrypted Text:
WE STAND TODAY ON THE BRINK OF A REVOLUTION IN CRYPTOGRAPHY. THE DEVELOPMENT OF CHEAP DIGITAL HARDWARE HAS FREED IT FROM THE DESIGN LIMITATIONS OF MECHANICAL COMPUTING AND BROUGHT THE COST OF HIGH GRADE CRYPTOGRAPHIC DEVICES DOWN TO WHERE THEY CAN BE USED IN SUCH COMMERCIAL APPLICATIONS AS REMOTE CASH DISPENSERS AND COMPUTER TERMINALS. IN TURN, SUCH APPLICATIONS CREATE A NEED FOR NEW TYPES OF CRYPTOGRAPHIC SYSTEMS WHICH MINIMIZE THE NECESSITY OF SECURE KEY DISTRIBUTION CHANNELS AND SUPPLY THE EQUIVALENT OF A WRITTEN SIGNATURE. AT THE SAME TIME, THEORETICAL DEVELOPMENTS IN INFORMATION THEORY AND COMPUTER SCIENCE SHOW PROMISE OF PROVIDING PROVABLY SECURE CRYPTOSYSTEMS, CHANGING THIS ANCIENT ART INTO A SCIENCE. THE DEVELOPMENT OF COMPUTER CONTROLLED COMMUNICATION NETWORKS PROMISES EFFORTLESS AND INEXPENSIVE CONTACT BETWEEN PEOPLE OR COMPUTERS ON OPPOSITE SIDES OF THE WORLD, REPLACING MOST MAIL AND MANY EXCURSIONS WITH TELECOMMUNICATIONS. FOR MANY APPLICATIONS THESE CONTACTS MUST BE MADE SECURE AGAINST BOTH EAVESDROPPING AND THE INJECTION OF ILLEGITIMATE MESSAGES. AT PRESENT, HOWEVER, THE SOLUTION OF SECURITY PROBLEMS LAGS WELL BEHIND OTHER AREAS OF COMMUNICATIONS TECHNOLOGY. CONTEMPORARY CRYPTOGRAPHY IS UNABLE TO MEET THE REQUIREMENTS, IN THAT ITS USE WOULD IMPOSE SUCH SEVERE INCONVENIENCES ON THE SYSTEM USERS, AS TO ELIMINATE MANY OF THE BENEFITS OF TELEPROCESSING. THE BEST KNOWN CRYPTOGRAPHIC PROBLEM IS THAT OF PRIVACY: PREVENTING THE UNAUTHORIZED EXTRACTION OF INFORMATION FROM COMMUNICATIONS OVER AN INSECURE CHANNEL. IN ORDER TO USE CRYPTOGRAPHY TO ENSURE PRIVACY, HOWEVER, IT IS CURRENTLY NECESSARY FOR THE COMMUNICATING PARTIES TO SHARE A KEY WHICH IS KNOWN TO NO ONE ELSE. THIS IS DONE BY SENDING THE KEY IN ADVANCE OVER SOME SECURE CHANNEL SUCH AS PRIVATE COURIER OR REGISTERED MAIL. A PRIVATE CONVERSATION BETWEEN TWO PEOPLE WITH NO PRIOR ACQUAINTANCE IS A COMMON OCCURRENCE IN BUSINESS, HOWEVER, AND IT IS UNREALISTIC TO EXPECT INITIAL BUSINESS CONTACTS TO BE POSTPONED LONG ENOUGH FOR KEYS TO BE TRANSMITTED BY SOME PHYSICAL MEANS. THE COST AND DELAY IMPOSED BY THIS KEY DISTRIBUTION PROBLEM IS A MAJOR BARRIER TO THE TRANSFER OF BUSINESS COMMUNICATIONS TO LARGE TELEPROCESSING NETWORKS.
Mean Index of Coincidence: 0.06633132431168898
=========
Decrypted Text 6
Vigenere Key: DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTO
Decrypted Text:
WE STAND TODAY ON THE BRINK OF A REVOLUTION IN CRYPTOGRAPHY. THE DEVELOPMENT OF CHEAP DIGITAL HARDWARE HAS FREED IT FROM THE DESIGN LIMITATIONS OF MECHANICAL COMPUTING AND BROUGHT THE COST OF HIGH GRADE CRYPTOGRAPHIC DEVICES DOWN TO WHERE THEY CAN BE USED IN SUCH COMMERCIAL APPLICATIONS AS REMOTE CASH DISPENSERS AND COMPUTER TERMINALS. IN TURN, SUCH APPLICATIONS CREATE A NEED FOR NEW TYPES OF CRYPTOGRAPHIC SYSTEMS WHICH MINIMIZE THE NECESSITY OF SECURE KEY DISTRIBUTION CHANNELS AND SUPPLY THE EQUIVALENT OF A WRITTEN SIGNATURE. AT THE SAME TIME, THEORETICAL DEVELOPMENTS IN INFORMATION THEORY AND COMPUTER SCIENCE SHOW PROMISE OF PROVIDING PROVABLY SECURE CRYPTOSYSTEMS, CHANGING THIS ANCIENT ART INTO A SCIENCE. THE DEVELOPMENT OF COMPUTER CONTROLLED COMMUNICATION NETWORKS PROMISES EFFORTLESS AND INEXPENSIVE CONTACT BETWEEN PEOPLE OR COMPUTERS ON OPPOSITE SIDES OF THE WORLD, REPLACING MOST MAIL AND MANY EXCURSIONS WITH TELECOMMUNICATIONS. FOR MANY APPLICATIONS THESE CONTACTS MUST BE MADE SECURE AGAINST BOTH EAVESDROPPING AND THE INJECTION OF ILLEGITIMATE MESSAGES. AT PRESENT, HOWEVER, THE SOLUTION OF SECURITY PROBLEMS LAGS WELL BEHIND OTHER AREAS OF COMMUNICATIONS TECHNOLOGY. CONTEMPORARY CRYPTOGRAPHY IS UNABLE TO MEET THE REQUIREMENTS, IN THAT ITS USE WOULD IMPOSE SUCH SEVERE INCONVENIENCES ON THE SYSTEM USERS, AS TO ELIMINATE MANY OF THE BENEFITS OF TELEPROCESSING. THE BEST KNOWN CRYPTOGRAPHIC PROBLEM IS THAT OF PRIVACY: PREVENTING THE UNAUTHORIZED EXTRACTION OF INFORMATION FROM COMMUNICATIONS OVER AN INSECURE CHANNEL. IN ORDER TO USE CRYPTOGRAPHY TO ENSURE PRIVACY, HOWEVER, IT IS CURRENTLY NECESSARY FOR THE COMMUNICATING PARTIES TO SHARE A KEY WHICH IS KNOWN TO NO ONE ELSE. THIS IS DONE BY

SENDING THE KEY IN ADVANCE OVER SOME SECURE CHANNEL SUCH AS PRIVATE COURIER OR REGISTERED MAIL. A PRIVATE CONVERSATION BETWEEN TWO PEOPLE WITH NO PRIOR ACQUAINTANCE IS A COMMON OCCURRENCE IN BUSINESS, HOWEVER, AND IT IS UNREALISTIC TO EXPECT INITIAL BUSINESS CONTACTS TO BE POSTPONED LONG ENOUGH FOR KEYS TO BE TRANSMITTED BY SOME PHYSICAL MEANS. THE COST AND DELAY IMPOSED BY THIS KEY DISTRIBUTION PROBLEM IS A MAJOR BARRIER TO THE TRANSFER OF BUSINESS COMMUNICATIONS TO LARGE TELEPROCESSING NETWORKS.

Mean Index of Coincidence: 0.06604121698016241

=========

Decrypted Text 7

Vigenere Key: DHKECRYPTODHKECRYPTO

Decrypted Text:

WE STAND TODAY ON THE BRINK OF A REVOLUTION IN CRYPTOGRAPHY. THE DEVELOPMENT OF CHEAP DIGITAL HARDWARE HAS FREED IT FROM THE DESIGN LIMITATIONS OF MECHANICAL COMPUTING AND BROUGHT THE COST OF HIGH GRADE CRYPTOGRAPHIC DEVICES DOWN TO WHERE THEY CAN BE USED IN SUCH COMMERCIAL APPLICATIONS AS REMOTE CASH DISPENSERS AND COMPUTER TERMINALS. IN TURN, SUCH APPLICATIONS CREATE A NEED FOR NEW TYPES OF CRYPTOGRAPHIC SYSTEMS WHICH MINIMIZE THE NECESSITY OF SECURE KEY DISTRIBUTION CHANNELS AND SUPPLY THE EQUIVALENT OF A WRITTEN SIGNATURE. AT THE SAME TIME, THEORETICAL DEVELOPMENTS IN INFORMATION THEORY AND COMPUTER SCIENCE SHOW PROMISE OF PROVIDING PROVABLY SECURE CRYPTOSYSTEMS, CHANGING THIS ANCIENT ART INTO A SCIENCE. THE DEVELOPMENT OF COMPUTER CONTROLLED COMMUNICATION NETWORKS PROMISES EFFORTLESS AND INEXPENSIVE CONTACT BETWEEN PEOPLE OR COMPUTERS ON OPPOSITE SIDES OF THE WORLD, REPLACING MOST MAIL AND MANY EXCURSIONS WITH TELECOMMUNICATIONS. FOR MANY APPLICATIONS THESE CONTACTS MUST BE MADE SECURE AGAINST BOTH EAVESDROPPING AND THE INJECTION OF ILLEGITIMATE MESSAGES. AT PRESENT, HOWEVER, THE SOLUTION OF SECURITY PROBLEMS LAGS WELL BEHIND OTHER AREAS OF COMMUNICATIONS TECHNOLOGY. CONTEMPORARY CRYPTOGRAPHY IS UNABLE TO MEET THE REQUIREMENTS, IN THAT ITS USE WOULD IMPOSE SUCH SEVERE INCONVENIENCES ON THE SYSTEM USERS, AS TO ELIMINATE MANY OF THE BENEFITS OF TELEPROCESSING. THE BEST KNOWN CRYPTOGRAPHIC PROBLEM IS THAT OF PRIVACY: PREVENTING THE UNAUTHORIZED EXTRACTION OF INFORMATION FROM COMMUNICATIONS OVER AN INSECURE CHANNEL. IN ORDER TO USE CRYPTOGRAPHY TO ENSURE PRIVACY, HOWEVER, IT IS CURRENTLY NECESSARY FOR THE COMMUNICATING PARTIES TO SHARE A KEY WHICH IS KNOWN TO NO ONE ELSE. THIS IS DONE BY SENDING THE KEY IN ADVANCE OVER SOME SECURE CHANNEL SUCH AS PRIVATE COURIER OR REGISTERED MAIL. A PRIVATE CONVERSATION BETWEEN TWO PEOPLE WITH NO PRIOR ACQUAINTANCE IS A COMMON OCCURRENCE IN BUSINESS, HOWEVER, AND IT IS UNREALISTIC TO EXPECT INITIAL BUSINESS CONTACTS TO BE POSTPONED LONG ENOUGH FOR KEYS TO BE TRANSMITTED BY SOME PHYSICAL MEANS. THE COST AND DELAY IMPOSED BY THIS KEY DISTRIBUTION PROBLEM IS A MAJOR BARRIER TO THE TRANSFER OF BUSINESS COMMUNICATIONS TO LARGE TELEPROCESSING NETWORKS.

Mean Index of Coincidence: 0.06597657835385747

=========

Decrypted Text 8

Vigenere Key: DHKECRYPTODHKECRYPTODHKECRYPTO

Decrypted Text:

WE STAND TODAY ON THE BRINK OF A REVOLUTION IN CRYPTOGRAPHY. THE DEVELOPMENT OF CHEAP DIGITAL HARDWARE HAS FREED IT FROM THE DESIGN LIMITATIONS OF MECHANICAL COMPUTING AND BROUGHT THE COST OF HIGH GRADE CRYPTOGRAPHIC DEVICES DOWN TO WHERE THEY CAN BE USED IN SUCH COMMERCIAL APPLICATIONS AS REMOTE CASH DISPENSERS AND COMPUTER TERMINALS. IN TURN, SUCH APPLICATIONS CREATE A NEED FOR NEW TYPES OF CRYPTOGRAPHIC SYSTEMS WHICH MINIMIZE THE NECESSITY OF SECURE KEY DISTRIBUTION CHANNELS AND SUPPLY THE EQUIVALENT OF A WRITTEN SIGNATURE. AT THE SAME TIME, THEORETICAL DEVELOPMENTS IN INFORMATION THEORY AND COMPUTER SCIENCE SHOW PROMISE OF PROVIDING PROVABLY SECURE CRYPTOSYSTEMS, CHANGING THIS ANCIENT ART INTO A SCIENCE. THE DEVELOPMENT OF COMPUTER CONTROLLED COMMUNICATION NETWORKS PROMISES EFFORTLESS AND INEXPENSIVE CONTACT BETWEEN PEOPLE OR COMPUTERS ON OPPOSITE SIDES OF THE WORLD, REPLACING MOST MAIL AND MANY EXCURSIONS WITH TELECOMMUNICATIONS. FOR MANY APPLICATIONS THESE CONTACTS MUST BE MADE SECURE AGAINST BOTH EAVESDROPPING AND THE INJECTION OF ILLEGITIMATE MESSAGES. AT PRESENT, HOWEVER, THE SOLUTION OF SECURITY PROBLEMS LAGS WELL BEHIND OTHER

AREAS OF COMMUNICATIONS TECHNOLOGY. CONTEMPORARY CRYPTOGRAPHY IS UNABLE TO MEET THE REQUIREMENTS, IN THAT ITS USE WOULD IMPOSE SUCH SEVERE INCONVENIENCES ON THE SYSTEM USERS, AS TO ELIMINATE MANY OF THE BENEFITS OF TELEPROCESSING. THE BEST KNOWN CRYPTOGRAPHIC PROBLEM IS THAT OF PRIVACY: PREVENTING THE UNAUTHORIZED EXTRACTION OF INFORMATION FROM COMMUNICATIONS OVER AN INSECURE CHANNEL. IN ORDER TO USE CRYPTOGRAPHY TO ENSURE PRIVACY, HOWEVER, IT IS CURRENTLY NECESSARY FOR THE COMMUNICATING PARTIES TO SHARE A KEY WHICH IS KNOWN TO NO ONE ELSE. THIS IS DONE BY SENDING THE KEY IN ADVANCE OVER SOME SECURE CHANNEL SUCH AS PRIVATE COURIER OR REGISTERED MAIL. A PRIVATE CONVERSATION BETWEEN TWO PEOPLE WITH NO PRIOR ACQUAINTANCE IS A COMMON OCCURRENCE IN BUSINESS, HOWEVER, AND IT IS UNREALISTIC TO EXPECT INITIAL BUSINESS CONTACTS TO BE POSTPONED LONG ENOUGH FOR KEYS TO BE TRANSMITTED BY SOME PHYSICAL MEANS. THE COST AND DELAY IMPOSED BY THIS KEY DISTRIBUTION PROBLEM IS A MAJOR BARRIER TO THE TRANSFER OF BUSINESS COMMUNICATIONS TO LARGE TELEPROCESSING NETWORKS.

Mean Index of Coincidence: 0.06502203419707386

=========

Decrypted Text 9

Vigenere Key: DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTO

Decrypted Text:

WE STAND TODAY ON THE BRINK OF A REVOLUTION IN CRYPTOGRAPHY. THE DEVELOPMENT OF CHEAP DIGITAL HARDWARE HAS FREED IT FROM THE DESIGN LIMITATIONS OF MECHANICAL COMPUTING AND BROUGHT THE COST OF HIGH GRADE CRYPTOGRAPHIC DEVICES DOWN TO WHERE THEY CAN BE USED IN SUCH COMMERCIAL APPLICATIONS AS REMOTE CASH DISPENSERS AND COMPUTER TERMINALS. IN TURN, SUCH APPLICATIONS CREATE A NEED FOR NEW TYPES OF CRYPTOGRAPHIC SYSTEMS WHICH MINIMIZE THE NECESSITY OF SECURE KEY DISTRIBUTION CHANNELS AND SUPPLY THE EQUIVALENT OF A WRITTEN SIGNATURE. AT THE SAME TIME, THEORETICAL DEVELOPMENTS IN INFORMATION THEORY AND COMPUTER SCIENCE SHOW PROMISE OF PROVIDING PROVABLY SECURE CRYPTOSYSTEMS, CHANGING THIS ANCIENT ART INTO A SCIENCE. THE DEVELOPMENT OF COMPUTER CONTROLLED COMMUNICATION NETWORKS PROMISES EFFORTLESS AND INEXPENSIVE CONTACT BETWEEN PEOPLE OR COMPUTERS ON OPPOSITE SIDES OF THE WORLD, REPLACING MOST MAIL AND MANY EXCURSIONS WITH TELECOMMUNICATIONS. FOR MANY APPLICATIONS THESE CONTACTS MUST BE MADE SECURE AGAINST BOTH EAVESDROPPING AND THE INJECTION OF ILLEGITIMATE MESSAGES. AT PRESENT, HOWEVER, THE SOLUTION OF SECURITY PROBLEMS LAGS WELL BEHIND OTHER AREAS OF COMMUNICATIONS TECHNOLOGY. CONTEMPORARY CRYPTOGRAPHY IS UNABLE TO MEET THE REQUIREMENTS, IN THAT ITS USE WOULD IMPOSE SUCH SEVERE INCONVENIENCES ON THE SYSTEM USERS, AS TO ELIMINATE MANY OF THE BENEFITS OF TELEPROCESSING. THE BEST KNOWN CRYPTOGRAPHIC PROBLEM IS THAT OF PRIVACY: PREVENTING THE UNAUTHORIZED EXTRACTION OF INFORMATION FROM COMMUNICATIONS OVER AN INSECURE CHANNEL. IN ORDER TO USE CRYPTOGRAPHY TO ENSURE PRIVACY, HOWEVER, IT IS CURRENTLY NECESSARY FOR THE COMMUNICATING PARTIES TO SHARE A KEY WHICH IS KNOWN TO NO ONE ELSE. THIS IS DONE BY SENDING THE KEY IN ADVANCE OVER SOME SECURE CHANNEL SUCH AS PRIVATE COURIER OR REGISTERED MAIL. A PRIVATE CONVERSATION BETWEEN TWO PEOPLE WITH NO PRIOR ACQUAINTANCE IS A COMMON OCCURRENCE IN BUSINESS, HOWEVER, AND IT IS UNREALISTIC TO EXPECT INITIAL BUSINESS CONTACTS TO BE POSTPONED LONG ENOUGH FOR KEYS TO BE TRANSMITTED BY SOME PHYSICAL MEANS. THE COST AND DELAY IMPOSED BY THIS KEY DISTRIBUTION PROBLEM IS A MAJOR BARRIER TO THE TRANSFER OF BUSINESS COMMUNICATIONS TO LARGE TELEPROCESSING NETWORKS.

Mean Index of Coincidence: 0.06493772893772894

=========

Decrypted Text 10

Vigenere Key: DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTO

Decrypted Text:

WE STAND TODAY ON THE BRINK OF A REVOLUTION IN CRYPTOGRAPHY. THE DEVELOPMENT OF CHEAP DIGITAL HARDWARE HAS FREED IT FROM THE DESIGN LIMITATIONS OF MECHANICAL COMPUTING AND BROUGHT THE COST OF HIGH GRADE CRYPTOGRAPHIC DEVICES DOWN TO WHERE THEY CAN BE USED IN SUCH COMMERCIAL APPLICATIONS AS REMOTE CASH DISPENSERS AND COMPUTER TERMINALS. IN TURN, SUCH APPLICATIONS CREATE A NEED FOR NEW TYPES OF CRYPTOGRAPHIC SYSTEMS WHICH MINIMIZE THE NECESSITY OF SECURE KEY DISTRIBUTION CHANNELS AND SUPPLY THE EQUIVALENT OF A WRITTEN SIGNATURE. AT THE SAME TIME, THEORETICAL DEVELOPMENTS IN

```
    INFORMATION THEORY AND COMPUTER SCIENCE SHOW PROMISE OF PROVIDING PROVABLY SECURE
   CRYPTOSYSTEMS, CHANGING THIS ANCIENT ART INTO A SCIENCE. THE DEVELOPMENT OF COMPUTER
    CONTROLLED COMMUNICATION NETWORKS PROMISES EFFORTLESS AND INEXPENSIVE CONTACT
   BETWEEN PEOPLE OR COMPUTERS ON OPPOSITE SIDES OF THE WORLD, REPLACING MOST MAIL AND
   MANY EXCURSIONS WITH TELECOMMUNICATIONS. FOR MANY APPLICATIONS THESE CONTACTS MUST
   BE MADE SECURE AGAINST BOTH EAVESDROPPING AND THE INJECTION OF ILLEGITIMATE MESSAGES
   . AT PRESENT, HOWEVER, THE SOLUTION OF SECURITY PROBLEMS LAGS WELL BEHIND OTHER
   AREAS OF COMMUNICATIONS TECHNOLOGY. CONTEMPORARY CRYPTOGRAPHY IS UNABLE TO MEET THE
   REQUIREMENTS, IN THAT ITS USE WOULD IMPOSE SUCH SEVERE INCONVENIENCES ON THE SYSTEM
   USERS, AS TO ELIMINATE MANY OF THE BENEFITS OF TELEPROCESSING. THE BEST KNOWN
   CRYPTOGRAPHIC PROBLEM IS THAT OF PRIVACY: PREVENTING THE UNAUTHORIZED EXTRACTION OF
   INFORMATION FROM COMMUNICATIONS OVER AN INSECURE CHANNEL. IN ORDER TO USE
   CRYPTOGRAPHY TO ENSURE PRIVACY, HOWEVER, IT IS CURRENTLY NECESSARY FOR THE
   COMMUNICATING PARTIES TO SHARE A KEY WHICH IS KNOWN TO NO ONE ELSE. THIS IS DONE BY
   SENDING THE KEY IN ADVANCE OVER SOME SECURE CHANNEL SUCH AS PRIVATE COURIER OR
   REGISTERED MAIL. A PRIVATE CONVERSATION BETWEEN TWO PEOPLE WITH NO PRIOR
   ACQUAINTANCE IS A COMMON OCCURRENCE IN BUSINESS, HOWEVER, AND IT IS UNREALISTIC TO
   EXPECT INITIAL BUSINESS CONTACTS TO BE POSTPONED LONG ENOUGH FOR KEYS TO BE
   TRANSMITTED BY SOME PHYSICAL MEANS. THE COST AND DELAY IMPOSED BY THIS KEY
   DISTRIBUTION PROBLEM IS A MAJOR BARRIER TO THE TRANSFER OF BUSINESS COMMUNICATIONS
   TO LARGE TELEPROCESSING NETWORKS.
Mean Index of Coincidence: 0.06347793845012976
=========
```

Παρατήρηση Βλέπουμε πως τα 10 πιο πιθανά κλειδιά είναι:

1. DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTO
2. DHKECRYPTODHKECRYPTODHKRCRYPTODHKECRYPTHDHKECRYPTODHKICRYPTODHKECRYPTODHKECRYPTOOHI
3. DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRLPTPDHKECLYPTODHKECRYPTODHKECRYPTO
4. DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYQVODHKECRYPTODHZECRYPTODH
5. DHKECRYPTO
6. DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTO
7. DHKECRYPTODHKECRYPTO
8. DHKECRYPTODHKECRYPTODHKECRYPTO
9. DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTO
10. DHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTODHKECRYPTO

Τα περισσότερα είναι η λέξη ”DHKECRYPTO” επαναλαμβανόμενη. Αυτό σημαίνει ότι έχει βρεθεί το βασικό κλειδί και κάθε φορά δοκιμάζονται οι πιο πιθανές περίοδοι.

Άσκηση 4

Ερώτημα 1

Θα δείξουμε ότι σε ένα κρυπτοσύστημα που διαθέτει τέλεια μυστικότητα, **δεν** είναι αναγκαίο κάθε κλειδί να επιλέγεται με την ίδια πιθανότητα, χρησιμοποιώντας ένα (αντι-)παράδειγμα.
Έστω το τροποποιημένο κρυπτοσύστημα 1-bit OTP με message space $\mathcal{M} = \{0, 1\}$ και key space $\mathcal{K} = \{00, 01, 10, 11\}$ (αντί για $\mathcal{K} = \{0, 1\}$), το οποίο κρυπτογραφεί κάποιο μήνυμα m κάνοντάς το 'XOR' με το πρώτο bit κάποιου κλειδιού k. Είναι προφανές ότι μπορούμε έχουμε μη ομοιόμορφη κατανομή κλειδιών και ταυτόχρονα perfect secrecy, με την προϋπόθεση η πιθανότητα να επιλεγεί κάποιο κλειδί με πρώτο bit 0 είναι ίση με την πιθανότητα να επιλεγεί κάποιο κλειδί με πρώτο bit 1 και μάλιστα να είναι 0.5. Έτσι, μπορούμε τα έχουμε διαφορετικές πιθανότητες για τα κλειδιά, για παράδειγμα:

$$\mathbb{P}(k = 00) = 1/8, \mathbb{P}(k = 01) = 3/8, \mathbb{P}(k = 10) = 1/8, \mathbb{P}(k = 11) = 3/8$$

Βλέπουμε ότι:

$$\mathbb{P}(M = m \mid C = c) = \frac{\mathbb{P}[C = c \mid M = m]\mathbb{P}[M = m]}{\mathbb{P}[C = c]}$$

$$\mathbb{P}[C = c \mid M = m] = \mathbb{P}[K[0] = k[0] = c \oplus m] = \sum_{k \text{ s.t. } k[0]=c\oplus m} \mathbb{P}[K = k] = \frac{1}{8} + \frac{3}{8} = \frac{1}{2}$$

$$\mathbb{P}[C = c] = \sum_{m\in\mathcal{M}} \mathbb{P}[M = m] \cdot \mathbb{P}[K[0] = k[0] = c \oplus m] = \sum_{m\in\mathcal{M}} \mathbb{P}[M = m] \cdot \sum_{k \text{ s.t. } k[0]=c\oplus m} \mathbb{P}[K = k]$$
$$= \frac{1}{2} \cdot \left(\frac{1}{8} + \frac{3}{8}\right) + \frac{1}{2} \cdot \left(\frac{1}{8} + \frac{3}{8}\right) = \frac{1}{2}$$

Τελικά δείξαμε ότι το κρυπτοσύστημα έχει perfect secrecy, παρόλο που η κατανομή των κλειδιών δεν είναι ομοιόμορφη

$$\mathbb{P}(M = m \mid C = c) = \frac{\mathbb{P}[C = c \mid M = m]\mathbb{P}[M = m]}{\mathbb{P}[C = c]} = \frac{\frac{1}{2}\mathbb{P}[M = m]}{\frac{1}{2}} = \mathbb{P}[M = m]$$

Η ιδέα αυτή μπορεί να γενικευτεί για το n-bit OTP, θεωρώντας χώρο κλειδιών πληθικότητας $n + 1$ (γενικά $n + k$ αλλά $n + 1$ αρκεί για την απόδειξη ότι δεν απαιτείται ομοιόμορφη κατανομή κλειδιών). Η προϋπόθεση τώρα είναι τα πρώτα n bits των κλειδιών να κατανέμονται ομοιόμορφα και το τελευταίο όχι, ώστε τελικά να ισχύει η σχέση του Shannon για perfect secrecy, ενώ η κατανομή των κλειδιών δεν είναι ομοιόμορφη.

Ερώτημα 2

Από τη συνθήκη τέλειας μυστικότητας του Shannon, έχουμε:

$$\forall x \in M, y \in C : \Pr[M = x \mid C = y\} = \Pr[M = x]$$

Σχέση i

Χρησιμοποιώντας τον ορισμό της δεσμευμένης πιθανότητας, η συνθήκη του Shannon γίνεται:

$$\forall x \in M, y \in C : \Pr[M = x] = \Pr[M = x \mid C = y] = \frac{\Pr[M = x, C = y]}{\Pr[M = x]}$$

Όμως τα ενδεχόμενα $M = x, C = y$ είναι ανεξάρτητα, συνεπώς:

$$\forall x \in M, y \in C : \Pr[M = x] = \Pr[M = x \mid C = y] = \frac{\Pr[M = x]\Pr[C = y]}{\Pr[M = x]}$$

Οπότε δείξαμε ότι η πρώτη πρόταση απορρέει από τον κανόνα τέλειας μυστικότητας του Shannon, δηλαδή:

$$\forall x \in M, y \in C : \Pr[M = x \mid C = y] = \Pr[M = x] \Rightarrow$$
$$\forall x \in M, y \in C : \Pr[M = x] = \Pr[M = x \mid C = y]$$

Σχέση ii

Χρησιμοποιώντας την προηγούμενη σχέση που αποδείξαμε, εύκολα μπορούμε να οδηγηθούμε στην επόμενη, αν τη θεωρήσουμε για 2 μηνύματα:

$$\forall x_1, x_2 \in M, y \in C \Pr[C = y \mid M = x_1] = \Pr[C = y \mid M = x_2] = \Pr[C = y]$$

Άσκηση 5

Έστω το κρυπτοσύστημα OTP όπου το κλειδί επιλέγεται ομοιόμορφα από το σύνολο $\{0,1\}^\lambda$. Σε αυτήν την περίπτωση, η κατανομή των ciphertexts είναι επίσης ομοιόμορφη καθώς για κάθε plaintext m υπάρχει μοναδικό κλειδί k τέτοιο ώστε $Enc(k,m) = k \oplus m = c$, όπου c το αντίστοιχο ciphertext του m και αυτό ισχύει για όλα τα δυνατά ciphertexts.

Έστω τώρα το τροποποιημένο σύστημα της Αλίκης. Το κλειδί πλέον δεν μπορεί να λάβει την τιμή 0^λ, άρα επιλέγεται από το χώρο $\{0,1\}^\lambda \backslash \{0^\lambda\}$. Για οποιοδήποτε plaintext m, υπάρχει κάποιο μοναδικό κλειδί k ώστε $Enc(k,m) = k \oplus m = c$, για κάποιο ciphertext c εκτός από την περίπτωση $c = m$, αφού προκύπτει μόνο όταν $k = 0^\lambda$, το οποίο έχουμε αποκλείσει από το χώρο των κλειδιών.
Όμως, αφού έχουμε αποκλείσει το $k = 0^\lambda$, η κατανομή των ciphertexts παύει να είναι ομοιόμορφη, αφού με πλήθος κλειδιών μικρότερο από το πλήθος των δυνατών plaintexts και ciphertexts $|\mathcal{K}| < |\mathcal{M}| = |\mathcal{C}|$, δεν μπορούμε να αντιστοιχίσουμε κάθε plaintext σε μοναδικό ciphertext (αρχή περιστεροφωλιάς). Συνεπώς, υπάρχει ciphertext $\overline{c}$ με μηδενική πιθανότητα να εμφανιστεί. Άρα, για οποιοδήποτε ciphertext $c \neq \overline{c}$ και για το ciphertext που δε θα προκύψει $\overline{c}$ έχουμε

$$\mathbb{P}[c] = \frac{1}{2^\lambda - 1} \neq 0 = \mathbb{P}[\overline{c}]$$

Άρα η κατανομή των ciphertexts δεν είναι ομοιόμορφη με αποτέλεσμα να μην έχουμε perfect secrecy στο νέο σύστημα.

Άσκηση 6

Έστω η πολλαπλασιαστική εκδοχή του one-time pad. Αν p πρώτος τότε η κρυπτογράφηση του plaintext m με κλειδί k, $k, m \in \mathbb{Z}_p^*$ ορίζεται ως $Enc(k, m) \equiv km \pmod p$

Ερώτημα 1

Για να βρούμε τη συνάρτηση αποκρυπτογράφησης, αρκεί να λύσουμε τη συνάρτηση κρυπτογράφησης ως προς το μήνυμα:

$$c = Enc(k, m) \equiv km \Rightarrow m \equiv k^{-1}c \pmod p$$

Όπου k^{-1} ο αντίστροφος του $k \pmod p$ και η λύση είναι μοναδική επειδή $\gcd(k, p) = 1$ (αφού p πρώτος και $k < p$).

Ερώτημα 2

Για perfect secrecy, πρέπει να δείξουμε ότι για ένα τυχαία επιλεγμένο κλειδί από ομοιόμορφη κατανομή κλειδιών η πιθανότητα να προκύψει κάποιο από τα ciphertexts είναι ίδια για καθένα, δηλαδή ότι και η κατανομή των ciphertexts είναι ομοιόμορφη.

$$\text{Για } k \xleftarrow{R} \mathcal{K}, \text{μηνύματα } m_1, m_2 \in \mathcal{M}, \text{κρυπτοκείμενο } c \in \mathcal{C} \text{πρέπει } \mathbb{P}[Enc(k, m_1) = c] = \mathbb{P}[Enc(k, m_2) = c]$$

Ισοδύναμα, μπορούμε να πούμε ότι

$$\forall(\text{fixed}) \, c, m \in GF(p). \exists! k \in \mathbb{Z}_p \text{ so that } km \equiv c \Rightarrow k \equiv m^{-1}c \pmod p$$

Άρα, αφού $\gcd(m, p) = 1$, η αποκρυπτογράφηση οδηγεί στο αρχικό μήνυμα μοναδικά, άρα το σύστημα είναι ορθό.

Ερώτημα 3

Θα αξιποιήσουμε το εξής θεώρημα: Αν το κρυπτοσύστημα έχει $\mid \mathcal{M} \mid = \mid \mathcal{C} \mid = \mid \mathcal{K} \mid$, τότε έχει τέλεια μυστικότητα αν ισχύουν τα εξής:

1. $\forall m \in \mathcal{M}, \forall c \in \mathcal{C}, \exists! k \in \mathcal{K}$ so that $Enc_k(m) = c$
2. κάθε κλειδί επιλέγεται με την ίδια πιθανότητα, συγκεκριμένα $1/ \mid \mathcal{K} \mid$

Πρόταση 1 Έστω ότι δεν ισχύει, ότι δηλαδή υπάρχουν τουλάχιστον 2 κλειδιά k_1, k_2 τέτοια ώστε

$$c = Enc_{k_1}(m) = Enc_{k_2}(m) \Rightarrow k_1 m \equiv k_2 m \pmod p$$

Και επειδή $\gcd(m, p) = 1$, αφού $p : prime \wedge m < p$, μπορούμε να διαιρέσουμε κατά μέλη με m και παίρνουμε $k_1 \equiv k_2 \pmod p$, δηλαδή καταλήξαμε σε άτοπο.
Άρα, η πρόταση (1) είναι αληθής.

Πρόταση 2 Αφού το κρυπτοσύστημα είναι τροποποίηση του One Time Pad, θα έχει ομοιόμορφη κατανομή κλειδιών (όπως το OTP).
Άρα, και η πρόταση (2) είναι αληθής.

Οι προϋποθέσεις του θεωρήματος ικανοποιούνται, άρα αποφαινόμαστε ότι το κρυπτοσύστημα είναι τέλεια ασφαλές!

Άσκηση 7

Υποερώτημα 1

Θεωρούμε τον πρώτο αριθμό $P = 2^n - 1, n \in \mathbb{N}^*$. Έστω ότι n δεν είναι πρώτος. Άρα μπορούμε να τον εκφράσουμε ως $n = xy, x, y \in \mathbb{N}$, οπότε έχουμε:

$$P = 2^n - 1 = 2^{xy} - 1 = (2^x)^y - 1^y = (2^x - 1)[(2^x)^{y-1} + (2^x)^{y-2} + \ldots + 2^x + 1]$$

Η παραπάνω σχέση προέκυψε βγάζοντας κοινό παράγοντα το $(2^x - 1)$.

Όμως, $1 < x, y < n$ (όχι τετριμμένη περίπτωση), άρα $1 < 2^x - 1 < 2^n - 1$ οπότε υποχρεωτικά και

$$1 < (2^x)^{y-1} + (2^x)^{y-2} + \cdots + 2^x + 1 < 2^n - 1$$

άρα δείξαμε ότι υπάρχει μια παραγοντοποίηση του P με δύο ακεραίους, έστω a, b, με $1 < a, b < P$, οπότε ο P δεν είναι πρώτος και συνεπώς καταλήξαμε σε άτοπο!

Άρα η πρόταση "Εστω $2^n - 1$ πρώτος, τότε και n πρώτος" είναι αληθής!

Υποερώτημα 2

Σχέση i

Έστω $p \in \mathbb{N}^+$ περιττός πρώτος και $M_p = 2^p - 1$.
Το μικρό θεώρημα Fermat μας δίνει $2^p \equiv 2 \Rightarrow 2^p - 1 \equiv 1 \Rightarrow M_p \equiv 1 \pmod{p}$.

Σχέση ii

Έστω $q : prime$ που διαιρεί τον M_p. Δηλαδή είναι $q \mid 2^p - 1 \Rightarrow 2^p - 1 \equiv 0 \Rightarrow 2^p \equiv 1 \pmod{q}$.
Όμως, $2 \in \mathbb{Z}_q$ και άρα η τάξη του (από τον ορισμό της) διαιρεί κάθε αριθμό k για τον οποιο είναι $2^k \equiv 1 \pmod{q}$, οπότε είναι και $ord(2) \mid p \xrightarrow{p:\text{prime}} ord(2) = p$.

Το μικρό θεώρημα Fermat μας δίνει: $2^q \equiv 2 \Rightarrow 2^{q-1} \equiv 1 \pmod{q}$, οπότε τελικά είναι $p \mid q - 1 \Leftrightarrow ord(2) \mid q - 1$ και $2 \mid q - 1$.
Δείξαμε συνολικά ότι $\exists m \in \mathbb{Z} : 2pm = q - 1$. Ακόμα έχουμε:

$$\begin{cases} M_p : prime \Rightarrow \phi(M_p) = 2^p - 2, p \mid 2^p - 2 = \phi(M_p) \\ M_p : composite \Rightarrow M_p = k \cdot q^\lambda \Rightarrow \phi(M_p) = \phi(q^\lambda)\phi(k) = \phi(k)q^k \left(1 - \frac{1}{q}\right) = \phi(k)\frac{q^k}{q}(q-1) = \phi(k)2pcq^{k-1} \end{cases}$$

Δηλαδή σε κάθε περίπτωση $p \mid \phi(M_p)$

Άσκηση 8

Θεωρούμε p και q διαφορετικούς, πρώτους. Από το μικρό θεώρημα Fermat, έχουμε για τα p και q:

$$p^q \equiv p \pmod{q} \Rightarrow p^{(q-1)} \equiv 1 \pmod{q} \Rightarrow q|p^{(q-1)} - 1 \tag{1}$$

$$q^p \equiv q \pmod{p} \Rightarrow q^{(p-1)} \equiv 1 \pmod{p} \Rightarrow p|q^{(p-1)} - 1 \tag{2}$$

Ακόμα, προφανώς ισχύουν τα εξής:

$$q|q^{(p-1)} \tag{3}$$

$$p|p^{(q-1)} \tag{4}$$

Συνδυάζοντας γραμμικά (πρόσθεση διαιρετέων) τις σχέσεις (1), (3) και (2), (4), έχουμε:

$$(1,3) \Rightarrow q|p^{(q-1)} - 1 + q^{(p-1)}$$

$$(2,4) \Rightarrow p|q^{(p-1)} - 1 + p^{(q-1)}$$

Οι τελευταίες σχέσεις σημαίνουν ότι υπάρχουν ακέραιες σταθερές c και c' τέτοιες ώστε:

$$p^{(q-1)} - 1 + q^{(p-1)} = cq$$

$$q^{(p-1)} - 1 + p^{(q-1)} = c'p$$

Άρα ισχύει και ότι $c'p = cq \Rightarrow$. Επειδή $p, q : primes \Rightarrow p \nmid q \wedge q \nmid p$, μπορούμε να συμπεράνουμε ότι $p \mid c \wedge q \mid c'$, άρα, χρησιμοποιώντας τη μία σχέση γράφουμε $kp = c$ και αντικαθιστούμε:

$$q^{(p-1)} - 1 + p^{(q-1)} = cq \Rightarrow q^{(p-1)} - 1 + p^{(q-1)} = (kp)q \Rightarrow$$
$$pq \mid q^{(p-1)} - 1 + p^{(q-1)} \Rightarrow q^{(p-1)} - 1 + p^{(q-1)} \equiv 0 \Rightarrow q^{(p-1)} + p^{(q-1)} \equiv 1 \pmod{pq}$$

Άσκηση 9

Μπορούμε εύκολα να δείξουμε τη μία σχέση:

$$\sum_{\beta \in \mathbb{Z}_p^*} \beta = 1 + 2 + \cdots + (p-1) = \frac{(p-1)p}{2}$$

Όμως, αφού ο p είναι περιττός πρώτος, έχουμε ότι ο $p-1$ είναι άρτιος, άρα ο $\dfrac{p-1}{2}$ είναι κάποιος ακέραιος, έστω k. Συνεπώς, βλέπουμε ότι είναι:

$$\sum_{\beta \in \mathbb{Z}_p^*} \beta = k \cdot p \equiv 0 \pmod{p}$$

Έστω τώρα η πολλαπλασιαστική ομάδα $\mathbb{Z}_p^*$. $\forall \beta \in \mathbb{Z}_p^*$ το β έχει μοναδικό αντίστροφο $\beta^{-1} \in \mathbb{Z}_p^*$. Οπότε, όλα τα στοιχεία της ομάδας αντιστοιχίζονται μοναδικά σε στοιχεία της ίδιας ομάδας (σε κάποια μετάθεση αυτών, δηλαδή). Οπότε, το άθροισμα των αντιστρόφων θα περιλαμβάνει μόνο όρους που ανήκουν στην ίδια ομάδα $\mathbb{Z}_p^*$. Επομένως δείξαμε ότι:

$$\sum_{\beta \in \mathbb{Z}_p^*} \beta^{-1} = \sum_{k \in \mathbb{Z}_p^*} k = \sum_{\beta \in \mathbb{Z}_p^*} \beta \equiv 0 \pmod{p}$$

Άσκηση 10

Υποερώτημα 1

Έστω φυσικός $n > 3$ και $m = \lfloor \sqrt{n} \rfloor$. Θα δείξουμε ότι:

$$n : prime \Leftrightarrow \sum_{j=1}^{m} \gcd(n, j) = m.$$

Το "ευθύ" είναι προφανές. Αφού ο n είναι πρώτος, τότε για κάθε $j \in \{1, 2, \dots m\}$ με $m < n$, είναι $\gcd(n, j) = 1$, γιατί σε άλλη περίπτωση, ο n θα ήταν σύνθετος. Οπότε έχουμε:

$$\sum_{j=1}^{m} \gcd(n, j) = \sum_{j=1}^{m} 1 = m$$

Για το "αντίστροφο", έστω $\sum_{j=1}^{m} \gcd(n, j) = m$

Γνωρίζουμε ότι $\forall j.\ \gcd(n, j) \geq 1$, συνεπώς από την υπόθεση συμπεραίνουμε ότι:

$$\forall j \in \{1, 2, \dots, m\}.\ \gcd(n, j) \geq 1 \wedge \sum_{j=1}^{m} \gcd(n, j) = m \Rightarrow \forall j \in \{1, 2, \dots, m\}.\ \gcd(n, j) = 1.$$

Άρα δείξαμε ότι οι αριθμοί $2, 3, \dots, m = \lfloor \sqrt{n} \rfloor$ δε διαιρούν τον n.
Ακόμα, εύκολα βλέπουμε πως κάθε φυσικός N με $\lfloor \sqrt{n} \rfloor + 1 \leq N < n$ δεν μπορεί να διαιρεί τον n, διότι αν τον διαιρούσε, θα είχαμε μια παραγοντοποίηση $n = N \cdot c$, όπου για τον c είναι:

$$c = \frac{n}{N} \xrightarrow{N \geq \lfloor \sqrt{n} \rfloor + 1} c \leq \lfloor \sqrt{n} \rfloor$$

Το οποίο όμως είναι αδύνατο, αφού αρχικά δείξαμε ότι κανένας φυσικός μικρότερος του $\lfloor \sqrt{n} \rfloor$ (εξαιρουμένων $0, 1$) διαιρεί τον n.

Αποδείξαμε δηλαδή ότι $2, 3, \dots, n - 1 \nmid n$, άρα ο n είναι πρώτος!

Υποερώτημα 2

Υλοποιήσαμε σε Python έναν αλγόριθμο ελέγχου πρώτων βάσει του κριτηρίου στο Υποερώτημα 1. Βέβαια, αντί να υπολογίζουμε το άθροισμα των μέγιστων κοινών διαιρετών του n με κάθε $j \in \{1, 2, \dots m = \lfloor \sqrt{n} \rfloor\}$ και να ελέγχουμε αν είναι μεγαλύτερο από το m, υπολογίζουμε κάθε φορά το υπόλοιπο της διαίρεσης με το j και αν είναι 0 σημαίνει πως ο μέγιστος κοινός διαιρέτης είναι μεγαλύτερος του 1, άρα το άθροισμα θα προκύψει μεγαλύτερο του m, οπότε είναι ισοδύναμο με το κριτήριο. Ο αλγόριθμος αυτός έχει πολυπλοκότητα $\mathcal{O}(\sqrt{n})$, αφού κάνει το πολύ $\sqrt{n}$ επαναλήψεις πριν τερματίσει.

```python
import math

def is_prime(n):
    if n <= 1:
        return False
    if n <= 3:
        return True
    if n % 2 == 0:
        return False
    for i in range(3, int(math.sqrt(n)) + 1, 2):
        if n % i == 0:
            return False
    return True
```

Παραθέτουμε και την υλοποίηση του αλγορίθμου Miller-Rabin σε Python. Η πολυπλοκότητά του είναι $\mathcal{O}(k \cdot \log^3(n))$. Δηλαδή καλύτερος αλόριθμος είναι ο Miller-Rabin.

```python
def miller_rabin(n, k):
    # If number is even, it's a composite number

    if n == 2:
        return True

    if n % 2 == 0:
        return False

    r, s = 0, n - 1
    while s % 2 == 0:
        r += 1
        s //= 2
        for _ in range(k):
            a = random.randrange(2, n - 1)
            x = pow(a, s, n)
            if x == 1 or x == n - 1:
                continue
        for _ in range(r - 1):
            x = pow(x, 2, n)
            if x == n - 1:
                break
        else:
            return False
    return True
```

Άσκηση 11

Έστω $n \in \mathbb{N}^+, L_n = \{a \in \mathbb{Z}_n^+ : a^{n-1} = a^{t2^h} = 1, t : odd, \text{ if } a^{t2^{k+1}} = 1 \text{ then } a^{t2^k} = \pm 1, k = 0, 1, \ldots, h-1\}$.

Αρχικά, αν $n : prime$, εύκολα βλέπουμε από το μικρό θεώρημα Fermat ότι $a^n \equiv a \Rightarrow a^{n-1} \equiv 1 \pmod{n}$.
Επιπλέον, έχουμε $a^{t2^{k+1}} = a^{2t2^k} = \left(a^{t2^k}\right)^2$. Για κάποιον αριθμό x, είναι: $x^2 \equiv 1 \Rightarrow x^2 - 1 \equiv 0 \pmod{n} \Rightarrow n \mid (x-1)(x+1)$, οπότε είναι $x \equiv 1 \vee x \equiv -1 \pmod{n}$. Οπότε, αν όπου x θεωρήσουμε το ζητούμενο a^{t2^k}, βλέπουμε πράγματι ότι αν $a^{t2^{k+1}} = 1$, τότε $a^{t2^k} = 1 \vee a^{t2^k} = -1$, δηλαδή $a^{t2^k} = \pm 1$.
Από τα παραπάνω συμπεραίνουμε ότι και οι 2 προϋποθέσεις για να ανήκει κάποιο στοιχείο a του $\mathbb{Z}_n^+$ στο L_n είναι αληθείς για κάθε $a \in \mathbb{Z}_n^+$, οπότε τα σύνολα L_n και $\mathbb{Z}_n^+$ ταυτίζονται αν ο n είναι πρώτος.

Άσκηση 12

Από τον ορισμό $\mathbb{B} = \{a_1 \in \mathbb{G}_1 : (a_1, b_1) \in \mathbb{B} \text{ για κάποιο } b_1 \in \mathbb{G}_2\}$ έχουμε ότι $\mathbb{B}_1 \subseteq \mathbb{G}_1$. Αρκεί να δείξουμε ότι το $(\mathbb{B}_1, +_1)$ είναι ομάδα.

Έστω $a_i, a_j \in \mathbb{B}_1$. Από τον ορισμό του $\mathbb{B}_1$, υπάρχουν στοιχεία $b_i, b_j \in \mathbb{G}_2$ τέτοια ώστε $(a_i, b_i), (a_j, b_j) \in \mathbb{B}$
Γνωρίζουμε, όμως, ότι $\mathbb{B}$ είναι ομάδα και συνεπώς

$$(a_i, b_i) + (a_j, b_j) = (a_i +_1 a_j, b_i +_2 b_j)$$

Από την κλειστότητα, όμως, είναι:

$$\begin{cases} (a_i +_1 a_j) \in \mathbb{G}_1 \\ (b_i +_2 b_j) \in \mathbb{G}_2 \end{cases}$$

Οπότε από τον ορισμό του $\mathbb{B}_1$ έχουμε ότι $(a_i +_1 a_j) \in \mathbb{G}_1$, δηλαδή αποδείξαμε την κλειστότητα της $\mathbb{B}_1$

Ακόμα, αφού $\mathbb{B}_1 \subseteq \mathbb{G}_1$ και $\mathbb{G}_1$ είναι ομάδα, τότε ισχύει η προσεταιριστικότητα και για τη $\mathbb{B}_1$.

Έστω τα ουδέτερα στοιχεία $e_1 \in \mathbb{G}_1, e_2 \in \mathbb{G}_2$ και κάποια στοιχεία $a_1 \in \mathbb{G}_1, a_2 \in \mathbb{G}_2$. Επειδή $\mathbb{G}_1, \mathbb{G}_2$ ομάδες, είναι $a_1 +_1 e_1 = a_1, a_2 +_2 e_2 = a_2$. Ακόμα έχουμε ότι $(a_1, a_2) + (e_1, e_2) = (a_1 +_1 e_1, a_2 +_2 e_2) = (a_1, a_2)$, δηλαδή το στοιχείο (e_1, e_2) είναι το ουδέτερο στοιχείο της $\mathbb{G}_1 \times \mathbb{G}_2$. Από ορισμό, λοιπόν, αφού $e_2 \in \mathbb{G}_2$, τότε $e_1 \in \mathbb{B}_1$, άρα η $\mathbb{B}_1$ έχει ουδέτερο στοιχείο το e_1.

Τέλος, έστω το στοιχείο $a_1 \in \mathbb{B}_1$. Υπάρχει $b_1 \in \mathbb{G}_2$ ώστε $(a_1, b_1) \in \mathbb{B}$. Επιπλέον, έχουμε το ουδέτερο στοιχείο $(e_1, e_2) \in \mathbb{B}$ και για το αντίστροφο του (a_1, b_1), το (a_2, b_2), είναι:

$$(a_1, b_1) + (a_2, b_2) = (e_1, e_2) \Rightarrow \begin{cases} a_1 +_1 b_1 = e_1 \\ a_2 +_2 b_2 = e_2 \end{cases}$$

Συνεπώς, $a_2 = a_1^{-1}, b_2 = b_1^{-1}$ και άρα $(a_2, b_2) \in \mathbb{B} \Rightarrow (a_2^{-1}, b_2^{-1}) \in \mathbb{B}$. Όμως, $b_1^{-1} \in \mathbb{G}_2, a_1^{-1} \in \mathbb{G}_1$, τότε είναι και $a_1^{-1} \in \mathbb{B}_1$. Άρα η $\mathbb{B}_1$ έχει αντίστροφο στοιχείο.

Από τα παραπάνω καταλήγουμε ότι το $\mathbb{B}_1$ είναι υποομάδα της $\mathbb{G}_1$.

Άσκηση 13

Έστω $\mathbb{Z}_p^*$ με p πρώτο και g ένας γεννήτορας, p, g γνωστά.

Ερώτημα 1

Για το στοιχείο $g^{\frac{p-1}{d}}$ έχουμε ότι η τάξη του είναι το πολύ d (από το μικρό θεώρημα Fermat) επειδή

$$(g^{\frac{p-1}{d}})^d = g^{d\cdot\frac{p-1}{d}} = g^{p-1} \equiv 1 \pmod p$$

Έστω ότι η τάξη είναι $k < d$. Τότε

$$(g^{\frac{p-1}{d}})^k = g^{k\cdot\frac{p-1}{d}} = e$$

Άρα ο γεννήτωρ g θα είχε τάξη $(p-1)k/d < p-1$ το οποίο είναι άτοπο (επειδή δε θα ήταν γεννήτωρ σε αυτήν την περίπτωση). Συνεπώς η τάξη είναι πράγματι d.
Άρα, βάσει των παραπάνω, το ζητούμενο στοιχείο $b \in \mathbb{Z}_p^*$ είναι το $g^{\frac{p-1}{d}} \pmod p$

Ερώτημα 2

Από το θεμελιώδες θεώρημα κυκλικών υποομάδων, η $\mathbb{Z}_p^*$ περιέχει ακριβώς μια κυκλική υποομάδα τάξης d την οποία συμβολίζουμε H_d. Το πλήθος των στοιχείων τάξης d στη $\mathbb{Z}_p^*$ είναι ίδιο με το πλήθος των γεννητόρων της H_d. Άρα για να απαντήσουμε στο ερώτημα πόσα στοιχεία τάξης d υπάρχουν μέσα στο $\mathbb{Z}_p^*$, αρκεί να δείξουμε πόσους γεννήτορες έχει η H_d.

Αρχικά θα δείξουμε ότι αν $\mathbb{G}$ κυκλική ομάδα, $g \in \mathbb{G}$ γεννήτωρ, $|\mathbb{G}| = n$ και $k \in \mathbb{N}$. Τότε

$$|g^k| = \frac{n}{\gcd(n,k)}$$

Έχουμε

$$(g^{\gcd(n,k)})^{\frac{n}{\gcd(n,k)}} = g^n = e \Rightarrow g^{\gcd(n,k)} \leq \frac{n}{\gcd(n,k)}$$

Αν ίσχυε η ανισότητα, θα είχαμε

$$(g^{\gcd(n,k)})^{\frac{n}{\gcd(n,k)}} = g^n = e$$

(?????)

Έχουμε λοιπόν για την H_d ότι $\forall k \in \{1,2,\dots d\} : g^k = \frac{n}{\gcd(d,k)}$. Όμως, g^k γεννήτωρ αν και μόνον αν $\gcd(d,k) = 1$ ώστε $g^k = d$. Οπότε οι γεννήτορες είναι $\phi(d)$ το πλήθος, άρα και τα στοιχεία τάξης d στο $\mathbb{Z}_p^*$ είναι $\phi(d)$ το πλήθος.

Ερώτημα 3

Από το ερώτημα 1, $b = g^d$ και όπως είπαμε στο ερώτημα 2, παράγει τη μοναδική κυκλική υποομάδα τάξης d, την οποία συμβολίσαμε H_d. Όμως, όλα τα στοιχεία τάξης d είναι γεννήτορες και παράγουν την H_d, άρα έχουμε $\phi(d)$ γεννήτορες, όσα και τα στοιχεία τάξης d.

Ερώτημα 4

Από θεμελιώδες θεώρημα κυκλικών υποομάδων, υπάρχει μόνο μία κυκλική υποομάδα τάξης d στο $\mathbb{Z}_p^*$, η $\langle g^{(p-1)/d}\rangle$. Η απόδειξη: Όταν $d \mid p-1$, έχουμε:

$$|\langle g^{(p-1)/d}\rangle| = |g^{(p-1)/d}| = \frac{p-1}{\gcd(p-1, \frac{p-1}{d})} = \frac{p-1}{\frac{p-1}{d}} = d$$

Άρα, πράγματι η $\langle g^{(p-1)/d}\rangle$ είναι κυκλική υποομάδα τάξης d.
Έστω τώρα $H, H \subseteq G, |H| = d, d \mid p-1$. Οπότε η H είναι υποομάδα της G και άρα υπάρχει $m \mid p-1$ ώστε $H = \langle g^m\rangle$. Οπότε:

$$|H| = d \Leftrightarrow |\langle g^m\rangle| = |g^m| = \frac{p-1}{\gcd(p-1,m)} = \frac{p-1}{m} = d \Rightarrow m = \frac{p-1}{d}$$

Επομένως, $H = \langle g^m\rangle = \langle g^{\frac{p-1}{d}}\rangle$.

Ερώτημα 5

Το στοιχείο h παράγει κυκλική υποομάδα τάξης d, την H_d. Για το h έχουμε ότι $\forall k \in \{1, 2, \ldots, d\} : (h^d)^k = (h^k)^d = e^k = e$. Οπότε κάθε h^k είναι στοιχείο της H_d. Μπορούμε σε τετραγωνικό χρόνο να ελέγξουμε αν κάποιο $a = h^k$ ανήκει στην υποομάδα που παράγει το h αν υπάρχει $k \in \{1, 2, \ldots, d\}$ ώστε ??????.

Άσκηση 14

Υλοποιούμε τον έλεγχο πρώτων αριθμών Miller-Rabin σε Python. Ο αλγόριθμος ελέγχει επανειλημμένα αν οι τυχαίοι αριθμοί (witnesses) δείχνουν ότι ο δεδομένος αριθμός δεν είναι πρώτος. Εάν ένας αριθμός περάσει το test πολλές φορές, θεωρείται πιθανότατα πρώτος.

```
import random

# Function to perform the Miller-Rabin primality test
def is_prime_miller_rabin(n, k=5):
    # Check for small values first
    if n <= 1:
        return False  # Numbers less than or equal to 1 are not prime
    if n <= 3:
        return True  # Numbers 2 and 3 are prime
    if n % 2 == 0:
        return False  # Even numbers (except 2) are not prime

    # Perform the Miller-Rabin test using "k" rounds
    for _ in range(k):
        a = random.randint(2, n - 2)  # Choose a random base "a"
        if not miller_rabin(n, a):  # Check if "a" is a witness for compositeness
            return False

    return True  # If "n" passes all rounds, it is probably prime

# Function to perform the Miller-Rabin primality test for a specific base "a"
def miller_rabin(n, k = 5):
    if n == 2:
        return True  # 2 is a prime number

    if n % 2 == 0:
        return False  # Even numbers (except 2) are not prime

    r, s = 0, n - 1
    while s % 2 == 0:
        r += 1
        s //= 2

    for _ in range(k):
        a = random.randrange(2, n - 1)  # Choose a random base "a" in the range [2, n-2]
        x = pow(a, s, n)  # Compute a^s mod n
        if x == 1 or x == n - 1:
            continue

        for _ in range(r - 1):
            x = pow(x, 2, n)
            if x == n - 1:
                break
        else:
            return False

    return True  # If "a" is not a witness for compositeness in all rounds, "n" is
        probably prime

# Numbers to check for primality
numbers_to_check = [
    67280421310721,
    170141183460469231731687303715884105 7,
```

```
    2**1001 - 1,
    2**2281 - 1,
    2**9941 - 1,
    2**19939 - 1
]

numsToPrint = [
    "67280421310721",
    "170141183460469231731687303715884105 7",
    "2**1001 - 1",
    "2**2281 - 1",
    "2**9941 - 1",
    "2**19939 - 1"
]

# Apply the Miller-Rabin test to each number and print the result
for i in range(len(numbers_to_check)):
    is_prime = miller_rabin(numbers_to_check[i])
    if is_prime:
        print(f"{numsToPrint[i]} is prime.")
    else:
        print(f"{numsToPrint[i]} is not prime.")
```

Τα αποτελέσματα φαίνονται εδώ:

```
67280421310721 is prime.
170141183460469231731687303715884105 7 is not prime.
2**1001 - 1 is not prime.
2**2281 - 1 is prime.
2**9941 - 1 is prime.
2**19939 - 1 is not prime.
```

References

[1] Practical Cryptography. Cryptanalysis of Simple Substitution Cipher. Retrieved from http://practicalcryptography.com/cryptanalysis/stochastic-searching/cryptanalysis-simple-substitution-cipher/