



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»

Навчально-науковий Фізико-технічний інститут
Кафедра інформаційної безпеки

КРИПТОГРАФІЯ

Комп'ютерний практикум №3
Криптоаналіз афінної біграмної підстановки

Виконали:
Студенти ФБ-33
Дохоян Юлія
Терещенко Микола

Київ – 2025

Мета роботи: Набуття навичок частотного аналізу на прикладі розкриття монографічної підстановки; опанування прийомами роботи в модулярній арифметиці.

Порядок виконання роботи:

0. Уважно прочитати методичні вказівки до виконання комп'ютерного практикуму.
1. Реалізувати підпрограми із необхідними математичними операціями: обчисленням оберненого елементу за модулем із використанням розширеного алгоритму Евкліда, розв'язуванням лінійних порівнянь. При розв'язуванні порівнянь потрібно коректно обробляти випадок із декількома розв'язками, повертуючи їх усі.
2. За допомогою програми обчислення частот біграм, яка написана в ході виконання комп'ютерного практикуму №1, знайти 5 найчастіших біграм запропонованого шифртексту (за варіантом).
3. Перебрати можливі варіанти співставлення частих біграм мови та частих біграм шифртексту (розглядаючи пари біграм із п'яти найчастіших). Для кожного співставлення знайти можливі кандидати на ключ (a,b) шляхом розв'язання системи (1).
4. Для кожного кандидата на ключ дешифрувати шифртекст. Якщо шифртекст не є змістовним текстом російською мовою, відкинути цього кандидата.
5. Повторювати дії 3-4 доти, доки дешифрований текст не буде змістовним.

Хід роботи

Всі завдання було реалізовано в одному скрипті, що прикріплений разом із протоколом. Далі показано фрагменти коду, що відповідають умовам у завданні:

1. Реалізувати підпрограми із необхідними математичними операціями: обчисленням оберненого елементу за модулем із використанням розширеного алгоритму Евкліда, розв'язуванням лінійних порівнянь. При розв'язуванні порівнянь потрібно коректно обробляти випадок із декількома розв'язками, повертаючи їх усі.

Розширений алгоритм Евкліда:

```
def extended_gcd(a, b):  
    if b == 0:  
        return a, 1, 0  
    g, x1, y1 = extended_gcd(b, a % b)  
    return g, y1, x1 - (a // b) * y1
```

Рис.1

Обчислення оберненого елемента за модулем:

```
def mod_inverse(a, m):  
    g, x, _ = extended_gcd(a, m)  
    if g != 1:  
        return None  
    return x % m
```

Рис.2

Розв'язування лінійних порівнянь:

```
def solve_for_a_candidates(x1, x2, y1, y2):  
    A = (x1 - x2) % M  
    B = (y1 - y2) % M  
    g = math.gcd(A, M)  
    if B % g != 0:  
        return []  
    inv = mod_inverse(A, M)  
    if inv is None:  
        return []  
    a0 = (inv * B) % M  
    return [(a0 + k*M) % M for k in range(g)]
```

Рис.3

2. За допомогою програми обчислення частот біграм, яка написана в ході виконання комп'ютерного практикуму №1, знайти 5 найчастіших біграм запропонованого шифртексту (за варіантом).

```
def top_bigrams(text, n=5):
    clean_text = re.sub(f'^{let}', '', text.lower())
    bigrams = [clean_text[i:i+2] for i in range(0, len(clean_text)-1, 2) if
len(clean_text[i:i+2])==2]
    counter = Counter(bigrams)
    top = counter.most_common(n)
    return [bg for bg, _ in top]
```

Рис.4

3. Перебрати можливі варіанти співставлення частих біграм мови та частих біграм шифртексту (розглядаючи пари біграм із п'яти найчастіших). Для кожного співставлення знайти можливі кандидати на ключ (a,b) шляхом розв'язання системи (1).

```
def build_key_candidates(cipher_top, lang_top):
    candidates = set()
    for i in range(len(cipher_top)):
        for j in range(len(cipher_top)):
            if i == j:
                continue
            for p in range(len(lang_top)):
                for q in range(len(lang_top)):
                    if p == q:
                        continue
                    y1 = bigram_to_num(cipher_top[i])
                    y2 = bigram_to_num(cipher_top[j])
                    x1 = bigram_to_num(lang_top[p])
                    x2 = bigram_to_num(lang_top[q])
                    a_list = solve_for_a_candidates(x1, x2, y1, y2)
                    for a in a_list:
                        if math.gcd(a, M) != 1:
                            continue
                        b = (y1 - a * x1) % M
                        candidates.add((a, b))
    return sorted(candidates)
```

Рис.5

4. Для кожного кандидата на ключ дешифрувати шифртекст. Якщо шифртекст не є змістовним текстом російською мовою, відкинути цього кандидата.

```
def decrypt(text, a, b):
    a_inv = mod_inverse(a, M)
    if a_inv is None:
        return ""
    plaintext = ""
    clean_text = re.sub(f'[{let}]', ' ', text.lower())
    for i in range(0, len(clean_text)-1, 2):
        bg = clean_text[i:i+2]
        if len(bg) != 2:
            continue
        y = bigram_to_num(bg)
        x = (a_inv * (y - b)) % M
        plaintext += num_to_bigram(x)
    return plaintext
```

Рис.6

5. Повторювати дії 3-4 доти, доки дешифрований текст не буде змістовним.

```
with open(OUTPUT_FILE, 'w', encoding="utf-8") as f_out:
    for idx, (a, b) in enumerate(candidates, start=1):
        dec = decrypt(text, a, b)
        if has_incorrectbig(dec):
            continue
        valid_count += 1
        ic = index_of_coincidence(dec)
        print(f"#{valid_count} a={a}, b={b} IC={ic:.6f}")
        print("Початок розшифрованого тексту:")
        print(dec[:200])
        print("-"*60)
        f_out.write(f"#{valid_count} a={a}, b={b} IC={ic:.6f}\n")
        f_out.write(dec + "\n" + "-"*60 + "\n")
```

Рис.7

Результат виконання скрипту:

```
(kali㉿kali)-[~/Desktop/crl3]
$ python crypto3fin.py
Довжина тексту: 4958
Топ-5 біграм шифртексту: ['їа', 'юа', 'чш', 'юд', 'рщ']
Знайдено 158 кандидатів ключів.

#1 a=27, b=211 IC=0.056115
Початок розшифрованого тексту:
однакоэтакартинаскакойбысторонымыеенирассматривалирасплываєтьсявнечтоонеопределеноеприпадкипроявляючиесярезк
осприкусываниемусиливаючиесядоапасногодляжизниприводящегоктажкомусамокалечениюмогутвсежевнеко

Всього валідних кандидатів: 1
```

Рис.8

Заданий шифротекст:

Отриманий відкритий текст і ключ:

#1 a=27, b=211 IC=0.052836

Висновок:

У ході лабораторної роботи реалізовано криптоаналіз афінного біграмного шифру. Створено підпрограми для обчислення обернених елементів за модулем та розв'язання лінійних порівнянь із використанням розширеного алгоритму Евкліда. Застосовано частотний аналіз біграм для визначення можливих ключів (a, b) і виконано автоматичне дешифрування.

Автоматичний розпізнавач російської мови дозволив відсіяти некоректні результати. Отримані результати підтвердили ефективність поєднання математичних та лінгвістичних методів у криптоаналізі.