МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

ЛАБОРАТОРНА РОБОТА №3

З дисципліни «Криптографія»

Варіант 1

Виконали:
студенти 3 курсу ФТІ
групи ФБ-93
Абдуллаєва Есміра
Шовак Мирослав

Викладач:
Селюх П. В.

м.Київ – 2021

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

Завдання:

1. Уважно прочитати методичні вказівки до виконання комп'ютерного практикуму.
2. 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.
3. За допомогою програми обчислення частот біграм, яка написана в ході виконання комп'ютерного практикуму No1, знайти 5 найчастіших біграм запропонованого шифртексту (за варіантом).
4. 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).
5. Для кожного кандидата на ключ дешифрувати шифртекст. Якщо шифртекст не є змістовним текстом російською мовою, відкинути цього кандидата.
6. Повторювати дії 3-4 доти, доки дешифрований текст не буде змістовним.

Хід роботи

Першим кроком у нашій лабораторній роботі було те , що ми розбили текст на біграми без перетину. Після цього ми підрахували частоту кожної біграми і виділили 5 найчастіших. Наступним кроком було те , що для розв'язання системи лінійних рівнянь і в подальшому знаходженні ключів, нам потрібно було перебрати 25 комбінацій біграм мови і біграм зашифрованого тексту . Ми вирішили перебрати вручну і в залежності від ключів дивитися на результат розшифрування. Змістовність тексту ми визначали теж вручну , аналізатор змістовного тексту вирішили не реалізовувати. В результаті ми знайшли нашу пару біграм і відповідно наш ключ , але розшифрований текст був дещо з помилками. Так як у методичці було написано , що можливо букву 'ь' змінювали на 'ы' , ми вирішили їх поміняти у нашому масиві літер, після чого результат розшифрування став правильним.

Код програми

```
file1 =
open("/Users/esmira.23/Desktop/KPI/3курс/Кр
ипта/1.txt", "r").read()
file2 =
open("/Users/esmira.23/Desktop/KPI/3курс/Кр
ипта/2.txt", "w")

alphabet =
['а','б','в','г','д','е','ж','з','и','й','к
','л','м','н','о','п','р','с','т','у','ф','
х','ц','ч','ш','щ','ь','ы','э','ю','я']

#bigram1 = [['с','т'], ['н','о'], ['т',
'о'], ['н','а'], ['е','н']] # вихідный
текст
#bigram2 = [['р','н'], ['ы','ч'],
['н','к'], ['ц','э'], ['и','а']] # вхідний
текст
bigram1 = [['с','т'], ['е','н']]
bigram2 = [['р','н'], ['н','к']]


def euclid ext(a, n):
    if n == 0:
        return a, 1, 0
    else:
        d, x, y = euclid_ext(n, a % n)
        return d, y, x - y * (a // n)

def reverse(a, n):
    gcd, x, y = euclid ext(a, n)
    if gcd == 1:
        return (x % n + n) % n
    else:
        return -1

def euclid(a, y, n):
    gcd, y1, x1 = euclid ext(a, n)
    if gcd == 1:          # знаходимо
обернений
        x = reverse(a,n)
        return x
    elif y % gcd != 0: # немає розв`язкі
        return False
    else:
        euclid(a/gcd, y/gcd, n/gcd)


def max_bigram(text):
    mass = []
    mass1 = []
    line = [text[k:k + 2] for k in range(0,
len(text), 2)]
    new line = set(line)
    for i in new line:
        number = line.count(i)
        mass.append([i,number])
    sorted bigrams = sorted(mass,
key=lambda x: x[1])
    for i in range(5):
        mass1.append(sorted bigrams[-
(i+1)])
        mass.clear()
    for i in range(len(mass1)):
        mass.append(mass1[i][0])
    print(mass)

def index(i):
    X1 = alphabet.index(bigram1[i][0]) * 31
+ alphabet.index(bigram1[i][1])
    Y1 = alphabet.index(bigram2[i][0]) * 31
+ alphabet.index(bigram2[i][1])
    X2 = alphabet.index(bigram1[i + 1][0])
* 31 + alphabet.index(bigram1[i + 1][1])
    Y2 = alphabet.index(bigram2[i + 1][0])
* 31 + alphabet.index(bigram2[i + 1][1])
    return X1, Y1, X2, Y2

def find key():
    mass = []
    for i in range(len(bigram2)-1):
        X1, Y1, X2, Y2 = index(i)
        a = (euclid(X1-X2, Y1-Y2, 31 ** 2)
* (Y1-Y2)) % (31 ** 2)
        b = (Y1 - a * X1) % (31 ** 2)
        mass.append([a, b])
    return (mass)

def decrypt(text):
    arr = []
    arr1 = []
    mass = find key()
    line = [text[k:k + 2] for k in range(0,
len(text), 2)]
    for i in range(len(line)):
        A = mass[0][0]
        B = mass[0][1]
        Y = alphabet.index(line[i][0])*31 +
alphabet.index(line[i][1])
        X = (reverse(A, 31**2) * (Y - B)) %
31**2
        arr.append(X)
    for i in range(len(arr)):
        letter = alphabet[arr[i] // 31] +
alphabet[arr[i] % 31]
        arr1.append(letter)
    answer = ''.join(arr1)
    return (answer)

#main
file2.write(decrypt(file1))
```

Результат роботи

```
1. [['рн', 62], ['ыч', 41], ['нк', 34], ['цз', 32], ['иа', 30]]
2. ['рн', 'ыч', 'нк', 'цз', 'иа']
3. X1: 545 Y1: 509
   X2: 168 Y2: 413
4. A: 13    B: 151
```

1. Знайшли у тексті 5 найчастіших біграм та вивели їх кількість.
2. Залишили лише самі біграми.
3. Знайдені X та Y, які підійшли нам для знаходження ключів A та B.
4. Ключі A та B, за допомогою яких було розшифровано текст.

Біграми, які допомогли знайти ключі:

```
Bigram plaintext:  ['с', 'т'] ['е', 'н']
Bigram ciphertext:  ['р', 'н'] ['н', 'к']
```

Спроби розшифрування з неправильно підібраними ключами:

```
A: 230 B: 89
юнгггґйаянщюоийнесэьяостхешсеоюозочныpцсщматуиьаэьачетйрбхстырангаопясатдлфкххфепротюкххкхкаырлитдлыэхигаюкхкррэшригагаьжтрщзрбйаэьбяя
A: 277 B: 534
уожсжсыхэоогьащомуизтчсэщдэукчйчэчыогккушеуэйащхизвлвзыкяшсэгкжодхфбфууэяыютзтрдукхэстзтзтщроыюзяыкщыадхстзтэкыфнадхдхюнпккцтяыхиземж
A: 877 B: 906
щхакакирчхифсагхенцьжуьлчжюнщуиухужхйпвнучюлзаерцьжбулопзеьлйпххфрхсснюлрщбыяыщжвпблоыяыяыяшчщнлрщхцгафроыяыапщянафрфрыдыпмвиоирцьитг
```