НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
Навчально-науковий фізико-технічний інститут
КАФЕДРА ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

«Криптографія»

Комп’ютерний практикум №2

Студенти: Маврикін Едуард
Слобода Ірина
Група: ФБ-25
Варіант 2

Київ – 2025

Мета роботи:

Засвоєння методів частотного криптоаналізу. Здобуття навичок роботи та аналізу потокових шифрів гамування адитивного типу на прикладі шифру Віженера.

1. Самостійно підібрати текст для шифрування (2-3 кб) та ключі довжини r = 2, 3, 4, 5, а також довжини 10-20 знаків. Зашифрувати обраний відкритий текст шифром Віженера з цими ключами.

Підібрано текст розміром 3132 символи (після очищення від пробілів та пунктуації)

Текст на російській мові, алфавіт m=32 символи

Ключі для шифрування:

Короткі ключі (r=2-5):

- - r=2: "ау"
- - r=3: "дно"
- - r=4: "куда"
- - r=5: "можно"

Довгі ключі (r=10-20):

- - r=10: "достижение"
- - r=11: "воображение"
- - r=12: "многообразие"
- - r=13: "автоматизация"
- - r=14: "взаимодействие"
- - r=15: "непосредственно"
- - r=16: "экспериментально"
- - r=17: "предусмотрительно"
- - r=18: "производительность"
- - r=19: "электрооборудование"
- - r=20: "антирадиолокационный"

```
(base) PS C:\Users\edmav\Desktop\kpi\crypto\crypto25-26\lab2\Mavrykin_FB_25_Sloboda_FB-25_cp2> python .\task1.py
Text length: 3132

ау (r=2) -> cipher-2.txt
дно (r=3) -> cipher-3.txt
куда (r=4) -> cipher-4.txt
можно (r=5) -> cipher-5.txt
достижение (r=10) -> cipher-10.txt
воображение (r=11) -> cipher-11.txt
многообразие (r=12) -> cipher-12.txt
автоматизация (r=13) -> cipher-13.txt
взаимодействие (r=14) -> cipher-14.txt
непосредственно (r=15) -> cipher-15.txt
экспериментально (r=16) -> cipher-16.txt
предусмотрительно (r=17) -> cipher-17.txt
производительность (r=18) -> cipher-18.txt
электрооборудование (r=19) -> cipher-19.txt
антирадиолокационный (r=20) -> cipher-20.txt
(base) PS C:\Users\edmav\Desktop\kpi\crypto\crypto25-26\lab2\Mavrykin_FB_25_Sloboda_FB-25_cp2>
```

node | sample.txt U | lab2.py U

crypto25-26 > lab2 > Mavrykin_FB_25_Sloboda_FB-25_cp2 > sample.txt

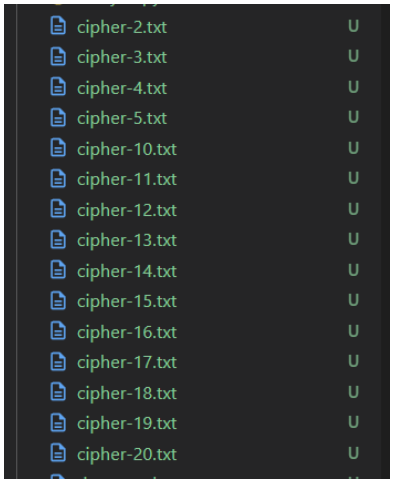
```
1  В тот день погода была переменчивой. Утром светило яркое солнце, но к полудню небо затянуло тучами. Ветер усилился и начал срывать последние листья с дерев
2
3  Город постепенно менял свой облик. Летние краски уступали место серым и коричневым тонам. Люди на улицах ходили быстрее, стараясь поскорее добраться до теп
4
5  Студенты возвращались в университеты после длинных каникул. Библиотеки наполнялись читателями, аудитории оживали от студенческих голосов. Преподаватели гот
6
7  В парках становилось тише. Птицы готовились к отлету на юг. Белки запасали орехи на зиму. Деревья сбрасывали листву, готовясь к зимнему сну. Природа демонс
8
9  Технологии продолжали развиваться стремительными темпами. Новые изобретения появлялись каждый день. Ученые работали над проектами, которые могли изменить б
10
11 Криптография играла важную роль в современном цифровом мире. Защита информации становилась критически важной задачей. Специалисты разрабатывали новые алгор
```

```python
from vigenere import encrypt, alh

keys = [
    'ау',
    'дно',
    'куда',
    'можно',
    'достижение',
    'воображение',
    'многообразие',
    'автоматизация',
    'взаимодействие',
    'непосредственно',
    'экспериментально',
    'предусмотрительно',
    'производительность',
    'электрооборудование',
    'антирадиолокационный'
]

with open('sample.txt', 'r', encoding='utf8') as f:
    text = f.read().lower()
    text = ''.join(c for c in text if c in alh)

with open('sample_cleaned.txt', 'w', encoding='utf8') as f:
    f.write(text)

print(f'Text length: {len(text)}')
print()

for key in keys:
    cipher = encrypt(text, key)
    filename = f'cipher-{len(key)}.txt'
    with open(filename, 'w', encoding='utf8') as f:
        f.write(cipher)
    print(f'{key} (r={len(key)}) -> {filename}')

```

Алгоритм шифрування:

$y[i] = (x[i] + k[i \bmod r]) \bmod 32$

Створено 15 файлів шифртекстів (cipher-2.txt ... cipher-20.txt)



2. Підрахувати індекси відповідності для відкритого тексту та всіх одержаних шифртекстів і порівняти їх значення

Індекс відповідності (IC) обчислюється за формулою:

$IC = (1/(n(n-1))) * \Sigma N_t(N_t - 1)$

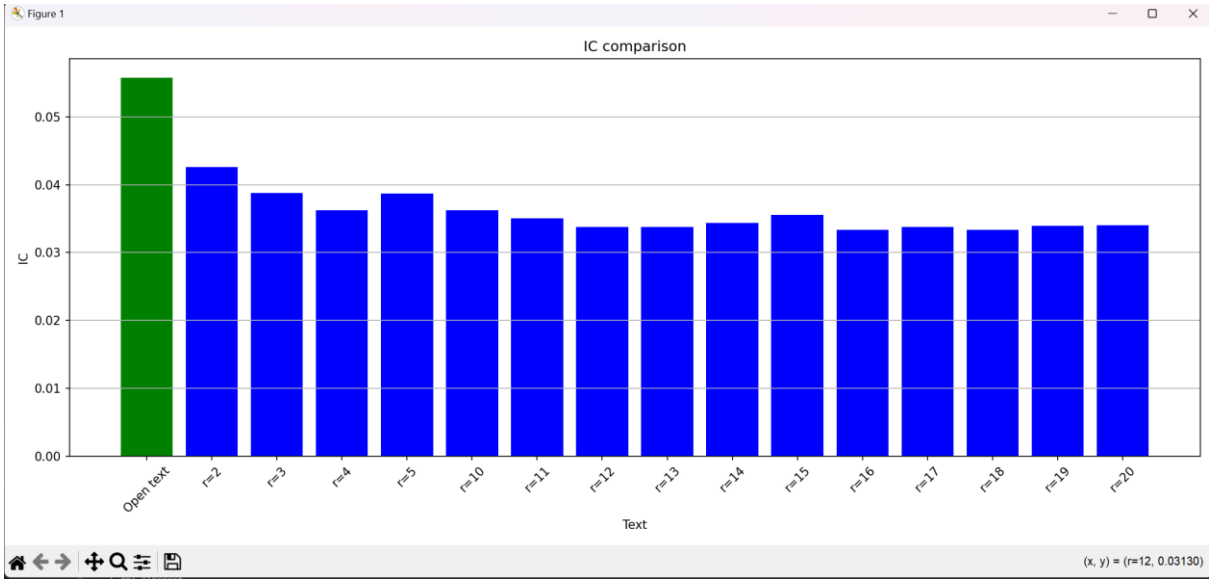
де N_t - кількість появ символу t у тексті.

node | task2.py A | lab2.py U

crypto25-26 > lab2 > Mavrykin_FB_25_Sloboda_FB-25_cp2 > task2.py > ...

```python
from analysis import ic
import matplotlib.pyplot as plt

with open('sample_cleaned.txt', 'r', encoding='utf8') as f:
    open_text = f.read()

open_ic = ic(open_text)
print(f'Open text IC: {open_ic:.6f}')
print()

results = {'Open text': open_ic}

key_lengths = [2, 3, 4, 5, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]

for r in key_lengths:
    filename = f'cipher-{r}.txt'
    try:
        with open(filename, 'r', encoding='utf8') as f:
            cipher = f.read()
        cipher_ic = ic(cipher)
        results[f'r={r}'] = cipher_ic
        print(f'r={r}: {cipher_ic:.6f}')
    except:
        pass

plt.figure(figsize=(14, 6))
labels = list(results.keys())
values = list(results.values())
colors = ['green'] + ['blue'] * (len(values) - 1)
plt.bar(labels, values, color=colors)
plt.xlabel('Text')
plt.ylabel('IC')
plt.title('IC comparison')
plt.xticks(rotation=45)
plt.grid(axis='y')
plt.tight_layout()
plt.show()
```

Результати:



```
(base) PS C:\Users\edmav\Desktop\kpi\crypto\crypto25-26\lab2\Mavrykin_FB_25_Sloboda_FB-25_cp2> python .\task2.py
Open text IC: 0.055771

r=2: 0.042620
r=3: 0.038777
r=4: 0.036242
r=5: 0.038713
r=10: 0.036226
r=11: 0.035065
r=12: 0.033790
r=13: 0.033729
r=14: 0.034356
r=15: 0.035525
r=16: 0.033344
r=17: 0.033786
r=18: 0.033331
r=19: 0.033897
r=20: 0.034021
```

1. IC відкритого тексту (0.0558) значно вище за теоретичне значення рівноімовірного алфавіту $I_0=1/32\approx0.0313$

2. Зі збільшенням довжини ключа IC падає і наближається до I_0

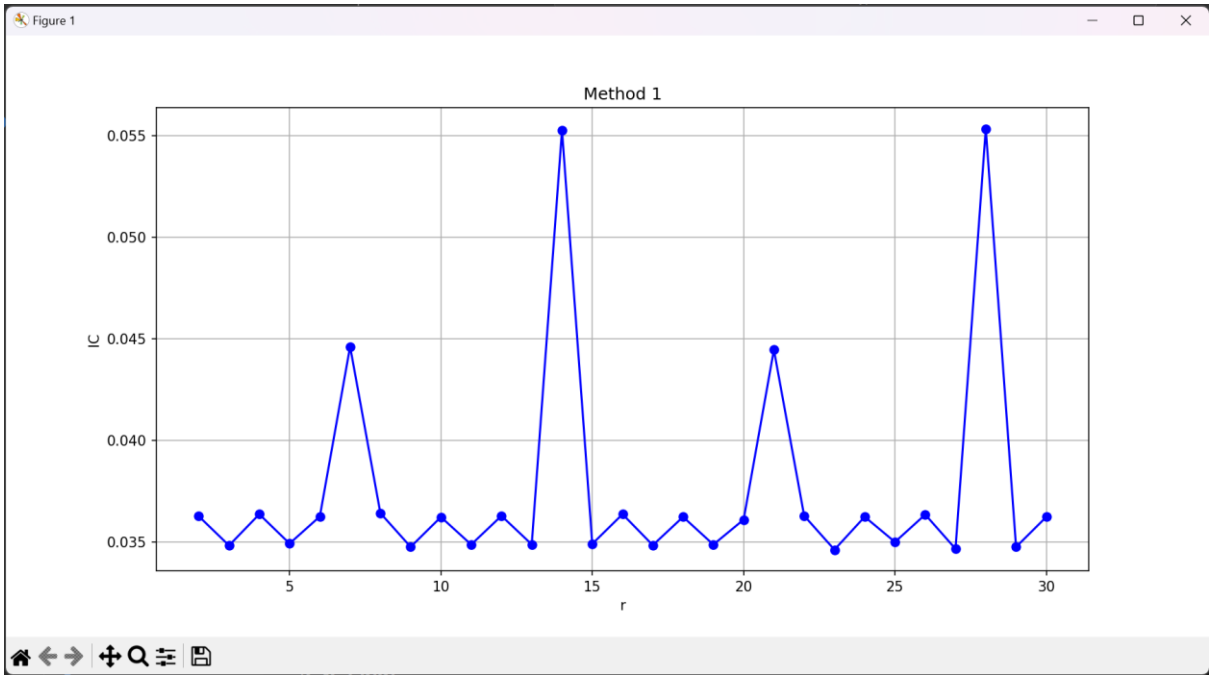
3. При $r\geq10$ IC стабілізується на рівні ~0.033-0.034

4. Це підтверджує, що шифр Віженера з довгим ключем добре маскує статистичні властивості мови

3. Використовуючи наведені теоретичні відомості, розшифрувати наданий шифртекст (згідно свого номеру варіанта).

Крок 1. Визначення періоду ключа (Method 1)

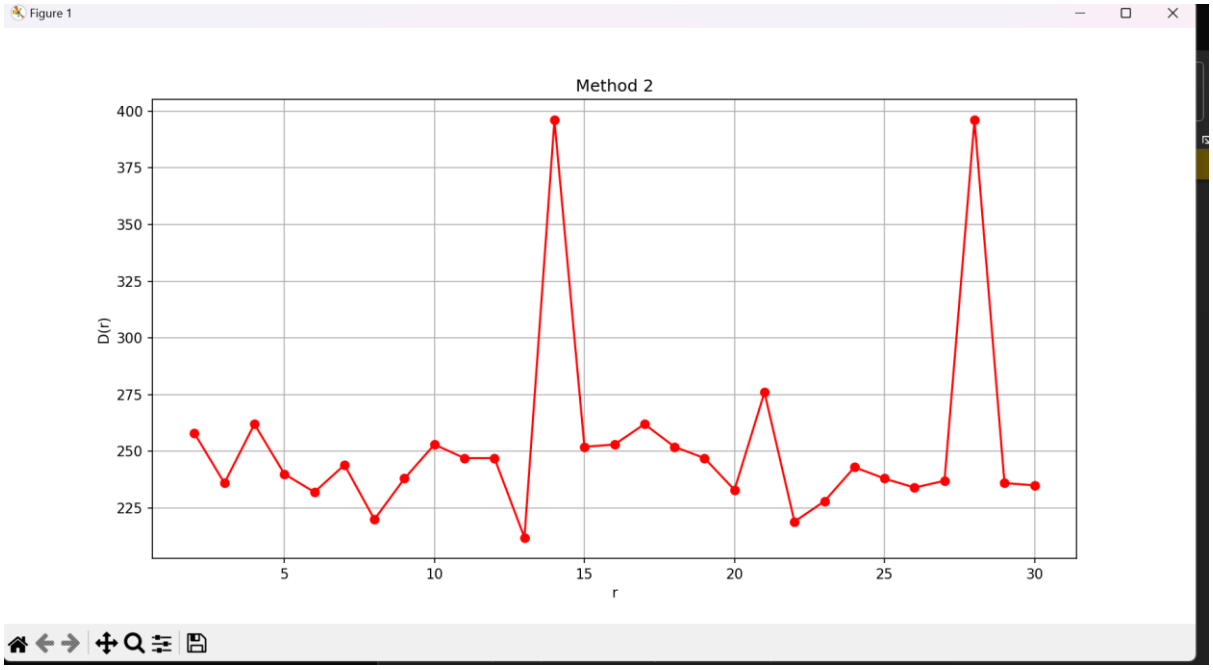
Розбиття тексту на блоки для різних r і обчислення середнього IC:



```
(base) PS C:\Users\edmav\Desktop\kpi\crypto\crypto25-26\lab2\Mavrykin_FB_25_Sloboda_FB-25_cp2> python .\main.py
Method 1:
r: 2, IC: 0.036268
r: 3, IC: 0.034828
r: 4, IC: 0.036368
r: 5, IC: 0.034923
r: 6, IC: 0.036257
r: 7, IC: 0.044626
r: 8, IC: 0.036423
r: 9, IC: 0.034758
r: 10, IC: 0.036230
r: 11, IC: 0.034873
r: 12, IC: 0.036287
r: 13, IC: 0.034881
r: 14, IC: 0.055282
r: 15, IC: 0.034905
r: 16, IC: 0.036370
r: 17, IC: 0.034842
r: 18, IC: 0.036236
r: 19, IC: 0.034872
r: 20, IC: 0.036100
r: 21, IC: 0.044470
r: 22, IC: 0.036292
r: 23, IC: 0.034620
r: 24, IC: 0.036242
r: 25, IC: 0.035000
r: 26, IC: 0.036350
r: 27, IC: 0.034663
r: 28, IC: 0.055361
r: 29, IC: 0.034754
r: 30, IC: 0.036236
```

Висновок: Виразні піки на r=7, 14, 21, 28. Це кратні 7 і 14. Ймовірний період: r=7 або r=14

Крок 2. Перевірка методом D(r) (Method 2)



Підрахунок кількості співпадінь символів на відстані r:

- D(r) показує кількість однакових символів на відстані r

- Для істинного періоду D(r) буде максимальним

Результат: Максимуми D(r) також на r=7, 14, 21, 28

Період найімовірніше r=14 (найбільший пік IC=0.0553)

Крок 3. Пошук ключа

Метод: Для кожного блоку Y_0, Y_1, ..., Y_{13} знайти найчастішу букву і припустити що вона відповідає "о" (найчастіша буква російської мови).

Автоматично знайдений ключ: жосвеыдиадозор

Розшифрування: "уакисхчгжтосдефатес..." (не читається)

Крок 4. Коригування ключа

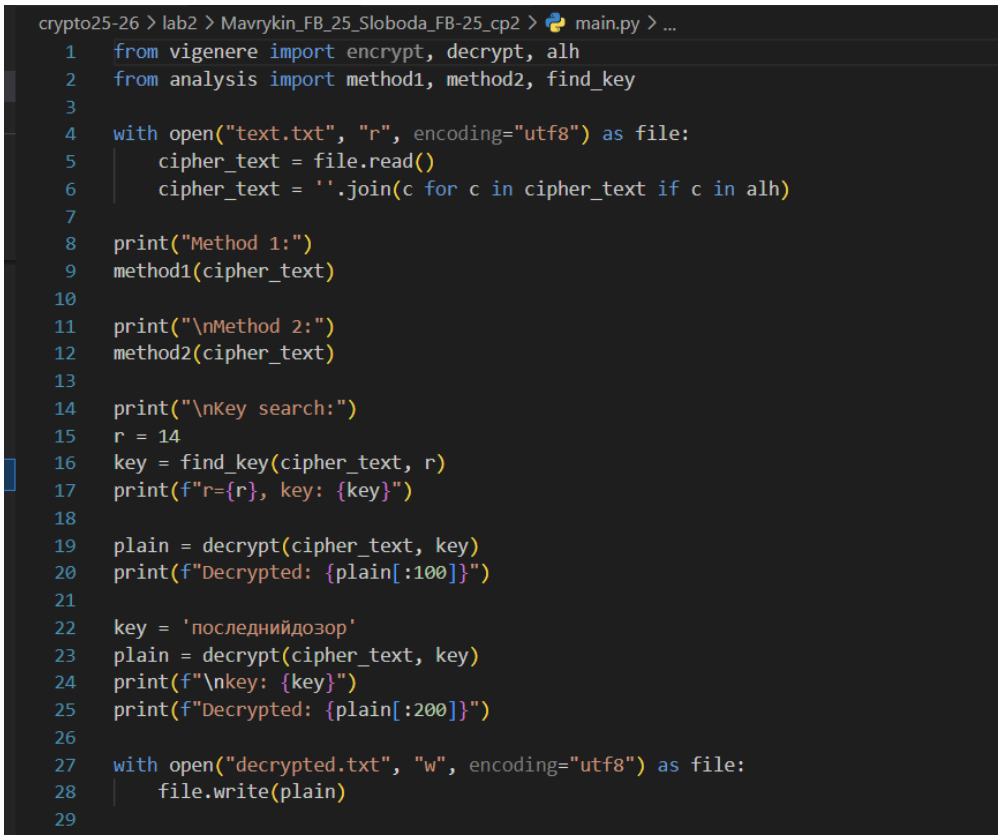
Аналізуючи структуру ключа "ос_е_ыдиадозо", можна припустити:

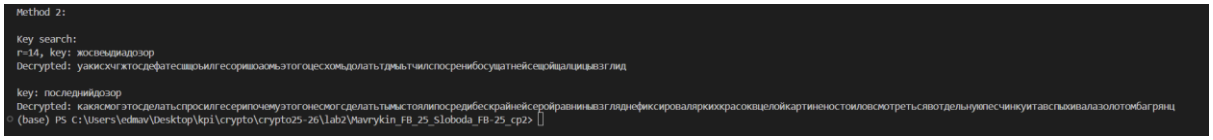
- "ос_е" схоже на початок слова

- "дозор" в кінці виглядає логічно

- Можливо слово: "последний" + "дозор"

Перевірка ключа: последнийдозор





Розшифрування:

какясмогэтосделатьспросилгесерипочемуэтогонесмогсделатьтымыстоялипосредибескрайнейсеройравнинывзгляднефиксироваларкихкрасоквцелойкартиненостоиловсмотретьсявотдельнуюпесчинкуитавспыхивалазолотомбагрянцем...

Це відрвок з книги Сергія Лук'яненка "Дозори: Останній Дозор"

1. Ефективність шифру Віженера: При довжині ключа r≥10 ІС наближається до рівноімовірного розподілу, що ускладнює криптоаналіз

2. Метод визначення періоду: Обидва методи (середній ІС блоків та D(r)) ефективно виявляють період ключа при достатньому обсязі шифртексту

3. Обмеження автоматичного пошуку: Припущення "найчастіша буква = о" не завжди справджується для окремих блоків. Потрібна ручна корекція або перебір варіантів

4. Практичне значення: Шифр Віженера з коротким ключем вразливий до частотного аналізу. Для надійності потрібен ключ довжиною близькою до довжини повідомлення