Міністерство освіти і науки України
НТУУ «Київський політехнічний інститут ім. Ігоря Сікорського»
Навчально-науковий Фізико-технічний інститут

Криптографія

Комп’ютерний практикум №2.

Криптоаналіз шифру Віженера

Варіант №3

Виконав:

Студент групи ФБ-31

Васалатій А. Ю., Яковчук О. С.

Перевірено:

Байденко П. В.

27.09.2025

Першочергово ми ознайомилися з поняттям шифру поліалфавітної підстановки, а далі власне шифром Віженера як його представником. Беручи за основу правила зашифрування і розшифрування для нього ми реалізували програмну імплементацію:

$y_i = (x_i + k_{i \bmod r}) \bmod m;\ x_i = (y_i - k_{i \bmod r}) \bmod m;\ i = \overline{0, n};$

де x_i *та* y_i *- літери ВТ та ШТ відповідно;* n - їх довжина; $k_{i \bmod r}$ *- символ ключа, який використовується зашифрування літери ВТ;* r *- довжина ключа і* m *- потужність алфавіту.*

Далі ознайомившись з методами першого етапу криптоаналізу шифру Віженера (пошуку довжини ключа) нами було створено скрипт, який, що обчислює індекс відповідності для ВТ (у файлі за шляхом, що був переданий при запуску) за формулою:

$I(Y) = \frac{1}{n(n-1)} \sum_{t \in Z_m} N_t(Y)(N_t(Y) - 1)$, де $N_t(Y)$ кількість появ літери t в ШТ Y.

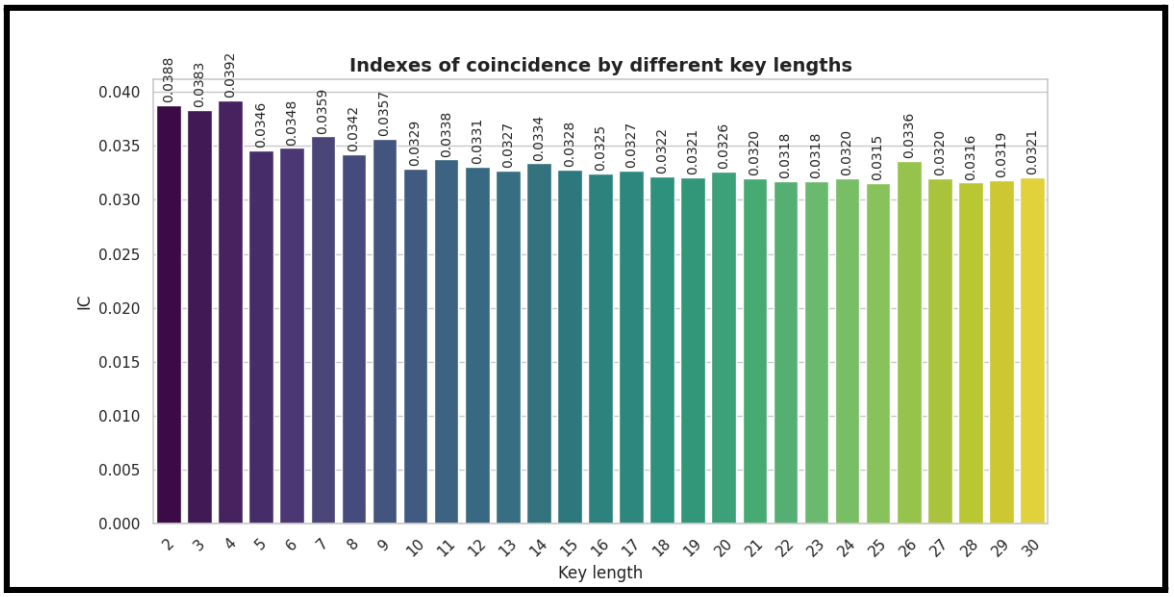
Потім ним формуються випадкові ключі довжиною від 2 до 30 символів включно, які використовуються для отримання відповідних шифротекстів; далі обчислюються їх індекси відповідності, а результати (ключ, шифрування, індекс) виводяться в окремі текстовий файл, (довжина ключа, індекс) xlsx, а також представляються на стовпчиковій діаграмі.

Приклад роботи скрипта:

```
================================================================
================================================================
File type of texts/test_pt.txt: text/plain
Encoding detection result: {'encoding': 'utf-8', 'confidence': 0.99, 'language': ''}
Used random keys: ['щн', 'дзп', 'тпмш', 'суктю', 'яцйнъх', 'пчцтшха', 'ьшховчва', 'эфтсотшта', 'муъчцъпхжа', 'уаацятхцибй', 'шсоасейхшхфб',
ицжлищабеыафу', 'иьяятцлкььбсгя', 'эжоыицъйасеыаьс', 'чйбюжптчфчхщолес', 'ихххвхошмяоллубкщм', 'чязрдфаоэгфдюдсзцу', 'зкътцрщызюугсфаъщжб',
ьыщппйфышыбянаяагть', 'отшмязжрюаюъгщйяыкгжъ', 'оыгыжмхузжплваомттацсю', 'хщыкъдсллщцъеяоюуоайчум', 'гыгцзпвнюсйжфвречаызйюсщ', 'ьхозыймхыаи
боргсюеяжфмэъ', 'щгзнвхгтаквдгюаъдввушвщчвг', 'юнлуржлйъцюцфожиобмзфчэигиа', 'жьйъоэсжттуйгфъкуихзщыюлвурх', 'шйблоедючвгьмвтмбуаежквпущгый',
 'суозглгсьеагщнаейрсжфгрффщезня']
Index of coincidence of plain text: 0.053743343655809205
```

key_len	ic
2	0.0387941
3	0.0383371
4	0.0392201
5	0.0346237
6	0.0348145
7	0.0358859
8	0.0342114
9	0.0356888
10	0.0329232
11	0.0337566
12	0.0330972

12	0.0330972
13	0.032744
14	0.0333858
15	0.0328198
16	0.0324545
17	0.0327156
18	0.0321828
19	0.0321196
20	0.0326361
21	0.0320132
22	0.0317617
23	0.0317624
24	0.0319979
25	0.0315453
26	0.0336087
27	0.0320279
28	0.0316149
29	0.0318519
30	0.032121



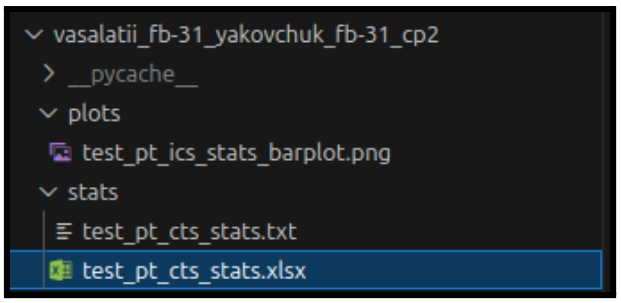

```
Key: щн
жырйитйтээзуэткяыыеьзюдтэъбцэтжйитйтээзуэткяыыеьйыстдфбщжмшмкъщмжырйжнкямьбшщрдмжадхапюфэиеткмппюшбдщпзьзсжк
Index of coincidence: 0.03879413744689805

Key: дзп
схжацффмуфххимацйэрцэхтфифчнлфсгюйчфичэклфхщстуюфхзйтцмуьгжохфпгфэыгьдшбчцчпзтпжьчтчлйфллкрмагэсйтчызстцэифс
Index of coincidence: 0.038337117283039075

Key: тпмш
яэгфбфьэцяъюцфэкфэшзаачэцьфбцфщфбфьэцяъюцфэкфэшзвэдээцфдяолчгьмчяэгфяпэкеюфгттччявчащссяцкшэговъчъфптсъзаущ
Index of coincidence: 0.03922013868454123

Key: суктю
юббонцгпцоящочпгхшюнядхчвюыуцгюпщчоцчъадхшыдаяящвмйшхщжэайсэвакслкжяювеэбжьунээюжхъеушсцщэшысфушхъхсхшбмхаi
Index of coincidence: 0.034623708687491025

Key: яцйнъх
мдаййъпынэиыгыъяьглечюеъггсцюъмтшткъгжчуюърилыждпдбтеьзвцмщфргймзгцтцнлзтесшъшкхцаеэжшофюрлыъмрчдбсдъчнечсзс
Index of coincidence: 0.034814493048347245
```

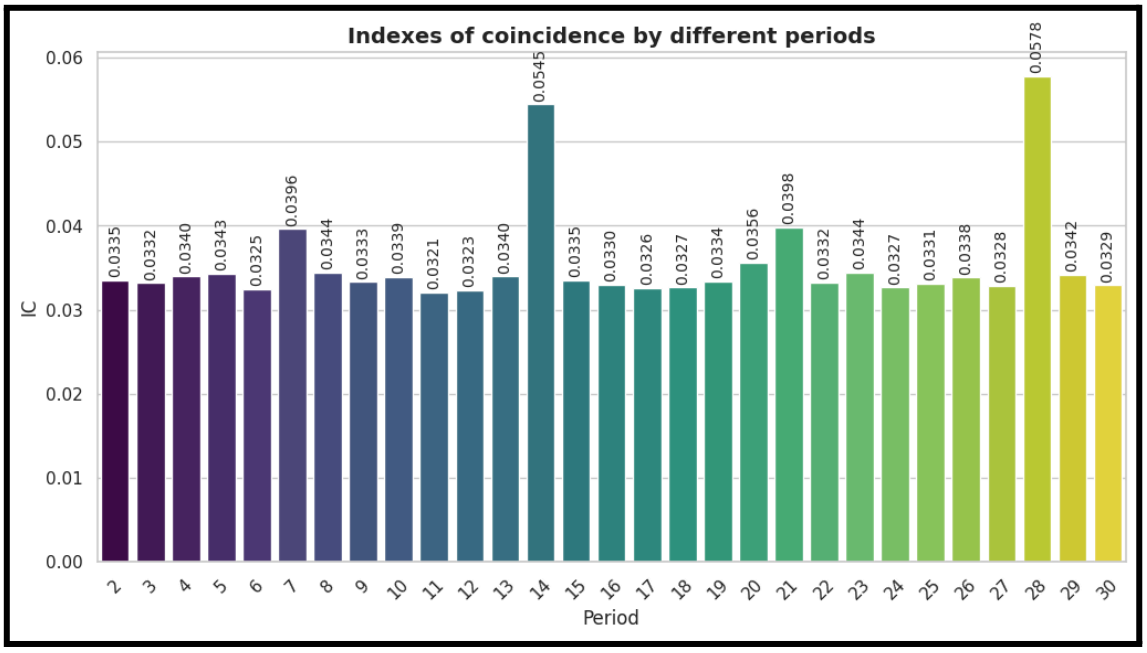


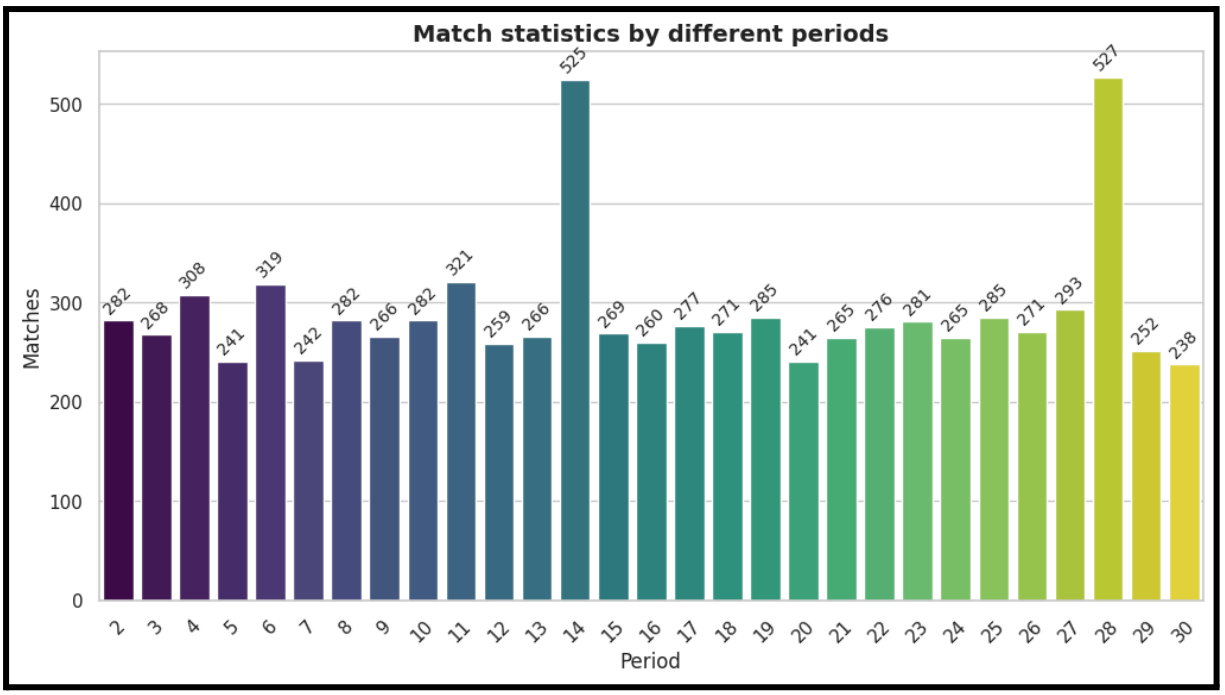
Дослідивши методи визначення довжини ключа, а також алгоритм подальших дій, як нам було запропоновано в методичних рекомендаціях, ми перейшли до другого етапу практикуму - розшифруванню наданого нам за варіантом ШТ.

Спочатку ми розрахували індекс відповідності ШТ, однак однозначно визначити довжину ключа через порівняння з ІВ отриманими на першому етапі практикуму не вдалося, бо даний алгоритм ефективний при застосування невеликих ключів довжини 2-5 символів. Тому далі ми застосували підхід із розбиття ШТ на блоки з різними періодами від 2 до 30 і обчисленням їх ІВ, а також вирішили порахувати статистику збігів з аналогічними періодами за формулою:

$D_r = \sum_{i=1}^{n-r} \delta(y_i, y_{i+r})$, де $\delta(y_i, y_{i+r})$ - символ Кронекера.

Серед отриманих значень сильно вирізнялися ті, що відповідали періодам 14 та 28, зокрема індекси відповідності наближалися до притаманних ВТ російською мовою. З цього можна зробити висновок, що довжина ключа - 14 символів.

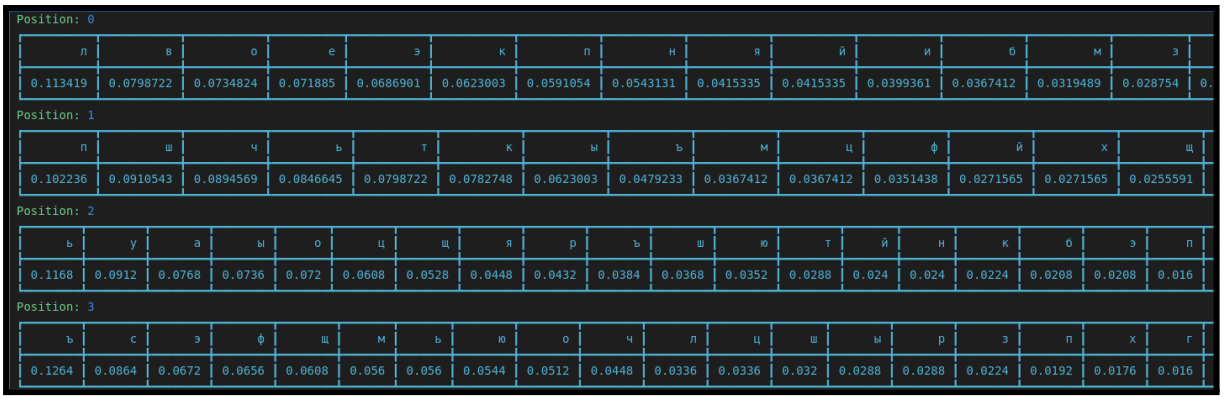




Мова	Індекс збігів
російська	0.0553[1]
англійська	0.0644[1] 0.0667[2]
італійська	0.0738[2]
іспанська	0.0775[2]
німецький	0.0762[2]
французька	0.0778[2]
ведійський санскрит	0.021076696
пракрит	0.046635758
класичний санскрит	0.045567736
гінді	0.041837864
урду	0.057535302

(source)

Виходячи з отриманої довжини ключа було розраховано частоти символів на кожному періоді (нижче наведено уривок, таблиці збережені у ipynb файлі):



Спробувавши застосувати розшифрування з ключем “о” для символів, що найчастіше зустрічаються (“о” - найчастіше зустрічається в російській мові) отримали “**эбомчцтникфуьо**”. Беручи до уваги надану нам інформацію про змістовність ключа і схожість “**мчцтникфуьо**” на “**маятникфуко**” *(фізичний дослід Леона Фуко в 1851 році, а також «Маятник Фуко» (італ. Il pendolo di Foucault) — історичний детективно-філософський пародійний фантастичний роман Умберто Еко” (source))*, ми спробували застосувати розшифрування з ключем “о” до других за частотою символів на кожній позиції, якщо показники частот досить мало відрізнялися, й отримали наступне:

```
Position 0: ['э']
Position 1: ['б', 'к']
Position 2: ['о']
Position 3: ['м']
Position 4: ['ч', 'а']
Position 5: ['ц', 'я']
Position 6: ['т']
Position 7: ['н']
Position 8: ['и']
Position 9: ['к', 'б']
Position 10: ['ф']
Position 11: ['у']
Position 12: ['ь', 'к']
Position 13: ['о', 'н']
```

Застосувавши ключ “**экомаятникфуко”** для розшифрування наданого нам ШТ, отримали (уривок, повний ВТ збережений в окремий файл):

“итутяувиделмаятникшарвисящийнадолгойнитиопущеннойсвольтых оравизохронномвеличииописывалколебанияязналноивсякийощутилбып одчарамимернойпульсации”

Висновки

Першочергово, під час виконання практикуму нами були засвоєні теоретичні відомості пов’язані з застосуванням і аналізом шифру Віженера. Були розглянуті алгоритми шифрування та розшифрування; кілька методів пошуку довжини ключа, що ним був зашифрований ВТ, які стосувалися як підрахунку й аналізу індексів відповідності, так і статистики збігів символів у ШТ; а також алгоритм розшифрування шифру

Цезаря на основі частотного аналізу до ряду застосування якого зводиться розшифрування шифру Віженера після визначення довжини ключа.

Базуючись на отриманих знаннях, в першій частині практикуму нами було реалізовано програмні імплементації шифру Віженера і калькулятора індексів відповідності для ШТ з ключами різної довжини на основі наявного ВТ. Далі, застосовуючи код реалізований в межах першої частини практикуму, нами було здійснено розшифрування ШТ отриманого за варіантом. Отриманий ключ: **“экомаятникфуко”**, розшифрований текст наведений у файлі var_3_pt.txt.