***Теоретична інфа***

Основою для виконання лаборатної роботи було математичне правило, що якщо ми маємо 2 повідомлення А та В, і їх хеш

*E(A) = A xor C*

*E(B) = B xor C*

, то

*E(A) xor E(B) = (A xor C) xor (B xor C) = A xor B xor C xor C = A xor B.*

Виходячи з цього правила, ми можемо підставляти часто вживані слова замість D в наступному виразі

*A xor B xor С*

, і якщо, наприклад, повідомлення А міститиме всередині слово С, то в результаті цього виразу ми отримаємо відповідну частину повідомлення В.

Звісно ж, це працює тільки якщо дотримуватися довжини D (обрізати всі байти довші за цю довжину) в виразі *A xor B xor С* та зміщувати *A xor B,* адже ми не знаємо, в якій саме позиції стоїть це слово. Функція, що це робить наведена на рисунку 1:

Text

Description automatically generated

Рисунок 1 – функція, що застосовує часто вживане слово (кріб) для відгадування

***Демонстрація***

Програма допомагає відгадати рядки, що зашифровані одним ключем із тексту

A screenshot of a computer

Description automatically generated with medium confidence

Спочатку вона питає в користувача які саме 2 рядки ми використовуватимемо:

A screenshot of a computer

Description automatically generated with medium confidence

Далі програма пропонує використати найчастіші англійські слова і подивитись чи не схожий результат на англійський текст. Користувач має задати як багато часто використовуваних слів ми беремо із словника

Graphical user interface

Description automatically generated

Далі користувач має подивитись на пропоновані варіанти і вибрати найбільш підходящий на його думку. Наприклад, нехай користувач вибере to оскільки «fo» схоже на розшифровану слово «for».

Text

Description automatically generated

Тепер програма виводить результат тільки для введеного слова “to”.

Логічним є, що далі має бути пробіл «to ». Спробуємо цю комбінацію:

Text

Description automatically generated

Бачимо, що і справді це слово for, перевіримо далі for з пробілом:

Text

Description automatically generated

«To g» схоже на дієслово, припустимо, що це так, перевіримо декілька найпоширеніших букв, які зустрічаються після g, і в результаті експерементів, отримали, що «to gr» - найкращий варіант.

Text

Description automatically generated

«for w», ми перевірили знову ж декілька букв: голосні та приголосні(r,h), отримали, що h показує хороший вивід, й перевірили слова where та who, who показало результат(уже разом з пробілом):

Text

Description automatically generated

Далі, протестуємо з пробілом «to grunt ».

Text

Description automatically generated

«for who w» виглядає як нам потрібно дієслово, перевіримо найпоширеніші – will і would. Would дало підходящий результат:

Text

Description automatically generated

Додали до «for who would » пробіл, отримаєм:

Graphical user interface, text

Description automatically generated

Найшли сталий вираз «to grunt and sweat», ввели і отримали:

Text

Description automatically generated

Таким чином, ми можемо відновити покроково вихідний текст, хоч це і займе немало часу.

Але ми просто ввели у пошуковій строці браузера for who would bear to grunt and sweat xD, автор твору: Шекспір.