Національний технічний університет України

«Київський політехнічний інститут ім. Ігоря Сікорського»

Кафедра цифрових технологій в енергетиці

ЗВІТ   
з виконання лабораторної роботи №2  
з дисципліни «Безпека інформаційних систем»

«**Шифр Тритеміуса**»

Виконав: студент групи ТР-12 Каркушевський В.Л.

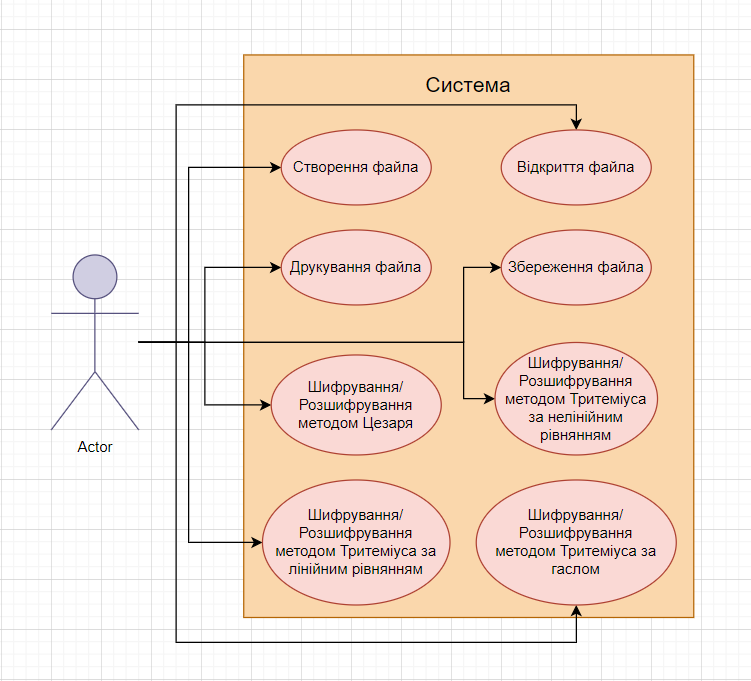
Перевірив: доцент, к.ф.-м.н.  
Тарнавський Ю.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

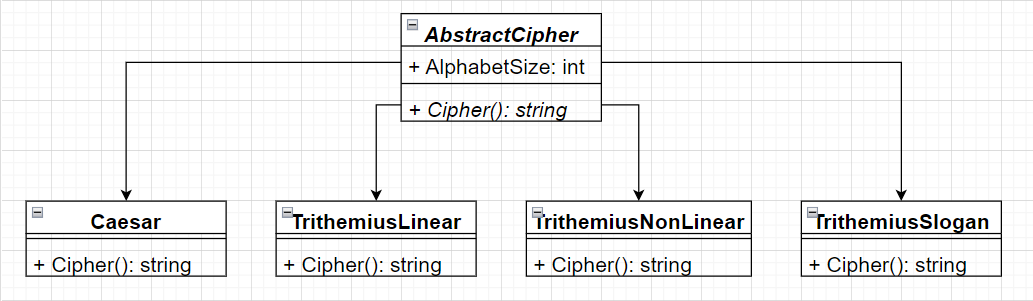
# **Мета роботи**

Розробити криптосистему на основі шифру Тритеміуса

# **Діаграма прецедентів**



# **Діаграма класів**



# **Фрагмент коду з реалізацією алгоритму шифрування/розшифрування**

**AbstractCipher**

public abstract class AbstractCipher  
**{** protected const int AlphabetSize = 1114112**;** // кількість символів в алфавіті Unicode  
   
 public abstract string Cipher(bool variant**,** string[] text**,** string startKey)**;  
}**

**TrithemiusLinear**

public class TrithemiusLinear:AbstractCipher  
**{** public override string Cipher(bool variant**,** string[] text**,** string startKey)  
 {  
 var cipherText = new StringBuilder()int A = (int)Char.*GetNumericValue*(startKey[0])**;** char symbol = startKey[1]**;** int B = (int)Char.*GetNumericValue*(startKey[2])**;** foreach (var lineOfText in text)  
 {  
 for (int i = 0**;** i < lineOfText.Length**;** i++)  
 {  
 int key**;** if (symbol == '+')  
 {  
 key = A \* i + B**;** }  
   
 else  
 {  
 key = A \* i - B**;** }  
   
 if (variant)  
 {  
 int intSymbol = lineOfText[i]**;** int res = (intSymbol + key) % AlphabetSize**;** cipherText.Append((char)res)**;** }  
 else  
 {  
   
 int intSymbol = lineOfText[i]**;** int res = (intSymbol - key + AlphabetSize) % AlphabetSize**;** cipherText.Append((char)res)**;** }  
  
 }  
  
 cipherText.AppendLine()**;** }  
  
 return cipherText.ToString()**;** }  
}

**TrithemiusNonLinear**

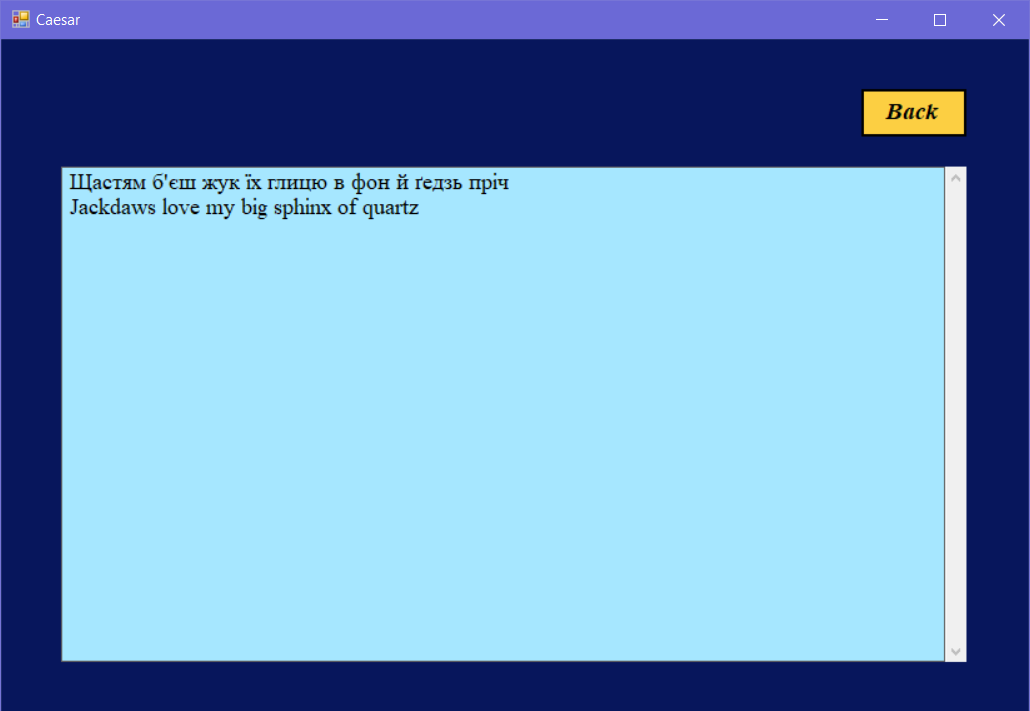
public class TrithemiusNonLinear:AbstractCipher  
**{** public override string Cipher(bool variant**,** string[] text**,** string startKey)  
 {  
 var cipherText = new StringBuilder()**;** int A = (int)Char.*GetNumericValue*(startKey[0])**;** char firstSymbol = startKey[1]**;** int B = (int)Char.*GetNumericValue*(startKey[2])**;** char secondSymbol = startKey[3]**;** int C = (int)Char.*GetNumericValue*(startKey[4])**;** foreach (var lineOfText in text)  
 {  
  
 for (int i = 0**;** i < lineOfText.Length**;** i++)  
 {  
 int key**;** if (firstSymbol == '+' && secondSymbol == '+')  
 {  
 key = A \* i\*i + B\*i + C**;** }else if (firstSymbol == '+' && secondSymbol == '-')  
 {  
 key = A \* i\*i + B\*i - C**;** }else if (firstSymbol == '-' && secondSymbol == '+')  
 {  
 key = A \* i\*i - B\*i + C**;** }else  
 {  
 key = A \* i\*i - B\*i - C**;** }  
 if (variant)  
 {  
 int intSymbol = lineOfText[i]**;** int res = (intSymbol + key) % AlphabetSize**;** cipherText.Append((char)res)**;** }  
 else  
 {  
   
 int intSymbol = lineOfText[i]**;** int res = (intSymbol - key + AlphabetSize) % AlphabetSize**;** cipherText.Append((char)res)**;** }  
 }  
  
 cipherText.AppendLine()**;** }  
  
 return cipherText.ToString()**;** }  
}

**TrithemiusSlogan**

public class TrithemiusSlogan:AbstractCipher  
**{** public override string Cipher(bool variant**,** string[] text**,** string key)  
 {  
 var cipherText = new StringBuilder()**;** foreach (var lineOfText in text)  
 {  
  
 for (int i = 0**;** i < lineOfText.Length**;** i++)  
 {  
 if (variant)  
 {  
 int intSymbol = lineOfText[i]**;** int intKey = key[i % key.Length]**;** int res = (intSymbol + intKey) % AlphabetSize**;** cipherText.Append((char)res)**;** }  
 else  
 {  
 int intSymbol = lineOfText[i]**;** int intKey = key[i % key.Length]**;** int res = (intSymbol - intKey + AlphabetSize) % AlphabetSize**;** cipherText.Append((char)res)**;** }  
 }  
 cipherText.AppendLine()**;** }  
 return cipherText.ToString()**;** }  
  
}

# **Скріншоти програми**

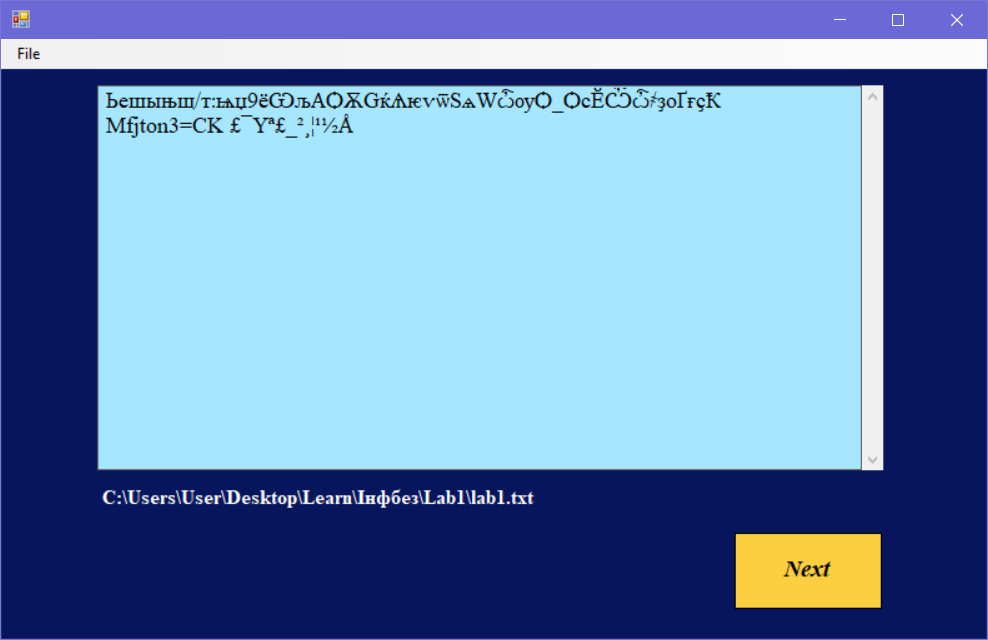
У цій роботі доповнено попередню програму новим шифром Тритеміуса. Розглянемо результат шифрування і розшифрування на прикладі.



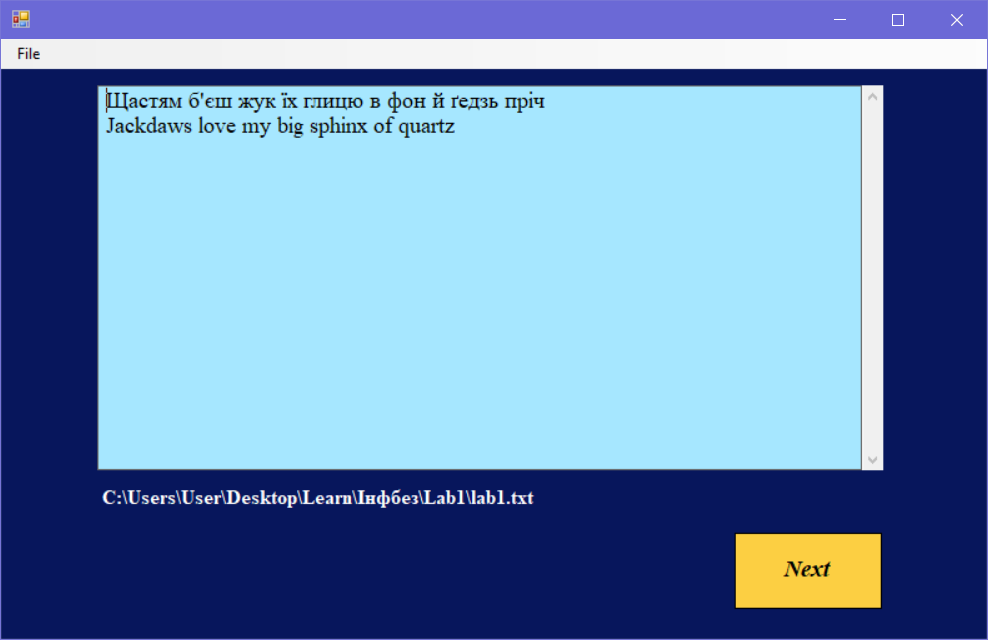
Зашифруємо текст шифром Тритеміуса використовуючи лінійне рівняння у вигляді ключа, наприклад : 2p + 3



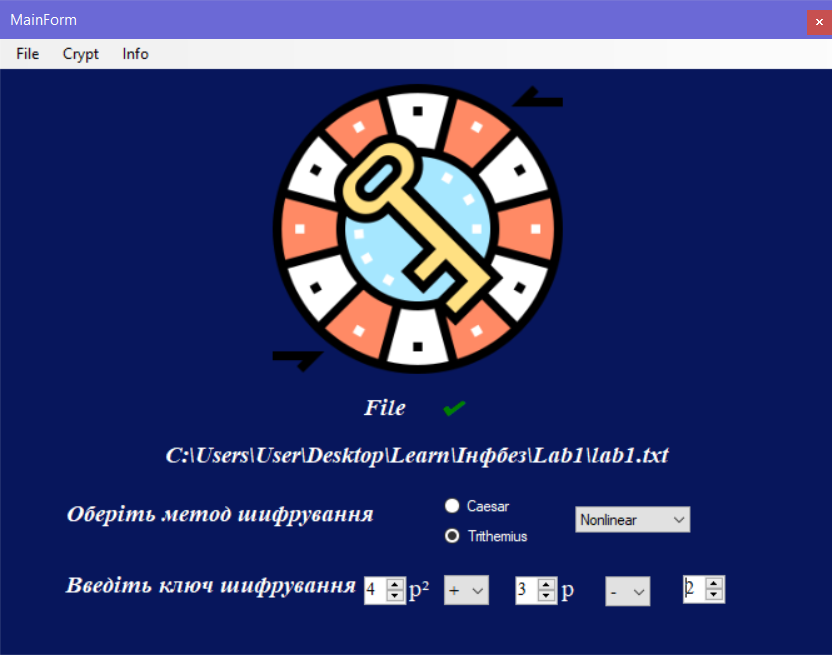
Після шифрування текст виглядає наступним чином:



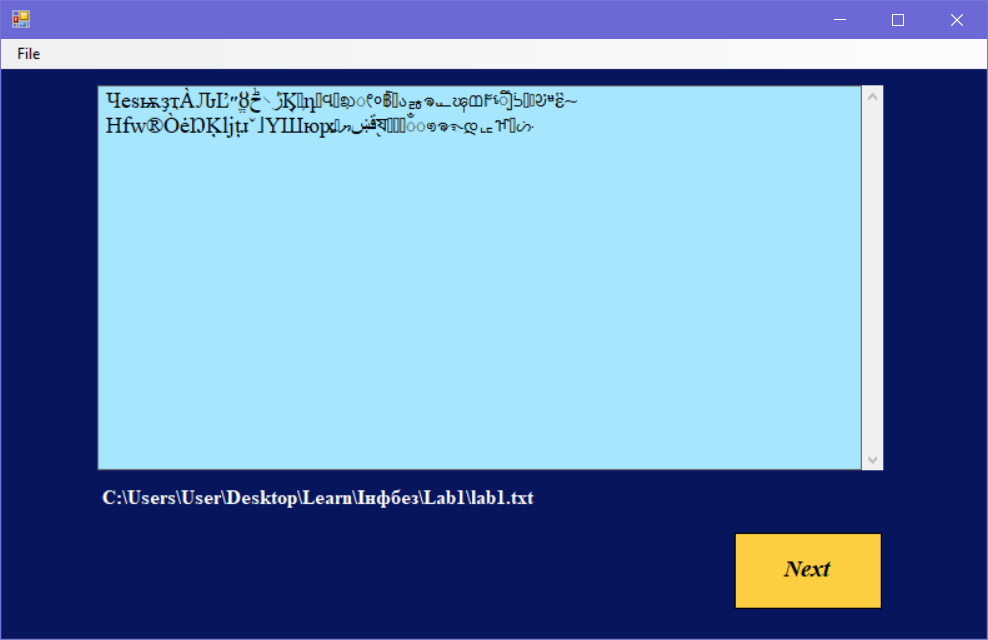
Збережемо результат у файл, та розшифруємо для перевірки результату. Текст буде виглядати наступним чином:



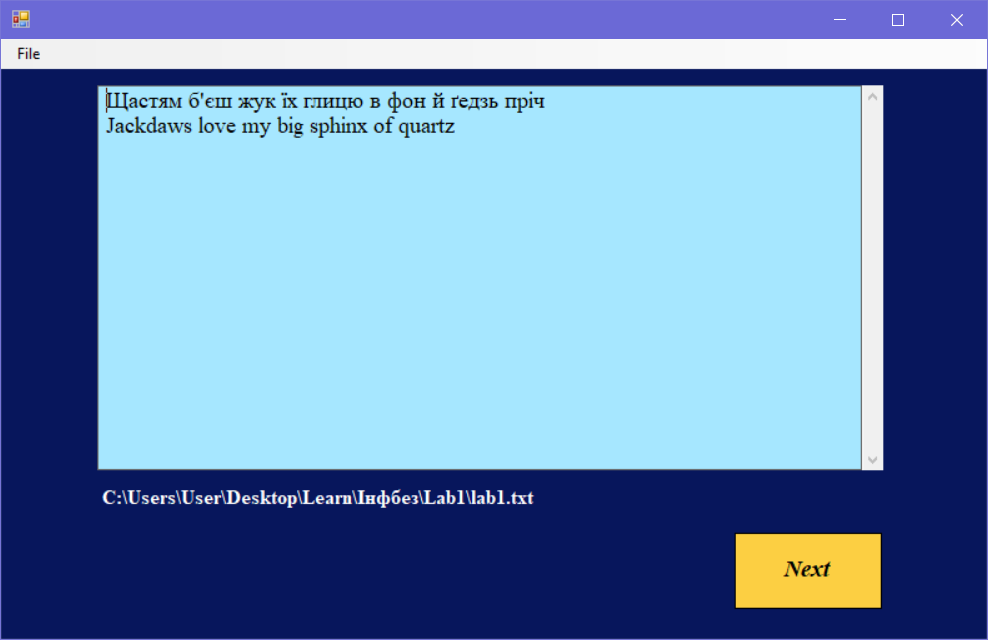
Зашифруємо текст шифром Тритеміуса використовуючи нелінійне рівняння у вигляді ключа, наприклад : 4p ² + 3p - 2



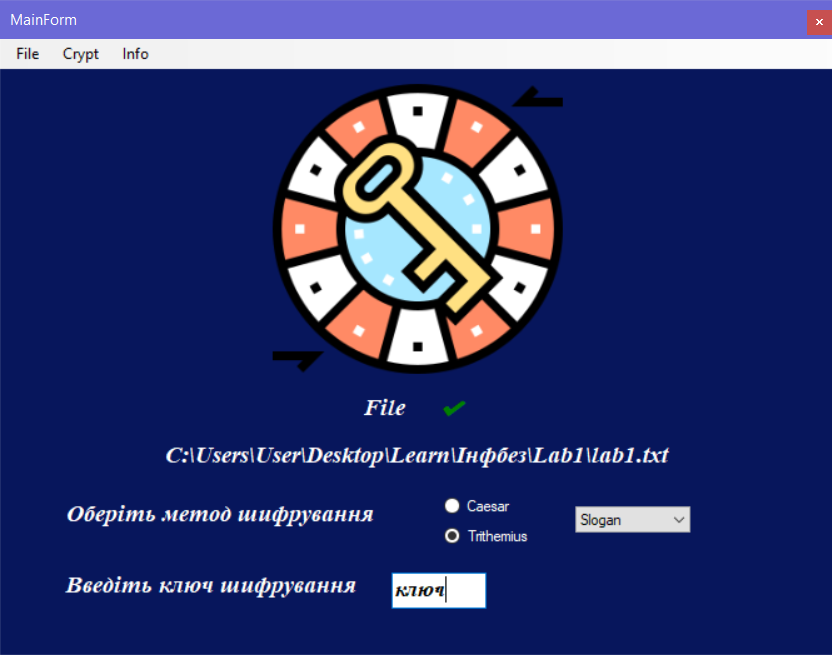
Після шифрування текст виглядає наступним чином:



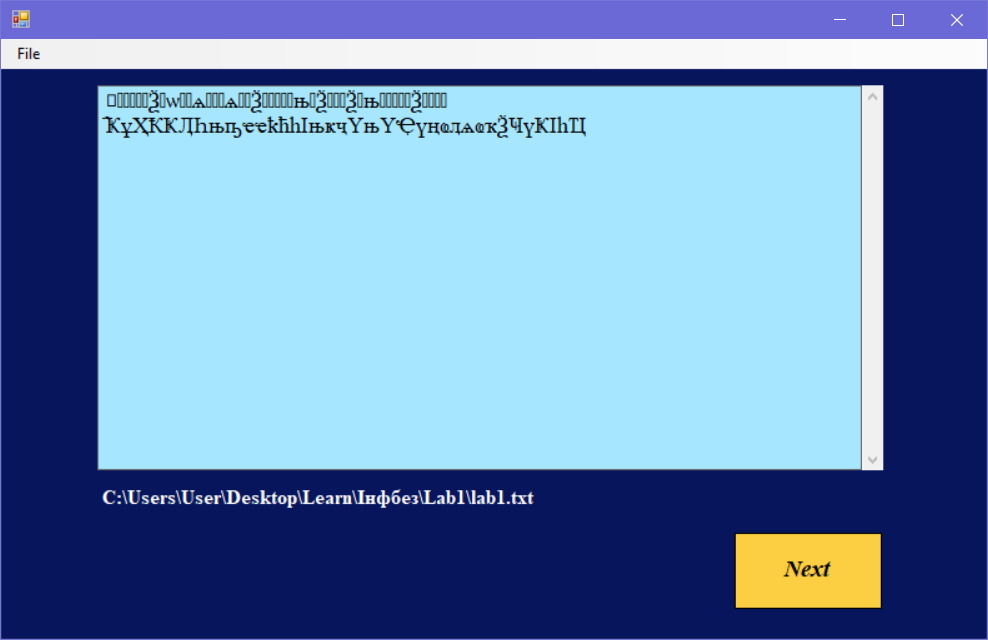
Збережемо результат у файл, та розшифруємо для перевірки результату. Текст буде виглядати наступним чином:



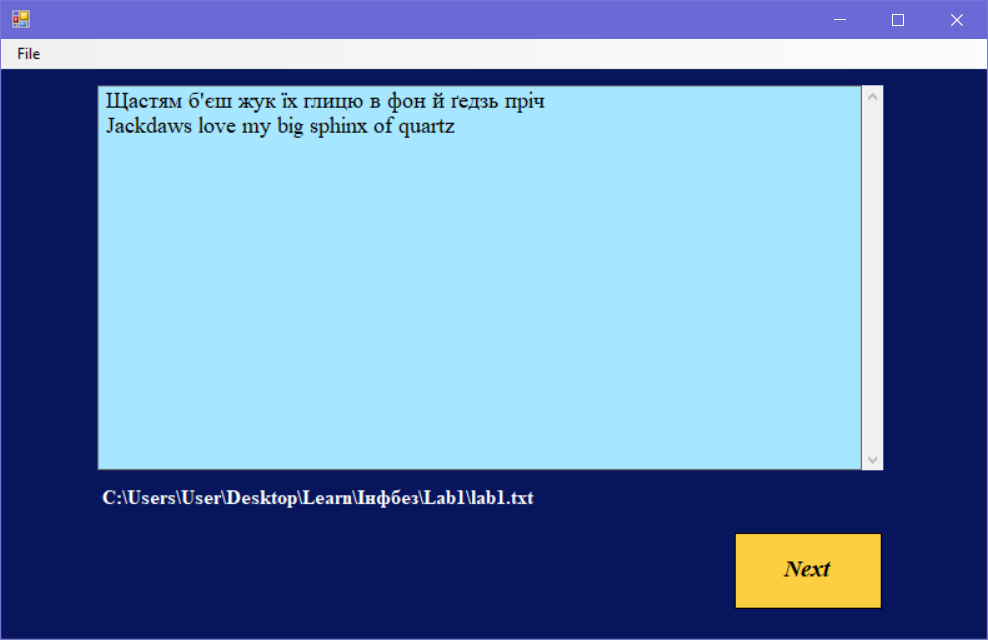
Зашифруємо текст шифром Тритеміуса використовуючи слоган у вигляді ключа, наприклад : ключ



Після шифрування текст виглядає наступним чином:



Збережемо результат у файл, та розшифруємо для перевірки результату. Текст буде виглядати наступним чином:



# **Висновки**

На цій лабораторній роботі було ознайомлено з теоретичними відомостями про шифрування методом Тритеміуса. Модифіковано програму, розроблену у 1 лабораторній роботі, з графічним інтерфейсом на мові програмування C# з використанням Windows Forms. У програмі реалізовано головні функції, такі як: створення, відкривання, збереження, друкування файлів, та доповнено метод Цезаря новими методами Тритеміуса з використанням різних видів ключів. Виконано тестування роботи шифрування та розшифрування.