Національний технічний університет України

«Київський політехнічний інститут ім. Ігоря Сікорського»

Кафедра цифрових технологій в енергетиці

ЗВІТ   
з виконання лабораторної роботи №4  
з дисципліни «Безпека інформаційних систем»

«**Книжковий шифр**»

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

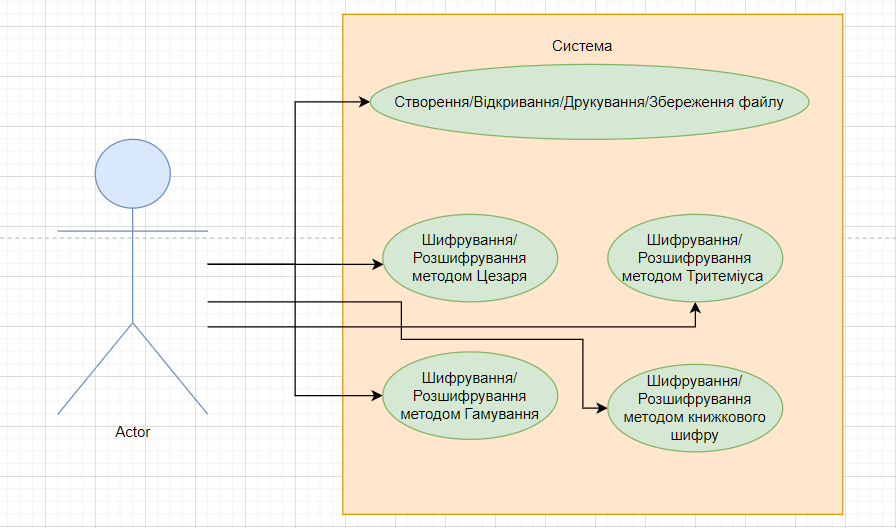
Перевірив: доцент, к.ф.-м.н.  
Тарнавський Ю.А.

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

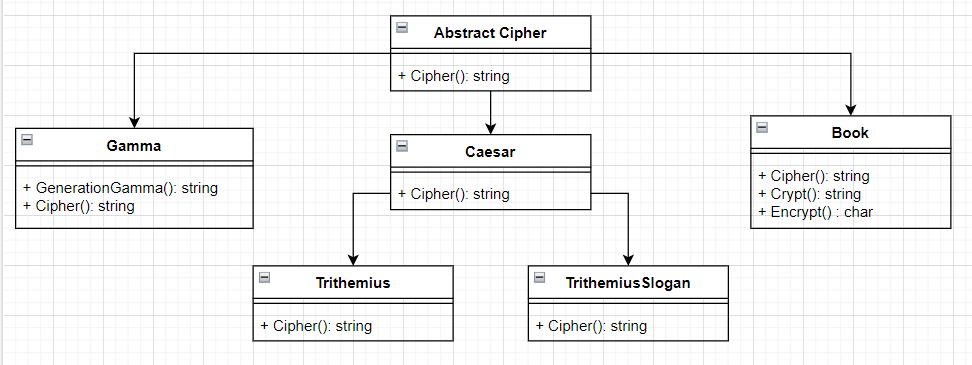
# **Мета роботи**

# Розробити криптосистему на основі використання віршованого фрагменту в якості ключа шифрування

# **Діаграма прецедентів**



# **Діаграма класів**



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

**AbstractCipher**

public abstract class AbstractCipher  
**{** protected const int AlphabetSize = 55295**;** // кількість символів в алфавіті Unicode  
   
 public abstract string Cipher(bool variant**,** string[] text**,** string startKey)**;**}

**Book**

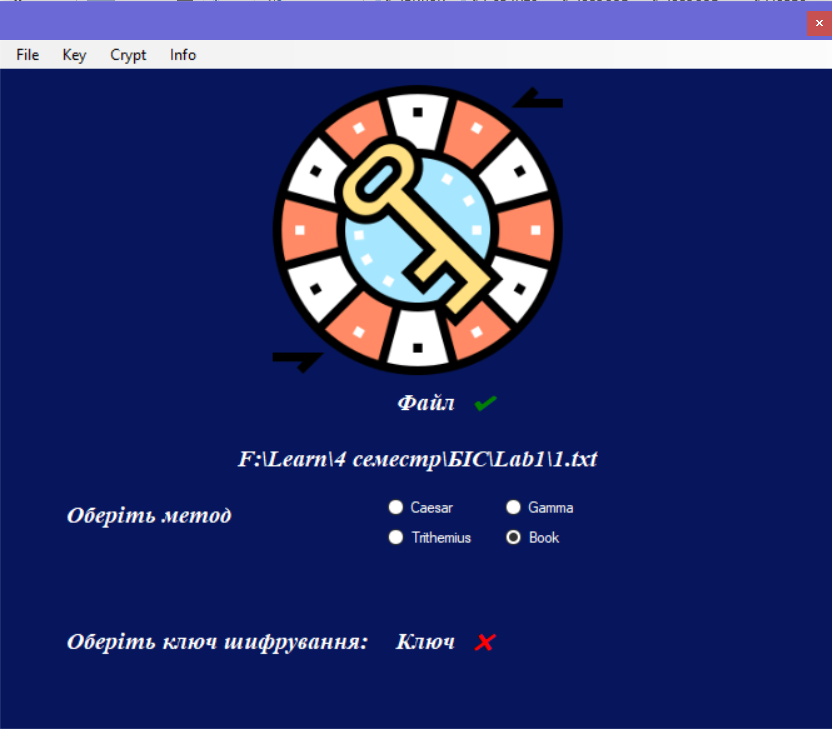
public class Book : AbstractCipher  
**{** public override string Cipher(bool variant**,** string[] text**,** string[] startKey)  
 {  
 var cipherText = new StringBuilder()**;** var random = new Random()**;** foreach (var oneLine in text)  
 {  
 if (variant)  
 {  
 for (var i = 0**;** i < oneLine.Length**;** i++)  
 {  
 var temp = *Crypt*(random**,** oneLine[i]**,** startKey)**;** if (temp == null)  
 {  
 return null**;** }  
  
 if (i != oneLine.Length - 1)  
 {  
 cipherText.Append(temp + ",")**;** }  
 else  
 {  
 cipherText.Append(temp)**;** }  
 }  
 }  
 else  
 {  
 var arr = oneLine.Split(',')**;** foreach (var t in arr)  
 {  
 cipherText.Append(*Encrypt*(t**,** startKey))**;** }  
 }  
  
 cipherText.AppendLine()**;** }  
  
 return cipherText.ToString()**;** }  
  
 private static string *Crypt*(Random random**,** char symbol**,** string[] startKey)  
 {  
 var result = new List<string>()**;** for (var i = 0**;** i < startKey.Length**;** i++)  
 {  
 for (var j = 0**;** j < startKey[i].Length**;** j++)  
 {  
 if (char.*ToLower*(startKey[i][j]).Equals(char.*ToLower*(symbol)))  
 {  
 result.Add($"{i}/{j}")**;** }  
 }  
 }  
  
 return result.Count == 0 ? null : result[random.Next(0**,** result.Count)]**;** }  
  
 private static char? *Encrypt*(string symbol**,** string[] startKey)  
 {  
 var arr = symbol.Split('/')**;** var indexI = int.*Parse*(arr[0])**;** var indexJ = int.*Parse*(arr[1])**;** if (indexI >= startKey.Length || indexJ >= startKey[indexI].Length)  
 {  
 return null**;** }  
  
 return char.*ToLower*(startKey[indexI][indexJ])**;** }  
**}**

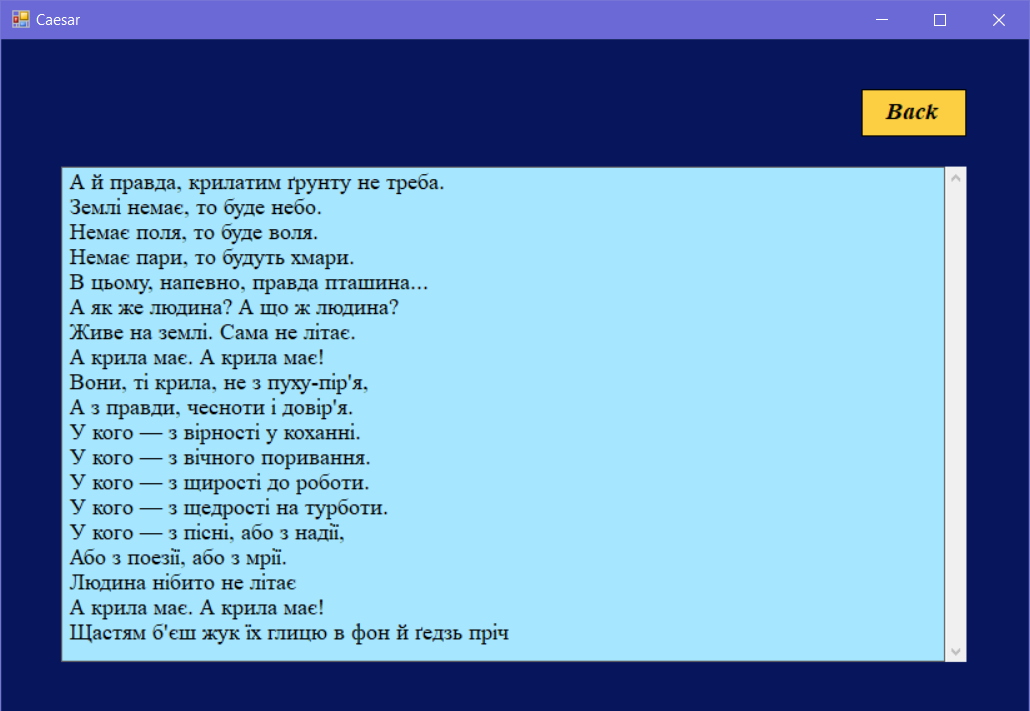
# **Скріншоти програми**

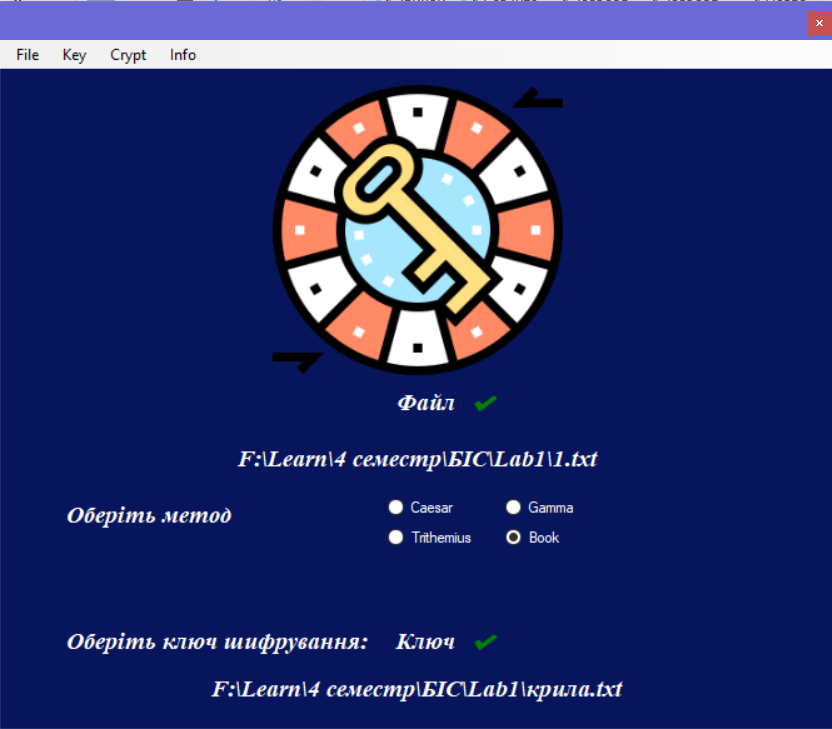
Оберемо панграму



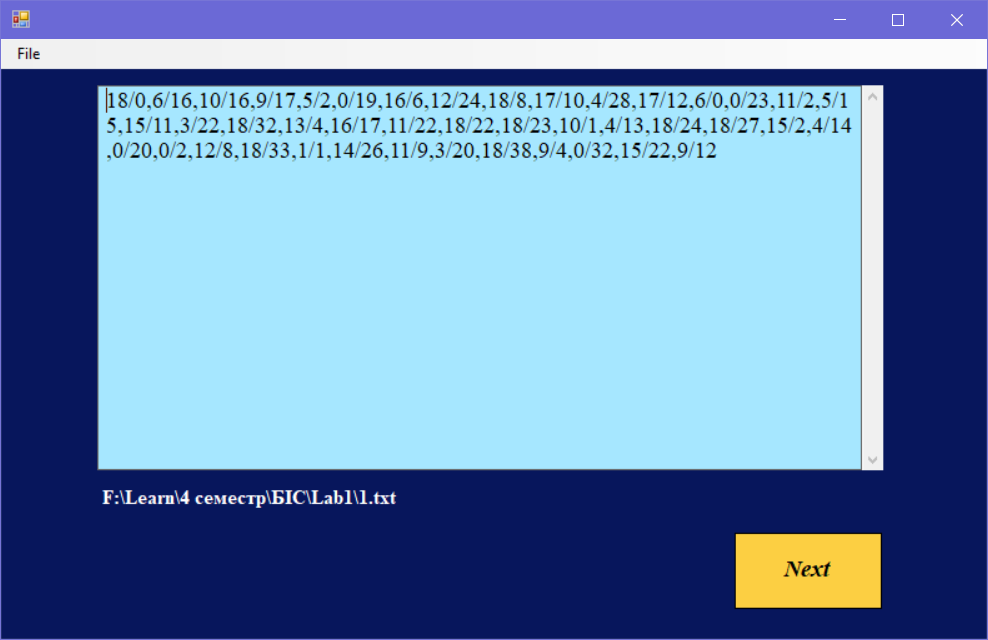
Оберемо метод книжкового шифру та оберемо ключ, у вигляді віршованого тексту.



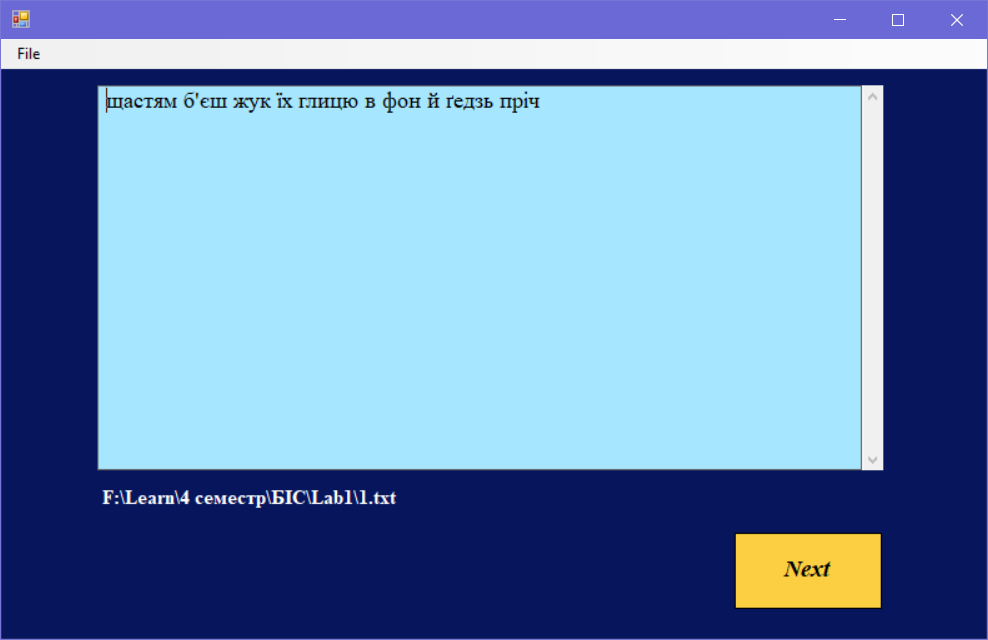




Після шифрування виводиться зашифрований текст



Зберігаємо файл. Для розшифрування вводимо ключ та розшифровуємо. У результаті маємо розшифрований текст



# **Висновки**

На цій лабораторній роботі було ознайомлено з теоретичними відомостями про книжковий метод шифрування. Модифіковано програму, розроблену у 3 лабораторній роботі, з графічним інтерфейсом на мові програмування C# з використанням Windows Forms. У програму доповнено новий метод книжкового шифру . Виконано тестування роботи шифрування та розшифрування.