Національний технічний університет України

«Київський політехнічний інститут ім. Ігоря Сікорського»

Кафедра цифрових технологій в енергетиці

ЗВІТ   
з виконання лабораторної роботи №5  
з дисципліни «Безпека інформаційних систем»

«**Шифр DES**»

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

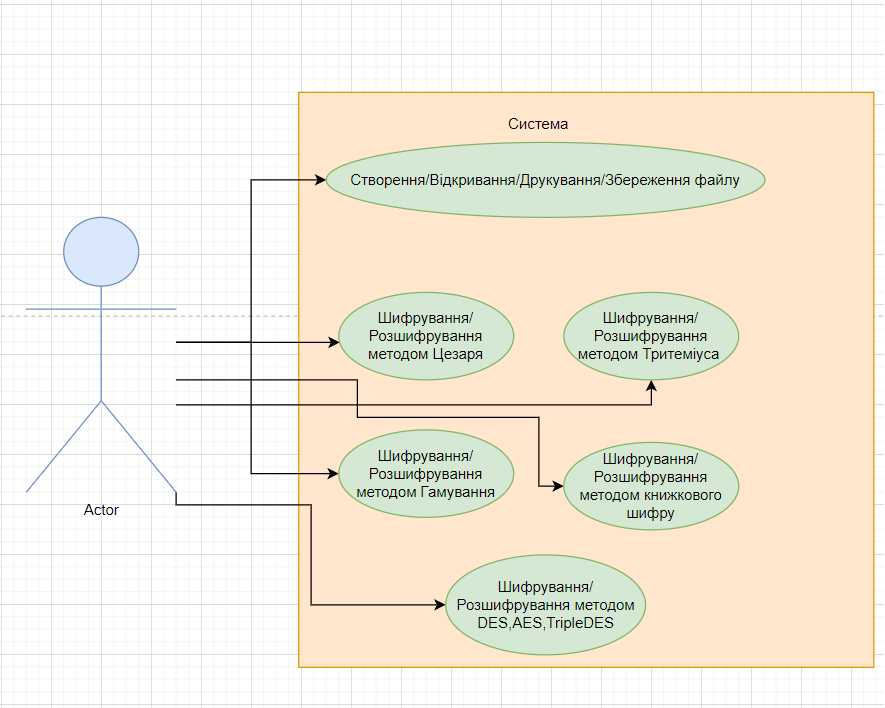
Перевірив: доцент, к.ф.-м.н.  
Тарнавський Ю.А.

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

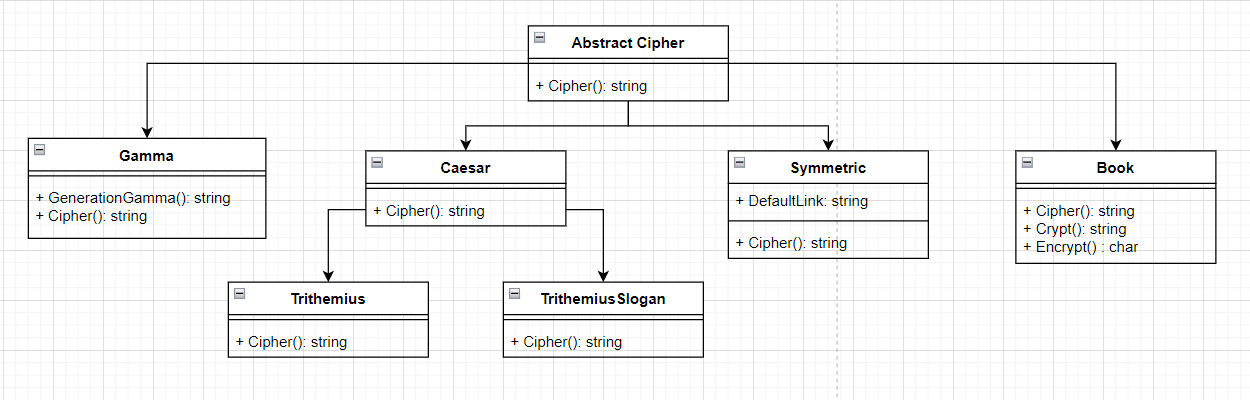
# **Мета роботи**

# Ознайомитись з використанням криптопровайдерів в прикладному програмуванні

# **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)**;**}

**Symmetric**

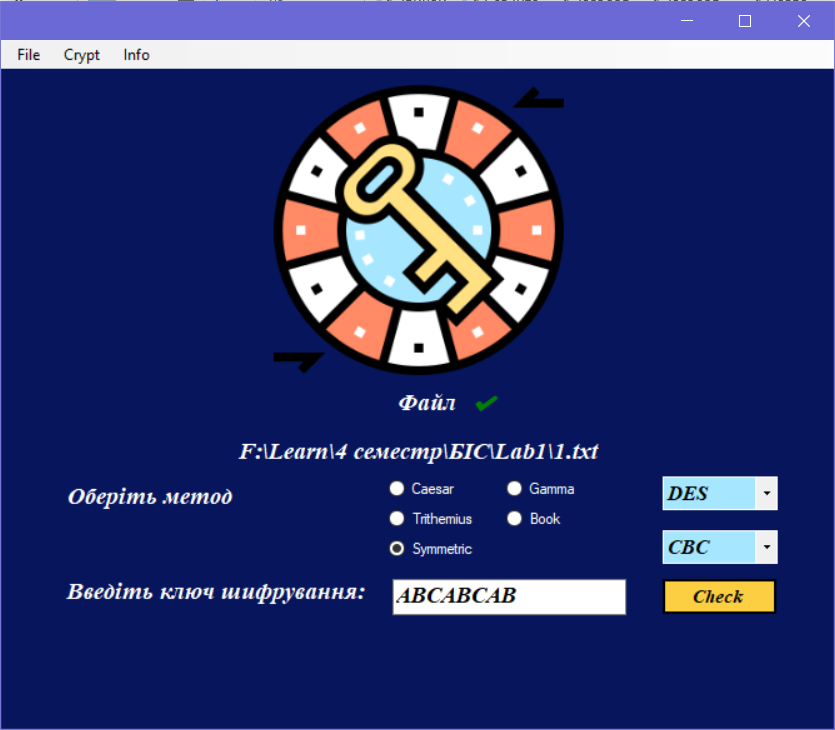
public class Symmetric : AbstractCipher  
**{** private const string DefaultLink = @"F:\Learn\4 семестр\БІС\Lab1\test.txt"**;** public override string Cipher(bool variant**,** string[] text**,** string[] startKey)  
 {  
 SymmetricAlgorithm cryptic**;** switch (startKey[0])  
 {  
 case "DES":  
 cryptic = new DESCryptoServiceProvider()**;** break**;** case "AES":  
 cryptic = new AesCryptoServiceProvider()**;** break**;** default:  
 cryptic = new TripleDESCryptoServiceProvider()**;** break**;** }  
  
 cryptic.Key = Encoding.ASCII.GetBytes(startKey[1])**;** switch (startKey[0])  
 {  
 case "AES":  
 cryptic.IV = Encoding.ASCII.GetBytes(startKey[1].Substring(0**,** 16))**;** break**;** case "TripleDES":  
 cryptic.IV = Encoding.ASCII.GetBytes(startKey[1].Substring(0**,** 8))**;** break**;** default:  
 cryptic.IV = Encoding.ASCII.GetBytes(startKey[1])**;** break**;** }  
  
 switch (startKey[2])  
 {  
 case "CBC":  
 cryptic.Mode = CipherMode.CBC**;** break**;** case "ECB":  
 cryptic.Mode = CipherMode.ECB**;** break**;** default:  
 cryptic.Mode = CipherMode.CFB**;** break**;** }  
  
  
 FileStream stream**;** CryptoStream crStream**;** var res = new StringBuilder()**;** if (variant)  
 {  
 File.*Delete*(DefaultLink)**;** stream = new FileStream(DefaultLink**,** FileMode.OpenOrCreate**,** FileAccess.Write)**;** crStream = new CryptoStream(stream**,** cryptic.CreateEncryptor()**,** CryptoStreamMode.Write)**;** var data = Encoding.UTF8.GetBytes(string.*Join*("\n"**,** text))**;** crStream.Write(data**,** 0**,** data.Length)**;** }  
 else  
 {  
 stream = new FileStream(DefaultLink**,** FileMode.Open**,** FileAccess.Read)**;** crStream = new CryptoStream(stream**,** cryptic.CreateDecryptor()**,** CryptoStreamMode.Read)**;** var reader = new StreamReader(crStream)**;** var result = reader.ReadToEnd()**;** reader.Close()**;** var temp = result.Split('\n')**;** for (var i = 0**;** i < temp.Length-1**;** i++)  
 {  
 res.Append(temp[i])**;** res.AppendLine()**;** }  
 res.Append(temp[temp.Length-1])**;** }  
  
 crStream.Close()**;** stream.Close()**;** return variant ? File.*ReadAllText*(DefaultLink) : res.ToString()**;** }  
}

# **Скріншоти програми**

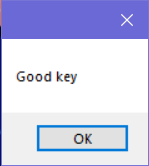
Оберемо панграму



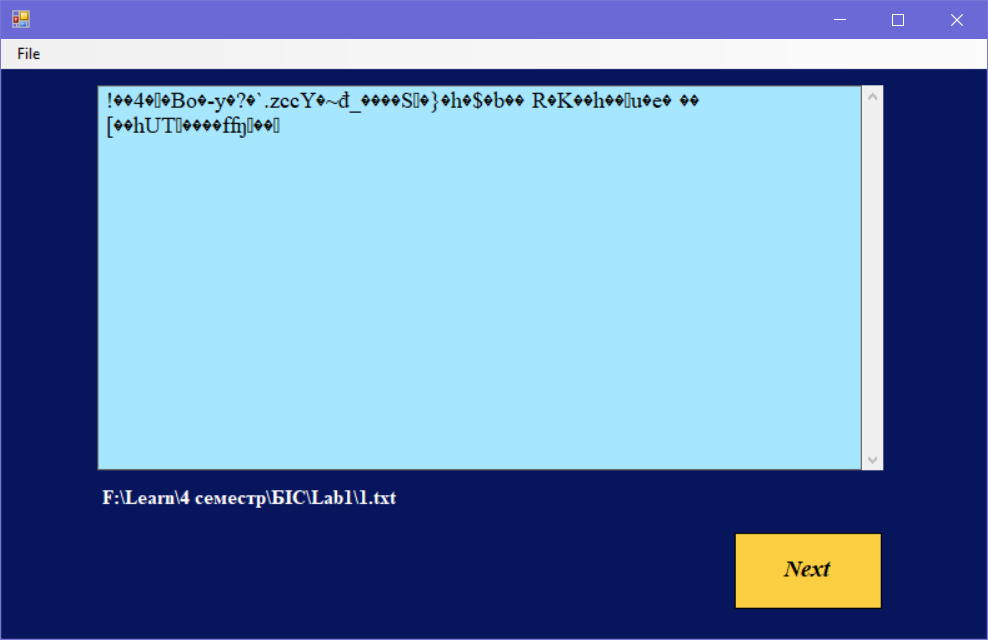
Оберемо метод симетричного шифру DES та режим шифрування CBC та введемо ключ. Довжина ключа повинна бути з довжиною у 8 байтів.



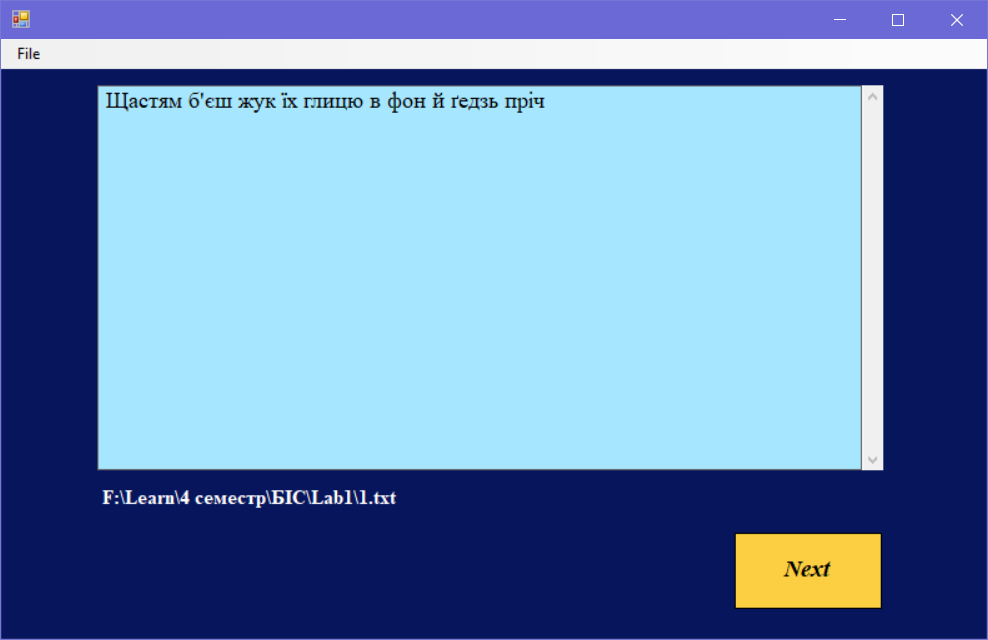
Також перевіримо ключ на коректність, натиснувши кнопку Check



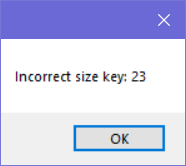
Зашифруємо текст. У результаті отримуємо



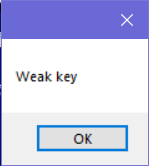
Розшифруємо текст. У результаті отримаємо



У випадку використання AES ключ може містити 16, 24 або 32 байти. У випадку некоректного розміру виводиться повідомлення



У випадку використання TripleDes ключ може містити 24 байти. Але також перевіряється ключ на слабкість. У випадку введення слабкого ключа виводиться повідомлення



# **Висновки**

На цій лабораторній роботі було ознайомлено з теоретичними відомостями про симетричні типи шифрування . Модифіковано програму, розроблену у 4 лабораторній роботі, з графічним інтерфейсом на мові програмування C# з використанням Windows Forms. У програму доповнено новий метод симетричного шифрування . Виконано тестування роботи шифрування та розшифрування.