# Практическая работа № 3

# БЛОЧНЫЕ ШИФРЫ

**Цели:**

1. Рассмотреть симметричный алгоритм блочного шифрования AES.

2. Приобрести навыки использования криптографических классов платформы Microsoft .Net.

3. Закрепить навыки применения технологии Windows Forms. **Приложение 3.1. Программная реализация блочного шифрования**

***Задание:*** Разработать приложение с пользовательским интерфейсом, которое позволит осуществлять шифрование и дешифрование открытых сообщений с помощью алгоритма AES в режиме сцепления блоков шифра на случайно генерируемом секретном ключе.

***Константы:*** отсутствуют.

***Переменные:***

1. text – для ввода и вывода текста (тип string)

***Исходные данные:***

1) text – для ввода и вывода текста

***Результат:*** зашифрованное и расшифрованное сообщение

***Код приложения:***

***Листинг 1 – Код приложения MainForm***

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Security.Cryptography;

using System.IO;

namespace AESApplication

{

public partial class MainForm : Form

{

AesManaged Aes;

byte[] key;

int KeyLength;

MemoryStream msEncrypt;

CryptoStream csEncrypt;

byte[] toEncrypt;

byte[] encrypted;

int[] ksizes;

public MainForm()

{

InitializeComponent();

Aes = new AesManaged();

Aes.GenerateIV();

Aes.Mode = CipherMode.CBC;

KeySizes[] ks = Aes.LegalKeySizes;

KeyLength = ks[0].MinSize;

int i = 0, isel = 0;

int ksmax = ks[0].MaxSize, ksdef = Aes.KeySize;

ksizes = new int[3];

do

{

ksizes[i] = KeyLength;

if (KeyLength == ksdef)

isel = i;

KeyLength += ks[0].SkipSize;

i++;

}

while (KeyLength <= ksmax);

KeyLen.DataSource = ksizes;

KeyLen.SelectedIndex = isel;

Decrypt.Enabled = CipherText.Text.Length == 0;

Encrypt.Enabled = PlainText.Text.Length == 0;

}

private void Encrypt\_Click\_1(object sender, EventArgs e)

{

msEncrypt = new MemoryStream();

toEncrypt = Encoding.Unicode.GetBytes(PlainText.Text);

KeyLength = ksizes[KeyLen.SelectedIndex];

Aes.KeySize = KeyLength;

RNGCryptoServiceProvider rand = new RNGCryptoServiceProvider();

key = new byte[KeyLength / 8];

rand.GetBytes(key);

Aes.Key = key;

ICryptoTransform encryptor = Aes.CreateEncryptor(Aes.Key, Aes.IV);

csEncrypt = new CryptoStream(msEncrypt, encryptor,

CryptoStreamMode.Write);

csEncrypt.Write(toEncrypt, 0, toEncrypt.Length);

csEncrypt.Close();

encrypted = msEncrypt.ToArray();

CipherText.Text = Encoding.Unicode.GetString(encrypted, 0,

encrypted.Length);

Encrypt.Enabled = false;

Decrypt.Enabled = true;

KeyLen.Enabled = false;

msEncrypt.Close();

PlainText.Clear();

}

private void Decrypt\_Click\_1(object sender, EventArgs e)

{

ICryptoTransform decryptor = Aes.CreateDecryptor(Aes.Key, Aes.IV);

encrypted = Encoding.Unicode.GetBytes(CipherText.Text);

msEncrypt = new MemoryStream(encrypted);

csEncrypt = new CryptoStream(msEncrypt, decryptor,

CryptoStreamMode.Read);

toEncrypt = new byte[encrypted.Length];

csEncrypt.Read(toEncrypt, 0, encrypted.Length);

PlainText.Text = Encoding.Unicode.GetString(toEncrypt, 0,

toEncrypt.Length);

csEncrypt.Close();

Decrypt.Enabled = false;

Encrypt.Enabled = true;

msEncrypt.Close();

}

}

}

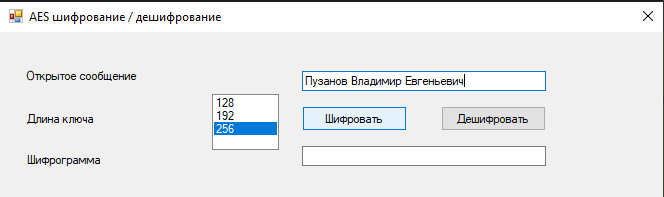


Рисунок 1. Ввод открытого сообщения и выбор длины ключа

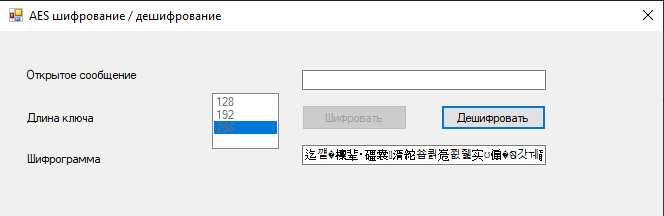


Рисунок 2. Шифрование

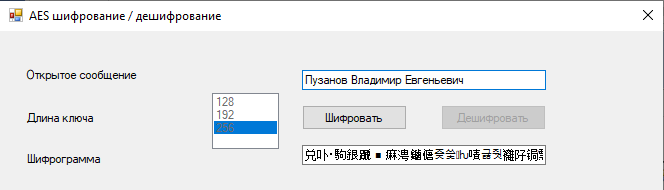


Рисунок 3. Дешифрование

Выполнил студент Пузанов В. Е., ФИТУ 010304-КМСб-о22

Проверил ст. преподаватель каф. ПМ Черноиван Д.Н.