

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки  
Кафедра програмної інженерії

Практична робота №1 з дисципліни: «Безпека програм та даних» на тему:  
«Шифр Цезаря»

Виконав:

студент групи ПЗПІ-20-1 Бабанін А.К.

Перевірив:

доцент кафедри Програмної інженерії Турута О.П

Харків 2022

## Завдання 1

Шифрований текст: жиш лмикизн! Жязщ чми мье ьивжнмгёи. Лзьсьёь щ ь йген  
яжн лмъёь ьильш оёгкмиьъмц л

Ключ: 27

Оригінальний текст: мою сторону! Меня это так возмутило. Сначала я в пику ему  
стала вовсю флиртовать с

Шифрування тексту було виконанно шифром Цезаря. Розшифрування тексту  
виконується наступним чином:  $x = (y + n) - (k \bmod n) \bmod n$

де

$x$  – символ відкритого тексту,

$y$  – символ шифрованого тексту,

$n$  – потужність алфавіту,

$k$  – ключ

### Код CSharp для розшифрування:

```
public string DecryptString(string encryptedString, int key)
{
    var shift = key % this.alphabetLength;

    var sb = new StringBuilder();

    foreach (var encryptedChar in encryptedString)
    {
        // special case
        if (this.specialChars.Contains(encryptedChar))
        {
            sb.Append(encryptedChar);
            continue;
        }

        var caseAddition = !this.IsLower(encryptedChar) ? 0 : this.alphabetLength;

        var encryptedIndex = this.alphabet.IndexOf((char) (encryptedChar +
caseAddition));

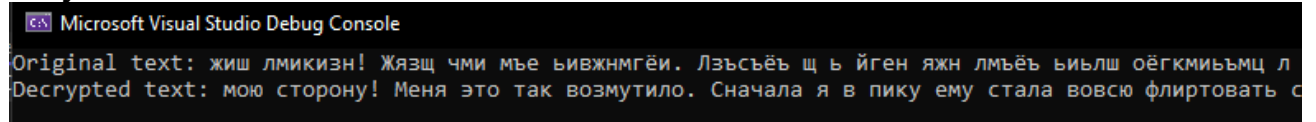
        var decryptedIndex = (encryptedIndex + this.alphabetLength - shift) %
this.alphabetLength;

        var decryptedChar = this.alphabet[decryptedIndex];

        sb.Append((char) (decryptedChar - caseAddition));
    }

    return sb.ToString();
}
```

### Результат виконання:



Microsoft Visual Studio Debug Console

Original text: жиш лмикизн! Жязщ чми мье ьивжнмгёи. Лзсьсьё щ ь йген яжн лмьёь ьильш оёгкмийьмц л

Decrypted text: мою сторону! Меня это так возмутило. Сначала я в пику ему стала всю флиртовать с

## Завдання 2

Шифрований текст: прщвжх ом ьпъщрциф ётът сбнмж рапъмлфск идф башср н 1984 лопъ. ош шррмпъфотсл, гыо рълф йы ычянсллъ нчзшчщчсое иэшочезъкаое в цичрътнн оюуржыооч кчжчл смк слф шогыондй лмррь ачссж, ыо иыо чсшффо мд сччжщью ыщовндящу лътрцтфэицицфс възкъло эхыэфа. пчлоче нщешз ипня димфща ъътлкачисз ксроло чсшз урлтъинчй цщиыыоощаасчръкът гъфончлъхкът, нъ к 2000 гъму, мфаочдлщя ъмнът иукеэынът укрвфхоэыи н ёлчспюсчръкът къспючгъиффс, ипню ямаччсз коыфоюстз к жфрнз.

Ключ: Али

Оригінальний текст: Первым об упрощении этой схемы задумался Ади Шамир в 1984 году. Он предположил, что если бы появилась возможность использовать в качестве открытого ключа имя или почтовый адрес Алисы, то это лишило бы сложную процедуру аутентификации всякого смысла. Долгое время идея Шамира оставалась всего лишь красивой криптографической головоломкой, но в 2000 году, благодаря одной известной уязвимости в эллиптической криптографии, идею удалось воплотить в жизнь.

Шифрування тексту було виконанно шифром Віженера. Розшифрування тексту виконується наступним чином:  $mi = (ci - ki) \bmod n$

де

$ci$  – індекс символу шифрованої букви

$ki$  – індекс символу ключа

$n$  – потужність алфавіту,

Код CSharp для розшифрування:

```
public string DecryptString(string encryptedString, string key)
{
    var resultBuilder = new StringBuilder();

    var processedCharCount = 0;

    for (var i = 0; i < encryptedString.Length; i++)
    {
        var textChar = encryptedString[i];
        var keyChar = key[processedCharCount % (key.Length)];

        if (this.specialChars.Contains(textChar))
        {
            resultBuilder.Append(textChar);

            continue;
        }

        var textCaseAddition = !this.IsLower(textChar) ? 0 : this.alphabetLength - 1;
        var keyCaseAddition = !this.IsLower(keyChar) ? 0 : this.alphabetLength - 1;

        var textCharIndex = this.alphabet.IndexOf((char)(textChar +
textCaseAddition));
        var keyCharIndex = this.alphabet.IndexOf((char)(keyChar + keyCaseAddition));

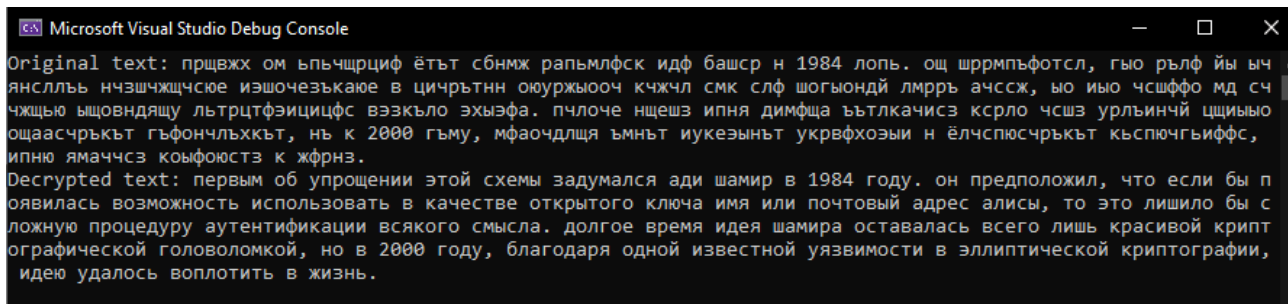
        var encryptedIndex = (textCharIndex - keyCharIndex + this.alphabetLength) %
this.alphabet.Count;
        var encryptedChar = this.alphabet[encryptedIndex];

        resultBuilder.Append(encryptedChar);

        processedCharCount++;
    }

    return resultBuilder.ToString();
}
```

## Результат работи:



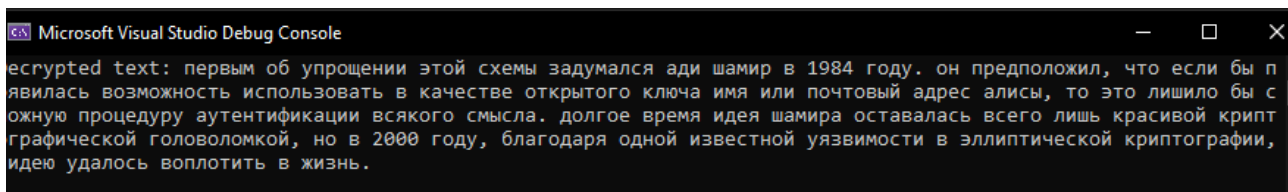
```
Microsoft Visual Studio Debug Console
Original text: прщвжх ом ьпчщрциф ётёт сбнмж рапъмлфск идф башср н 1984 лопь. оц шррмпъфотсл, гьо рълф йы ыч
янсллъ нчзшчщсые изшочезъкаые в цичрътнн оюржыооч кчжчл смк слф шогьондй лмррь ачссж, ьо иью чсшффо мд сч
чжщью ышовндящу льтрцтфэицифс възкьло эхызфа. пчлоче нщешз ипня димфща ьътлкачисз ксрло чсшз урльинчй цщиьюо
ощаасчръкът гъфончлъхкът, нь к 2000 гъму, мфаочдлщя ьмнът иукезынът укрвфхозыи н ёлчспюсчръкът кьспючгъифс,
ипню ямаччсз коыфюотсз к жфрнз.
Decrypted text: первым об упрощении этой схемы задумался ади шамир в 1984 году. он предположил, что если бы п
оявилась возможность использовать в качестве открытого ключа имя или почтовый адрес алисы, то это лишило бы с
ложную процедуру аутентификации всякого смысла. долгое время идея шамира оставалась всего лишь красивой крипт
ографической головоломкой, но в 2000 году, благодаря одной известной уязвимости в эллиптической криптографии,
идею удалось воплотить в жизнь.
```

## Завдання 3

Зашифрований текст: ыникжы ьй лшьэенёсф лючв ьбушд аипвшидък опс ришчь к 1984  
ычпв. ьц зщртычдчтчч, акч рачс щд ьэккбфлаз кжршэтцжъюк фъзччкучъиюк н ушараюкэ  
чющъдкчюэ цфцал чшз бфф юъакчнйх иыщра лфбъж, бъ ёкч ччдсдч мй эфжпщвй шичвупийь  
лвюнёыфгфушяфч нъчуъсь ьедэъл. мжфоэр киншн фмэз дошсии ьаюиъичюэе ьърсь фббз  
щыййснэх уисыбълииагчнйуъш очдчнэччеуъш, щч ь 2000 лътя, йдиоэпииз ьтщчв сурръкцъш  
язакфыъъкс н лчфбшюгчнйуъш цщбшюэошщэфч, фмэж ятлфжъз рьшдчючюе ь пфцще.

Ключ: Лишило

Використовуючи дешифрування шифру Віженера отримуємо наступний результат:



```
Microsoft Visual Studio Debug Console
Decrypted text: первым об упрощении этой схемы задумался ади шамир в 1984 году. он предположил, что если бы п
оявилась возможность использовать в качестве открытого ключа имя или почтовый адрес алисы, то это лишило бы с
ложную процедуру аутентификации всякого смысла. долгое время идея шамира оставалась всего лишь красивой крипт
ографической головоломкой, но в 2000 году, благодаря одной известной уязвимости в эллиптической криптографии,
идею удалось воплотить в жизнь.
```