

## Лабораторна робота 2

### Шифр Тритеміуса

Кравець Ольга

ПМО-31

Інтерфейс криптографічної системи:

The screenshot shows a web application window titled "МОК". It has a navigation bar with tabs: "Файл", "Атака", "Частотні таблиці", "Шифрування", "Про розробника", "Допомога", and "Вихід". The "Шифрування" tab is active. The interface includes two large text input areas labeled "До:" and "Після:". To the right of these areas are three settings sections: "Команда" with radio buttons for "Зашифрувати" and "Розшифрувати"; "Мова" with radio buttons for "Українська" and "English"; and "Тип" with radio buttons for "Лінійне рівняння", "Нелінійне рівняння", and "Гасло". A "Виконати" button is located at the bottom right.

Лінійне шифрування українською мовою:

This screenshot shows the same interface as before, but with data entered. The "До:" field contains the word "кравець". The "Після:" field contains the encrypted word "нхсінеї". In the settings, "Зашифрувати" is selected under "Команда", "Українська" is selected under "Мова", and "Лінійне рівняння" is selected under "Тип". Additionally, there are input fields for "А:" with the value "2" and "В:" with the value "3". The "Виконати" button is highlighted with a blue border.

Лінійне дешифрування українською мовою:

МОК

Файл Атака Частотні таблиці Шифрування Про розробника Допомога Вихід

До: нхєінеї

Після: крavecь

Команда

☐ Зашифрувати

☒ Розшифрувати

Мова

☒ Українська

☐ English

Тип

☒ Лінійне рівняння

☐ Нелінійне рівняння

☐ Гасло

А: 2

В: 3

Виконати

Лінійне шифрування англійською мовою:

МОК

Файл Атака Частотні таблиці Шифрування Про розробника Допомога Вихід

До: kravets

Після: qсuсstmk

Команда

☒ Зашифрувати

☐ Розшифрувати

Мова

☐ Українська

☒ English

Тип

☐ Лінійне рівняння

☒ Нелінійне рівняння

☐ Гасло

А: 2

В: 3

С: 6

Виконати

Лінійне дешифрування англійською мовою:

МОК

Файл Атака Частотні таблиці Шифрування Про розробника Допомога Вихід

До: qсuсstmk

Після: kravets

Команда

☐ Зашифрувати

☒ Розшифрувати

Мова

☐ Українська

☒ English

Тип

☐ Лінійне рівняння

☒ Нелінійне рівняння

☐ Гасло

А: 2

В: 3

С: 6

Виконати

Шифрування гаслом:

МОК

Файл Атака Частотні таблиці Шифрування Про розробника Допомога Вихід

До:

Після:

Команда

☒ Зашифрувати

☐ Розшифрувати

Мова

☐ Українська

☒ English

Тип

☐ Лінійне рівняння

☐ Нелінійне рівняння

☒ Гасло

Гасло:

Виконати

Дешифрування гаслом:

МОК

Файл Атака Частотні таблиці Шифрування Про розробника Допомога Вихід

До:

Після:

Команда

☐ Зашифрувати

☒ Розшифрувати

Мова

☐ Українська

☒ English

Тип

☐ Лінійне рівняння

☐ Нелінійне рівняння

☒ Гасло

Гасло:

Виконати

Лінійна атака:

МОК

Файл Атака Частотні таблиці Шифрування Про розробника Допомога Вихід

До:

Після:

Команда

☒ Зашифрувати

☐ Розшифрувати

Мова

☐ Українська

☒ English

Тип

☒ Лінійне рівняння

☐ Нелінійне рівняння

☐ Гасло

А:

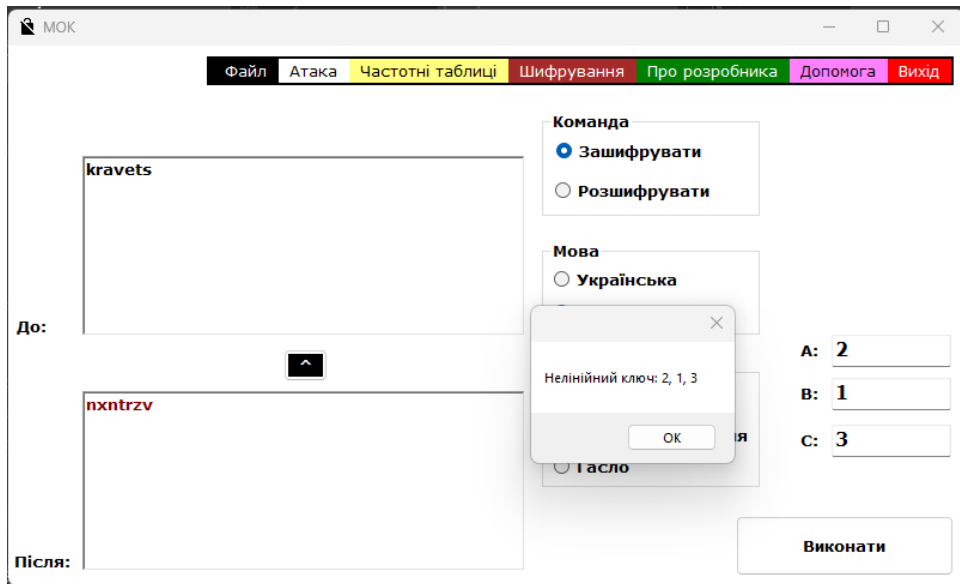
В:

Виконати

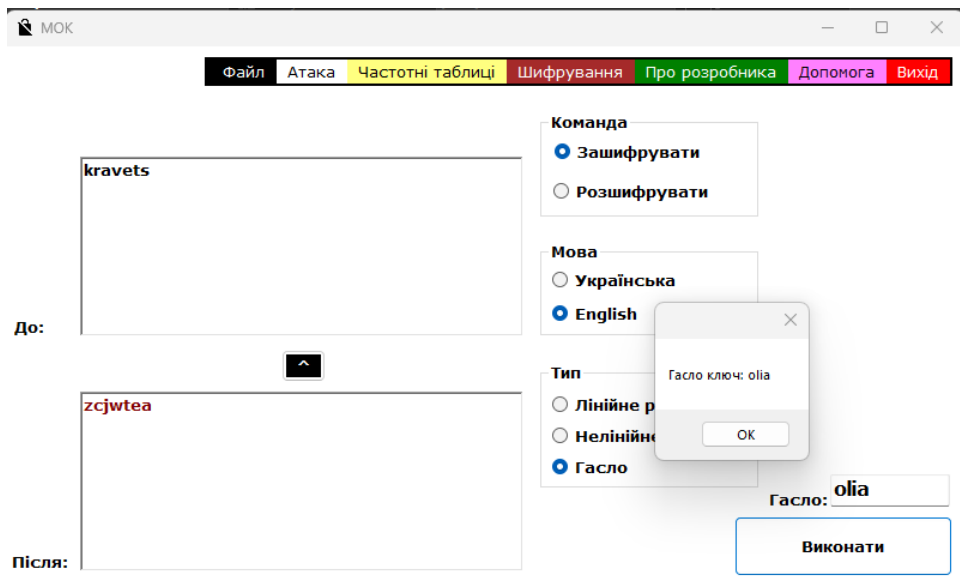
Лінійний ключ: 2, 3

ОК

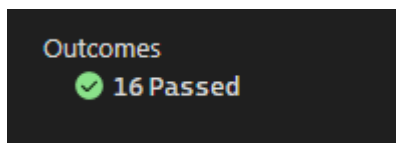
Нелінійна атака:



Атака гаслом:



Тестування програми юніт тестами:



[TestClass()]

0 references

public class TrithemiusTests

{

Trithemius t = new Trithemius();

[TestMethod()]

0 references

public void TestEncryptLinearAB()

{

string input = "kravets";

string expected = "nwhepgh";

int step1 = 2;

int step2 = 3;

string output = t.EncryptLinearAB(input, step1, step2);

Assert.AreEqual(expected, output);

}

[TestMethod()]

0 references

public void TestUKREncryptLinearAB()

{

string input = "кравець";

string expected = "нхєінеї";

int step1 = 2;

int step2 = 3;

string output = t.UKREncryptLinearAB(input, step1, step2);

Assert.AreEqual(expected, output);

}

[TestMethod()]

0 references

public void TestDecryptLinearAB()

{

string input = "nwhepgh";

string expected = "kravets";

int step1 = 2;

int step2 = 3;

string output = t.DecryptLinearAB(input, step1, step2);

Assert.AreEqual(expected, output);

}

[TestMethod()]

0 references

public void TestUKRDecryptLinearAB()

{

string input = "нхєінеї";

string expected = "кравець";

int step1 = 2;

int step2 = 3;

string output = t.UKRDecryptLinearAB(input, step1, step2);

Assert.AreEqual(expected, output);

}

```

[TestMethod()]
0 references
public void TestEncryptNonLinearABC()
{
    string input = "kravets";
    string expected = "oasaaki";

    int step1 = 2;
    int step2 = 3;
    int step3 = 4;

    string output = t.EncryptNonLinearABC(input, step1, step2, step3);
    Assert.AreEqual(expected, output);
}

[TestMethod()]
0 references
public void TestUKREncryptNonLinearABC()
{
    string input = "кравець";
    string expected = "ооаоасцх";

    int step1 = 2;
    int step2 = 3;
    int step3 = 4;

    string output = t.UKREncryptNonLinearABC(input, step1, step2, step3);
    Assert.AreEqual(expected, output);
}

```

```

[TestMethod()]
0 references
public void TestDecryptNonLinearABC()
{
    string input = "oasaaki";
    string expected = "kravets";

    int step1 = 2;
    int step2 = 3;
    int step3 = 4;

    string output = t.DecryptNonLinearABC(input, step1, step2, step3);
    Assert.AreEqual(expected, output);
}

[TestMethod()]
0 references
public void TestUKRDecryptNonLinearABC()
{
    string input = "ооаоасцх";
    string expected = "кравець";

    int step1 = 2;
    int step2 = 3;
    int step3 = 4;

    string output = t.UKRDecryptNonLinearABC(input, step1, step2, step3);
    Assert.AreEqual(expected, output);
}

```

```
[TestMethod()]
```

```
● | 0 references
```

```
public void TestEncryptMotto()
```

```
{
```

```
    string input = "kravets";
```

```
    string pass = "olia";
```

```
    string expected = "zcjwtea";
```

```
    string output = t.EncryptMotto(input, pass);
```

```
    Assert.AreEqual(expected, output);
```

```
}
```

```
[TestMethod()]
```

```
● | 0 references
```

```
public void TestDecryptMotto()
```

```
{
```

```
    string input = "zcjwtea";
```

```
    string pass = "olia";
```

```
    string expected = "kravets";
```

```
    string output = t.DecryptMotto(input, pass);
```

```
    Assert.AreEqual(expected, output);
```

```
}
```

```
}
```