Systems and Networks Security

LAB 4

Main Points

- **Substitution:**
  - **Poly-Alphabetic.**
  - **Simple Shift Vigenere Cipher.**
  - **Examples.**


طرق التعمية التقليدية
الإحلال أو الاستبدال (Substitution)
الإبدال أو التبديل (Transposition)
الأبجدية الأحادية (Mono-alphabetic)
أبجدية بولي (Poly-alphabetic)
الإبدال أو التبديل (Transposition)
1. شيفرة قيصر(الجمع).
2. شيفرة الضرب.
3. الشيفرة المختلطة.
4. أحادية الاستبدال.
5. المفتاح التلقائي.
6. شيفرة بلا فير.
7. شيفرة فجينير.
8. شيفرة العكس.
9. شيفرة سكة الحديد (ريل فينس).
10. تبديل الصفوف.
2

Substitution

- **شفرة فجينير.**

النص المشفر = (الحرف الاولى من النص الاصلي + قيمة الحرف الاولى من المفتاح) بتكرار قيمة مفتاح التشفير حتى انتهاء من النص الأصلي.

- **فك شفرة فجينير.**

النص الاصلي = (الحرف الاولى من النص المشفر – قيمة الحرف الاولى من المفتاح) بتكرار قيمة مفتاح التشفير حتى انتهاء من النص المشفر.

Plaintext →	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext →	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Value →	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Example “1”

Original Text (Key = ONE) :

MYNAME

Cipher Text = ????

Solution "1"

Original Text	M	Y	N	A	M	E
Original Text Value	12	24	13	0	12	4
Key	O	N	E	O	N	E
Key Value	14	13	4	14	13	4
Original Text Value + Key Value	26	37	17	14	25	8
Cipher Text	A	L	R	O	Z	I

Key = ONE

ALROZI

Plaintext →	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext →	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Value →	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Cipher Text	A	L	R	O	Z	I
Cipher Text Value	0	11	17	14	25	8
Key	O	N	E	O	N	E
Key Value	14	13	4	14	13	4
Cipher Text Value - Key Value	-14	-2	13	0	12	4
Original Text	M	Y	N	A	M	E

Key = ONE

MYNAME

Plaintext →	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext →	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Value →	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Example “2”

Original Text (Key = LAB) :

COMPUTER

Cipher Text = ????

Solution "2"

Original Text	C	O	M	P	U	T	E	R
Original Text Value	2	14	12	15	20	19	4	17
Key	L	A	B	L	A	B	L	A
Key Value	11	0	1	11	0	1	11	0
Original Text Value + Key Value	13	14	13	26	20	20	15	17
Cipher Text	N	O	N	A	U	U	P	R

Key = LAB

NONAUUPR

Plaintext →	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext →	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Value →	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Cipher Text	N	O	N	A	U	U	P	R
Cipher Text Value	13	14	13	0	20	20	15	17
Key	L	A	B	L	A	B	L	A
Key Value	11	0	1	11	0	1	11	0
Cipher Text Value - Key Value	2	14	12	-11	20	19	4	17
Original Text	C	O	M	P	U	T	E	R

Key = LAB

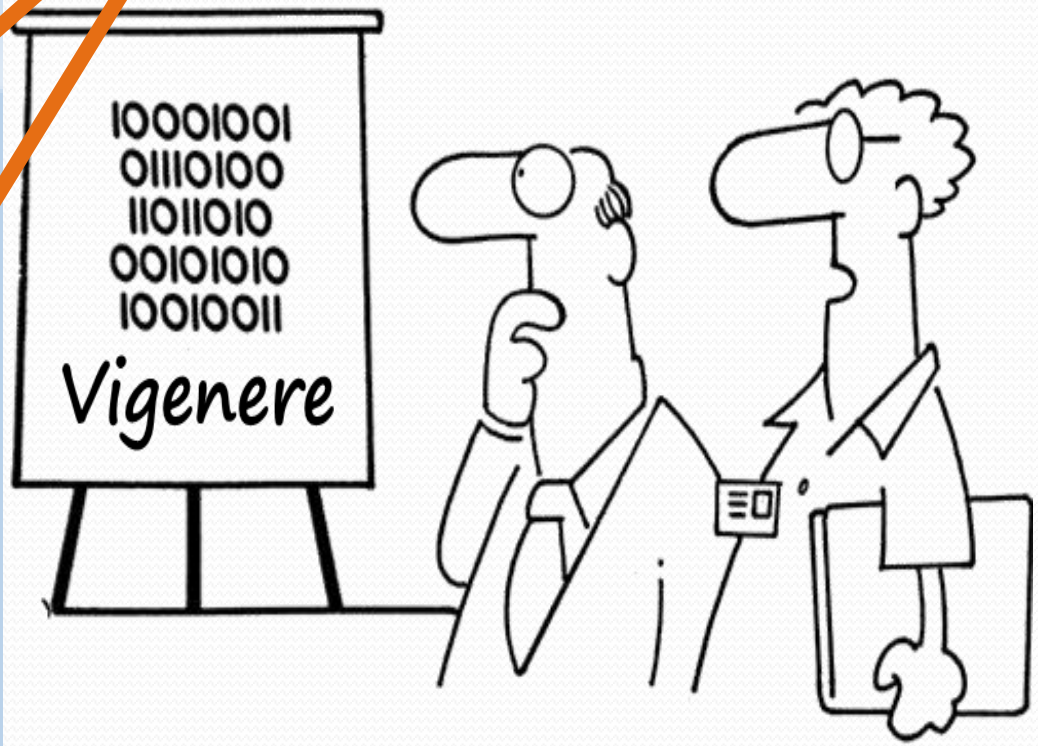
COMPUTER

Plaintext →	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext →	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Value →	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Label

Button

Text Box



1

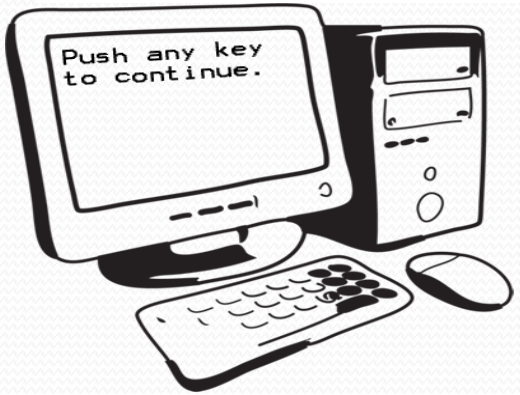
```
int tovalue(char m)
{  switch (m)
{ case 'A': return 0; break;
case 'B': return 1; break;
case 'C': return 2; break;
case 'D': return 3; break;
case 'E': return 4; break;
case 'F': return 5; break;
case 'G': return 6; break;
case 'H': return 7; break;
case 'I': return 8; break;
case 'J': return 9; break;
case 'K': return 10; break;
case 'L': return 11; break;
case 'M': return 12; break;
case 'N': return 13; break;
case 'O': return 14; break;
case 'P': return 15; break;
case 'Q': return 16; break;
case 'R': return 17; break;
case 'S': return 18; break;
case 'T': return 19; break;
case 'U': return 20; break;
case 'V': return 21; break;
case 'W': return 22; break;
case 'X': return 23; break;
case 'Y': return 24; break;
case 'Z': return 25; break;
default: return 26;
} }
```

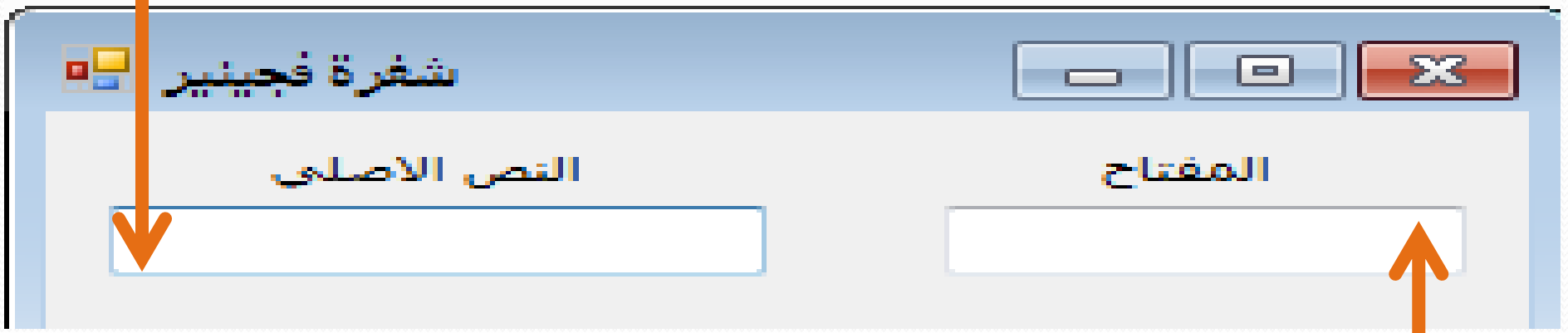
```
char tochar(int m)
{  switch (m)
{ case 0: return 'A'; break;
case 1: return 'B'; break;
case 2: return 'C'; break;
case 3: return 'D'; break;
case 4: return 'E'; break;
case 5: return 'F'; break;
case 6: return 'G'; break;
case 7: return 'H'; break;
case 8: return 'I'; break;
case 9: return 'J'; break;
case 10: return 'K'; break;
case 11: return 'L'; break;
case 12: return 'M'; break;
case 13: return 'N'; break;
```

2

```
case 14: return 'O'; break;
case 15: return 'P'; break;
case 16: return 'Q'; break;
case 17: return 'R'; break;
case 18: return 'S'; break;
case 19: return 'T'; break;
case 20: return 'U'; break;
case 21: return 'V'; break;
case 22: return 'W'; break;
case 23: return 'X'; break;
case 24: return 'Y'; break;
case 25: return 'Z'; break;
default: return ' ';
}}
```

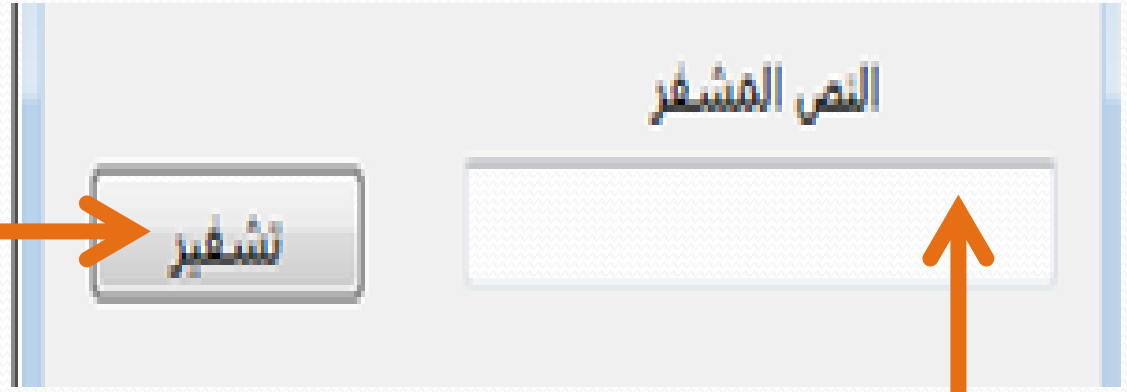
```
private void textBox1_TextChanged(object sender, EventArgs e)
{
textBox1.CharacterCasing = CharacterCasing.Upper;
}
```






```
private void textBox2_TextChanged(object sender, EventArgs e)
{
textBox2.CharacterCasing = CharacterCasing.Upper;
}
```

```
private void button1_Click(object sender, EventArgs e)
{
string t2 = textBox2.Text;
do{
textBox2.Text = textBox2.Text + t2;}
while (textBox2.Text.Length <= textBox1.Text.Length);
        int m1 = textBox1.Text.Length;
        int m2 = textBox2.Text.Length;
        string yo = textBox2.Text.Remove(m1);
        textBox2.Text = yo;
textBox3.Text = en(textBox1.Text, textBox2.Text);
}
```

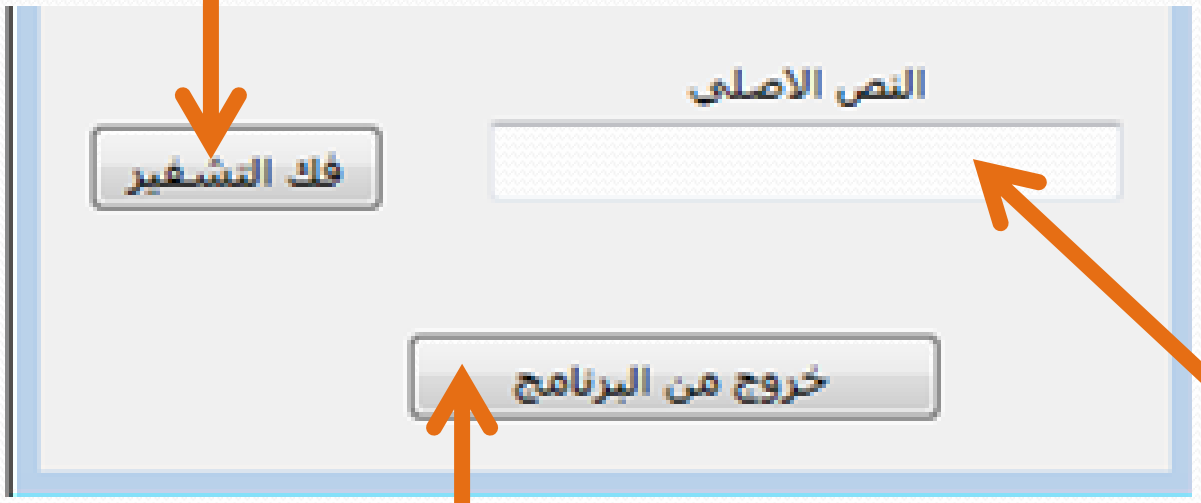


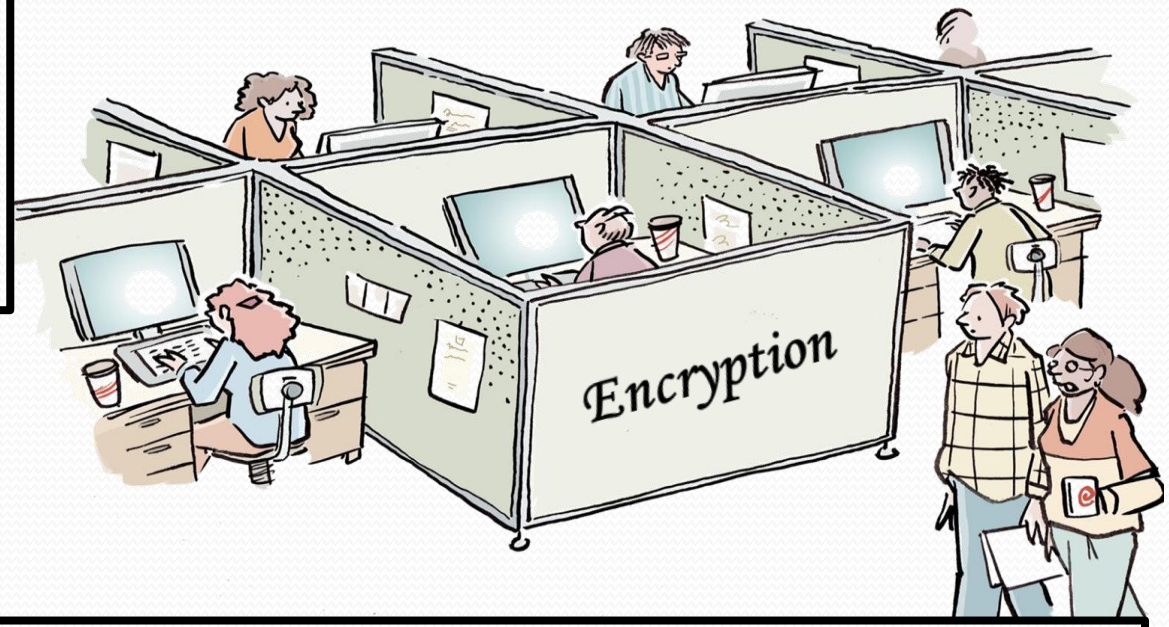

```
private void textBox3_TextChanged(object sender, EventArgs e)
                                    {  }
```

3

```
string en(string a, string b)
{
        string d = "";
        int x = 0;
        for (int i = 0; i < a.Length; i++)
        {
            x = tovalue(a[i]) + tovalue(b[i]);
            if (x > 25) x = x - 26;
            d = d + tochar(x);
        }
        return d;
}
```

```
private void button2_Click(object sender, EventArgs e)
{
textBox4.Text = de(textBox3.Text, textBox2.Text);
}
```






```
private void textBox4_TextChanged(object sender,
EventArgs e)
                                        { }
```

```
private void button3_Click(object sender, EventArgs e)
                { Application.Exit(); }
```

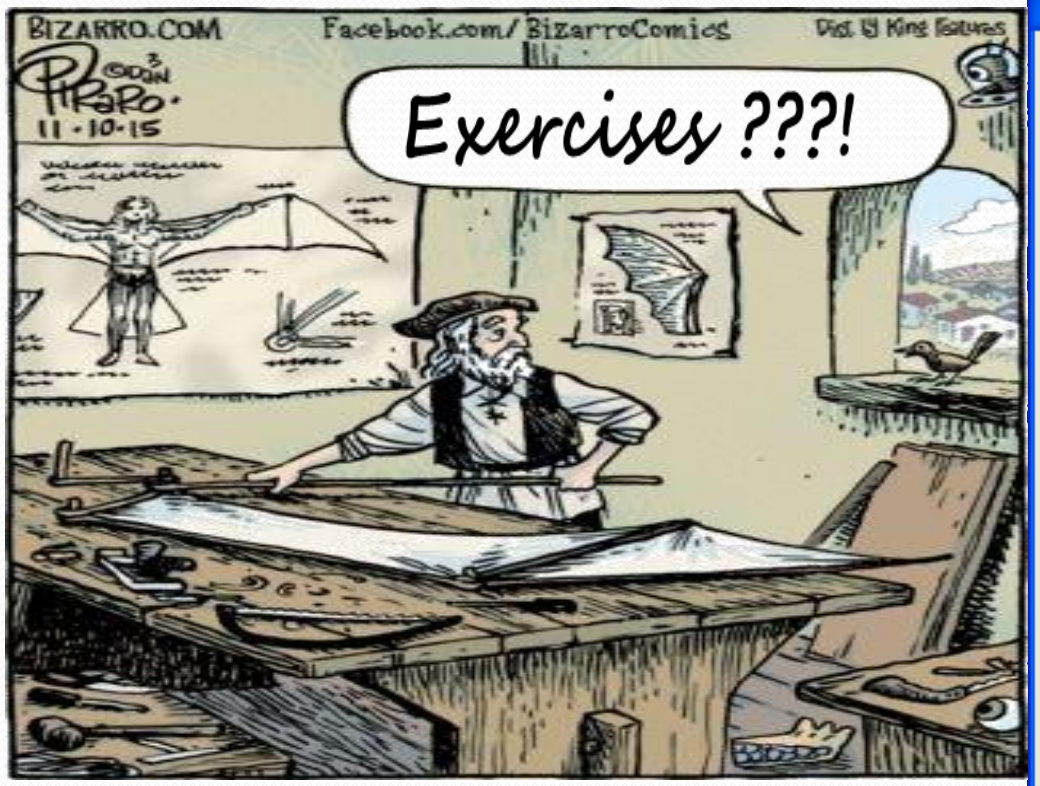
4

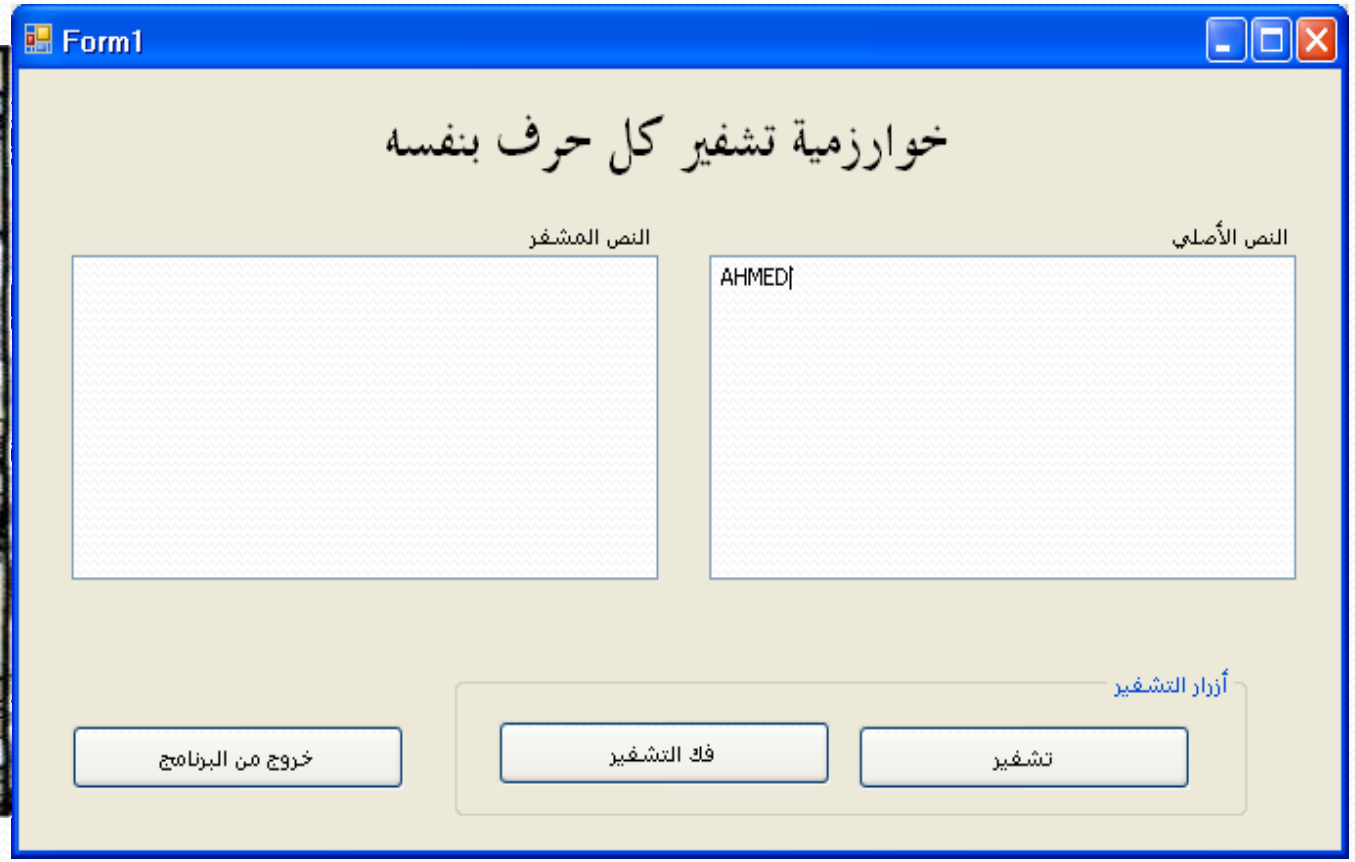

```
string de(string a, string b)
{
        string d = "";
         int x = 0;
         for (int i = 0; i < a.Length; i++)
         {
            x = tovalue(a[i]) - tovalue(b[i]);
            if (x < 0) x = x + 26;
            d = d + tochar(x);
         }
         return d;
}
```


شفرة فجينير
المفتاح
ONEONE
النص الاصلي
MYNAME
النص المشفر
ALROZI
تشفير
النص الاصلي
MYNAME
فك التشفير
خروج من البرنامج


شفرة فجينير
المفتاح
LABLABLA
النص الاصلي
COMPUTER
النص المشفر
NONAUUPR
تشفير
النص الاصلي
COMPUTER
فك التشفير
خروج من البرنامج

- اكتب برنامج لتنفيذ الخوارزمية تقوم على أساس تشفير كل حرف بنفسه وفق خوارزمية الجمع (قيصر) المعروفة بحيث يتم تشفير كل حرف بنفسه وفق الخوارزمية المذكورة آنفاً. فمثلاً عند تشفير كلمة AHMED سيكون الناتج AOYIG وهكذا كلمة HUSSEIN ستكون OOKKIQA ؟





Thank you