Homework-1

Bir Mesajın Anahtarla Şifreleme ve Deşifreleme Sistemi

※ Sistemimiz kullancıdan bir mesaj ve anahtar alıp **Vigenere Sifreleme** yöntemi ile mes aji şifrelenmektedir, aynı anahtarla deşifremleme yapmaktadır.

Vigenere Cipher : $P = C = K = (Z_{26})^a$; a sabit bir tamsayi.

Ex: encrypyion dncrypyion

P	V	I	G	E	N	E	R	E	21 8 6 4 13 4 17 4
K	K	E	Y	K	E	Y	K	E	10 4 24 10 4 24 10 4
P + K ≡ C MOD(26)									5 12 4 14 17 2 1 8

Şifreli mesaj: F M E O R B A I

Deşifreleme yapmak için anahtarla:

C - K ≡ P MOD(26)

※ Sistemin Yapılışı, C# programlama dili kullanılarak masaüstü uygulaması (Windows form) üç temel fonksiyondan oluşmaktadır.

1-Bir String alıp Tamsayı dizisi döndürüyor (getCharIndex) : Mesajın karakterleri gezip ve her karaktarin index karşılığı değer diziye eklemektedir.

```
public static string alphaChars = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ ().,?!";

// Bir string alıp, bir tamsayı dizisi döndürür.
public int[] getCharIndex(string txt)
{
    int size = txt.Length;
    int[] index = new int[size];

    for (int i = 0; i < size; i++)
    {
        index[i] += alphaChars.IndexOf(txt[i]); // index'leri diziye ekle.
    }
    return index;
}
```

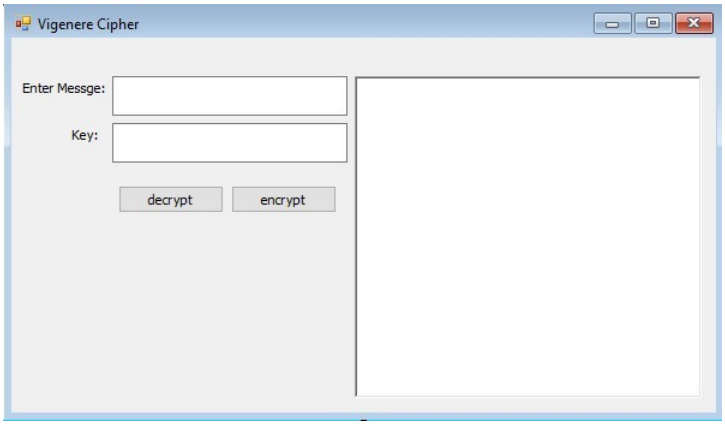
2-Şifreleme işlemi (encryption) : Mesaj ve Anahtar string cinsinden alıp şifreli mesajı string cinsinden döndüren bir fonkisitondur. Alınan mesaj ve anahtar **getCharIndex** fonksiyonu kullanılark index değer dizileri oluşturduktan sonra iki dizinin toplayıp ve vigenenere şifreleme işlemleri uygulanmaktadır, şifreli mesajı elde edilir.

```
// Şifreleme Fonksiyonu
public string encryption(string message, string key)
{
    int[] messageCharIndex = getCharIndex(message);  // mesajın index'ler dizisi
    int[] keyCharIndex = getCharIndex(key);          // Anahtar index'ler dizisi
    string cipher = "";

    int messageLength = message.Length;
    int keyLength = key.Length;
    int temp;

    for (int n = 0; n < messageLength; n++)
    {   // P + K = C mod(26), 26 alfabe için. Bu sistem büyük / küçük harfleri ve temel karakterleri içeriyor.
        temp = (messageCharIndex[n] + keyCharIndex[n % keyLength]) % alphaChars.Length;
                                                // anahtar uzunluğu yetmediği durumda tekrarlansın
        cipher += alphaChars[temp];
    }
    return cipher;
}
```

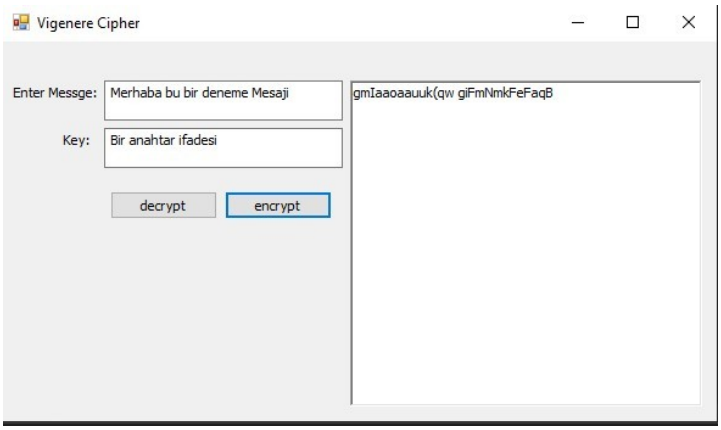
3-Difreleme işlemi (decryption): Deşifreleme fonksiyonu, şifreleme işlemi (**encryption**) hemen henem aynı fakat dizileri toplamak yerine çıkartma işlemi yapılmaktadır

```
// Deşifreleme Fonksiyonu
public string decryption(string cipher, string key)
{
    int[] cipherCharIndex = getCharIndex(cipher);
    int[] keyCharIndex = getCharIndex(key);

    int cipherLength = cipher.Ldecryptionength;
    int keyLength = key.Length;
    string message = "";
    int temp;

    for (int n = 0; n < cipherLength; n++)
    {   // C - K = P mod(26)
        temp = (cipherCharIndex[n] - keyCharIndex[n % keyLength] + alphaChars.Length) % alphaChars.Length;
        message += alphaChars[temp];
    }
    return message;
}
```

※ TASARIM VE UYGULAMA :

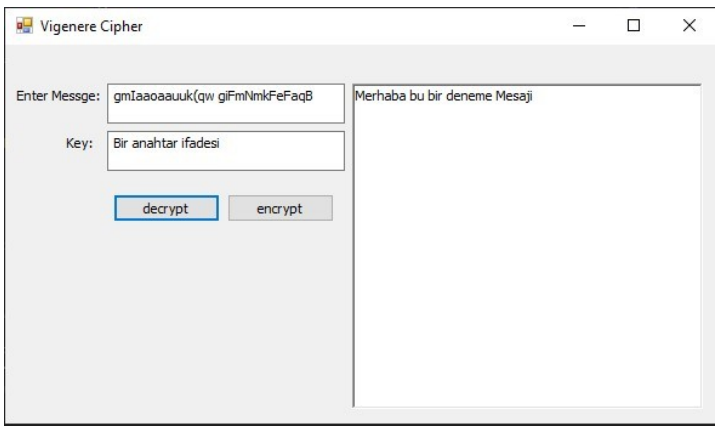


※ ŞİFRELEME :



NOT: Bu uyğulama türkçe karakterleri içermemektedir.

※ DEŞİFRELEME :



Github: https://github.com/Ctaljibini/Bilgi-G-venli-ine-Giri-/tree/main/Homework-1

Cuma Taljibini - 20060905