KRİPTOGRAFİ TEMELLERİ

PROJE 1

\*Furkan Kaya Azat Turunç

191216002 - 191216054

## 

## ÖĞRETİM GÖREVLİSİ Ahmad Hasan Abed Al Khan

# Giriş:

### Programı çalıştırma:

Main.py dosyasının bulunduğu dizininde terminalinizi açıp python -linux: python3- .\main.py yazarak programı çalıştırabilirsiniz.

**-NOT: Python çalıştırma ortamınızda pandas ve numpy kütüphaneleri bulunmalıdır.-**

### Programın işleyişi:

Metin Al ->

Metin karakterlerini unicode biçiminde listeye al ->

Listedeki elemanları ikilik sisteme dönüştür ->

Binary metne dönüştürülmüş listeyi 256’lık parçalara böl ->

**-ANAHTAR OLUŞTURMA-**

Rastgele olarak 4 elemanlı 1. anahtarı oluştur ->

Rastgele olarak 8 elemanlı 3. anahtarı oluştur ->

Rastgele 128 bit oluştur ve bunu tersine çevirip 256 bit’e tamamla ->

Oluşturulan 256 bit’i anahatar2’ye ata ->

**-ŞİFRELEME ADIMLARI-**

Eksik kalırsa son bloğun başına gerekli kadar 0 ekle ->

Her bir bloğu tersine çeviri (flip) ->

Bu anahtardaki duruma göre her bir bloğu karıştır ->

Anahtar2’yi kullanarak tüm 256 bitlik blokları XOR’la ->

Bu anahtardaki duruma göre her bir bloğu karıştır ->

Çıkan sonucu ekrana bas ->

**-ŞİFRE ÇÖZME ADIMLARI-**

Yukarıda oluşan şifreli binary’i

Şifrelemedeki adımları tersten başlayarak tekrar yap ->

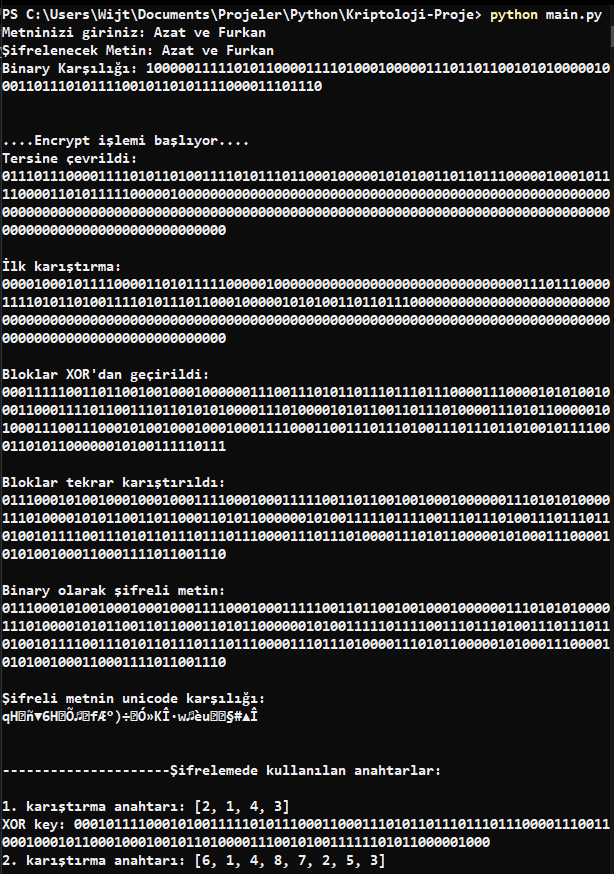
Çıkan sonucu referans bit uzunluğu ile bölümle->

Oluşan sayıları unicode karşılığını listeye al-

Listeyi (şifresiz metni) ekrana bas->

## Programın ekran görüntüleri:

### Şifreleme:



### Şifre çözme:

# Sorular:

## Bu sistemin ait olduğu çalışma modunun adı nedir?

Programın çalışma modu ECB’dir. CBC modunda paralelleştirme yapılamaz çünkü blokların ardı ardına şifrelenmesi gerekir. Bizden istenen algoritmada ise 256 bloklar oluşturulan key ile teker teker şifrelenmektedir. Bu key 1 defa oluşturup daha sonrasında paralelleştirme yapılabilir.

## 1. Algoritma size göre standartlaştırılacak kadar güvenli mi? Neden?

Bizce güvenlidir. Her şifreleme algoritması eninde sonunda kırılabilir. Kıralamayacağını düşünerek değil ne zaman kırılabileceğini düşünerek hareket etmeliyiz. Elimizde 3 farklı anahtar var bunların her birinin rastgele bir şekilde bulunma ihtimali çok düşüktür.

## 2. Sistemin savunmasız kalmasına neden olabilecek saldırı türleri nelerdir?

Her metin rastgele oluşturulan kendine has anahtarlar ile şifrelenmez ise aşağıdaki yöntemler kullanılabilir.

1. Seçilmiş düz metin saldırıları
2. Kriptanalitik yazılım
3. XOR keyine brute force yapılabilir 128 in permitasyonu gibi bir karmaşılığı vardır. Bu bulunsa dahi dönüştürme yapıldığında karışık bir metin ortaya çıkacaktır.

## 3. Algoritma size göre standartlaştırılacak kadar güvenli mi? Neden?

Güvenliğin temel hedeflerini sağlamaktadır. Dolayısıyla bizce standartlaştırılabilir.

## 4. Algoritma, güvenliğin temel hedeflerini sağlıyor mu? (Gizlilik, Bütünlük, Erişilebilirlik)

1. ISO/IEC 27000 BGYS sözlük standardında “gizlilik” ; “Bilginin yetkisiz kişiler, varlıklar veya prosesler için elverişli yapılmaması ya da açıklanmaması özelliği” olarak tanımlanmıştır. Algoritma, bilgiyi şifreleyerek yetkisiz kişilerin bilgiye erişimini engeller. Gizlilik ilkesi sağlanmıştır.
2. ISO/IEC 27000 BGYS sözlük standardında “bütünlük”, “Varlıkların doğruluğunu ve tamlığını koruma özelliği” olarak tanımlanır. Veri kaynaktan çıktığı haliyle alıcıya aktarılır. Şifrelenmiş bilgi bozulmaya uğramaz. Bütünlük ilkesi sağlanmıştır.
3. ISO/IEC 27000 BGYS sözlük standardında “erişilebilirlik”, “Yetkili bir tüzel kişilik tarafından talep edildiğinde erişilebilir ve kullanılabilir olma özelliği” olarak tanımlanır. Saldırıların yapılma amacı erişebilirliği düşürmektir. Girilen veri şifrelenebilir ve anahtarlar ile şifrelenmiş veri deşifre edilebilir. Erişilebilirlik ilkesi sağlanmıştır.

## 5. Şifreli metnin şifresini çözdükten sonra orijinal düz metni nasıl elde edebiliriz? Çözülmez ise sebebi nedir?

Şifreli metnin şifresini çözünce elimize 2’lik sistemde uzun bir metin geçmektedir bu metnin referans alınmış byte uzunluğu bilindiği takdirde orjinal haline çevrilebilir. Ya da sırayla olası byte uzunlukları denenip döndürülebilir.

## 6. Algoritmanın zayıf yönleri nelerdir? Bunları nasıl çözülmelidir?

ECB modunda yapılan şifreleme bloğu değiştirse dahi aynı bloklar girdiğinde aynı sonucu üretmektedir bu da güvenlik açığı ortaya çıkarmaktadır. Biz is şifrelemeye sokmadan önce ve sonrasında blokların sırasını karıştırarak buna çözüm getirdik. Giriş blokları merhaba merhaba bile olsa çıktığında tamamen rastgele şekilde oluşan sıra ile şifreye girip tamamen rastgele oluşan sıra ile kullanıcıya sunulmaktadır. **Rastgele oluşan bu anahtarlar bilinmeden de geriye dönüş mümkün değildir.** -en azından çok uzun bir süre için-

## 7. Lütfen şifrelenmiş fotoğrafı görüntüleyin ve orijinal olanlarla karşılaştırmak için şifreli metni yazdırın. Lütfen şifrelenmiş fotoğrafı görüntüleyin ve orijinal olanlarla karşılaştırmak için şifreli metni yazdırın.

Programın ekran görüntüleri üst bölümde verilmiştir.

**Şifresiz metin:**

Furkan ve Azat

**Şifrelemede kullanılan anahtarlar:**

1. karıştırma anahtarı:

[2, 1, 3, 4]

XOR key:

00010001001110100000110001111101100010110010001110101001110000000001001111000001011111101010100101100001011000011010010100000000

2. karıştırma anahtarı:

[8, 6, 7, 5, 2, 4, 1, 3]

**Şifreli metin:**

ZÿtÜV?ì>VîÅóË#©À5º'w▲`v<Ì☻¹