T.C.
DÜZCE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

GÜVENLİ SİBER İLETİŞİM AMACIYLA WEB ADRESLERİ ÜZERİNDEN YENİ BİR STEGANOGRAFİK YAKLAŞIM

YÜKSEK LİSANS TEZİ

OĞUZHAN KENDİRLİ

ARALIK 2013

DÜZCE

KABUL VE ONAY BELGESİ

Oğuzhan KENDİRLİ tarafından hazırlanan *"Güvenli Siber İletişim Amacıyla Web Adresleri Üzerinden Yeni Bir Steganografik Yaklaşım"* isimli lisansüstü tez çalışması, Düzce Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 23.12.2013 tarih ve 1326 sayılı kararı ile oluşturulan jüri tarafından Bilgisayar Mühendisliği Anabilim Dalı'nda Yüksek Lisans Tezi olarak kabul edilmiştir.

Yrd. Doç. Dr. Esra ŞATIR
(Tez Danışmanı)
Düzce Üniversitesi

Doç. Dr. Resul KARA
Düzce Üniversitesi

Yrd. Doç. Dr. Selman KULAÇ
Düzce Üniversitesi

Doç. Dr. Pakize ERDOĞMUŞ
Düzce Üniversitesi

Yrd. Doç. Dr. İkrime Orkan UÇAR
Düzce Üniversitesi

Tezin Savunulduğu Tarih: 31.12.2013

ONAY

Bu tez ile Düzce Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu Oğuzhan KENDİRLİ'nin Bilgisayar Mühendisliği Anabilim Dalı'nda Yüksek Lisans derecesini almasını onamıştır.

Prof. Dr. Haldun MÜDERRİSOĞLU
Fen Bilimleri Enstitüsü Müdürü

BEYAN

Bu tez çalışmasının kendi çalışmam olduğunu, tezin planlanmasından yazımına kadar bütün aşamalarda etik dışı davranışımın olmadığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara kaynak gösterdiğimi ve bu kaynakları da kaynaklar listesine aldığımı, yine bu tezin çalışılması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını beyan ederim.

31.12.2013

(İmza)

Oğuzhan KENDİRLİ

Sevgili Aileme

TEŞEKKÜR

Yüksek lisans öğrenimim ve bu tezin hazırlanmasında süresince gösterdiği her türlü destek ve yardımdan dolayı çok değerli hocam Yrd. Doç. Dr. Esra ŞATIR'a en içten dileklerimle teşekkür ederim.

Tez çalışmam boyunca değerli katkılarını esirgemeyen Doç. Dr. Resul KARA'ya şükranlarımı sunarım.

Bu çalışma boyunca yardımlarını ve desteklerini esirgemeyen sevgili aileme ve arkadaşlarıma sonsuz teşekkürlerimi sunarım.

31 Aralık 2013 **Oğuzhan KENDİRLİ**

İÇİNDEKİLER **Sayfa**

ŞEKİL LİSTESİ

.

ÇİZELGE LİSTESİ

SİMGELER VE KISALTMALAR

AES	Advanced encryption standard - Gelişmiş şifreleme standardı
ASCII	American standard code for information interchange Bilgi dönüşümü için Amerikan standart kodlama sistemi
BMP	Bitmap
C#	C sharp
CAST128	Carlisle Adams - Stafford Tavares - 128
DCT	Discrete cosine transform - Ayrık kosinüs dönüşümü
DES	Data encryption standard - Veri şifreleme standardı
DSP	Digital signal processing - Sayısal işaret işleme
EXE	Executable file - Çalıştırılabilir dosya
FIPS	Federal information processing standard Federal bilgi işleme standartları
FP	Final Permutation – Çıkış permütasyonu
GF	Galois Alanı
GIF	Graphics interchange format - Grafik değişim biçimi
HTML	Hypertext markup language - Hipermetin işaretleme dili
IP	Initial Permutation - Başlangıç permütasyonu
IP	Internet protocol address - İnternet protokol
JPEG	Joint photographic experts group - Birleşik fotoğraf uzmanları grubu
LSB	Least significant bit - en az duyarlı bit
LZW	Lempel–Ziv–Welch
MARS	Multiplication, addition, rotation and substitution Çarpma, toplama, rotasyon ve değişiklik
MSB	Most significant bit - En duyarlı bit
MSE	Mean Square Error – Karesel hata
NIST	National institute of standards and technology Ulusal standartlar ve teknoloji enstitüsü
OCR	Optical Character Recognition - Optik Karakter Tanıma
PNG	Portable network graphics - Taşınabilir ağ grafiği

PSNR	Peak signal to noise ratio – Tepe sinyal gürültü oranı
RC5	Rivest cipher 5
RC6	Rivest cipher 6
RSA	Rivest, Shamir, Adleman
S-Box	Substitution-box - Değiştirme kutusu
URL	Uniform resource locator - Tek tip kaynak konumlayıcı
XML	Extensible markup language - Genişleyebilir işaretleme dili
XOR	Exclusive or - Özel veya
&nbps	Html kodunda boşluk oluşturma kodu
$\Delta(D)$	Kısa kaynak bilgi
[.]	Yuvarlama operatörü
$\Phi(n)$	Ortak bölen
A	Örnek alfabe
A	S içindeki karakterlerin ASCII karşılıkları
A(x)	Polinom
B	Kaydedici
C	Konum bilgilerini içeren dizi
C	Taşıyıcı nesne
C	Sıkıştırma oranı
C'	Stego görüntü
C'	Şifrelenmiş C dizisi
d	Özel anahtar
db	Desibel
E	Şifreleme sonucu elde edilen matris
e	Açık anahtar
E_m	Gömme aşaması
E_x	Çıkarım aşaması
F	Giriş görüntüsü
F	Derecelendirme
I	Görüntü tabanındaki her bir görüntü
i	Piksel değeri
j	Piksel değeri
K	Seçimlik anahtar

K(x)	Anahtar değerleri
L	Kaydedici
m	Değişken
M	Giriş görüntü boyutları
M	Gönderilecek mesaj
M	Açık metin
$M(x_i)$	Etiket
n	Değişken
N	Giriş görüntü boyutları
n	Yeni üye
N	Asal sayıların çarpımı
N_b	Durum uzunluğu
N_k	Anahtar uzunluğu
N_r	Tekrarlanan tur sayısı
p	Asal sayı
P	Açık metin
P	Örüntü kümesi
Q	Kalite faktörü
q	Asal sayı
R	Sıkıştırma miktarı
S	Gizli mesaj
S_c	Sıkıştırılan dosya boyutu
S_o	Orijinal dosya boyutu
T	Şifrelenmiş metin
T	Çıkış görüntüsü
U	Sıkıştırma sonucu elde edilen matris
X	Orijinal görüntü
x	Sıkıştırma sonucu karakter sayısı
x	Görüntü koordinatları
X'	Gömme aşaması sonucu elde edilen görüntü
x’	Şifreleme sonucu karakter sayısı
y	Görüntü koordinatları

ÖZET

GÜVENLİ SİBER İLETİŞİM AMACIYLA WEB ADRESLERİ ÜZERİNDEN YENİ BİR STEGANOGRAFİK YAKLAŞIM

Oğuzhan KENDİRLİ
Düzce Üniversitesi
Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı
Yüksek Lisans Tezi
Danışman: Yrd. Doç. Dr. Esra ŞATIR
Aralık 2013, 100 sayfa

Bilgi teknolojilerinin ve internetin hızlı büyümesi ile güvenlik kritik bir konu haline gelmiştir. Bu nedenle veri gizleme, gizli mesajların iletimi için önem kazanmıştır. Veri gizleme teknikleri video, görüntü, metin gibi gizli mesajları saklamaktadır. Veri gizleme tekniği kriptolojiden farklıdır. Bir kriptoloji şeması, mesajı şifreler ve mesaj alıcı tarafına daha güvenli ve içeriği tahmin edilemez bir şekilde gönderilir. Mesaj anlamsız bir içeriğe sahip olduğundan, iletişim esnasında gözlemci, mesaj değiş tokuşunu fark edebilmektedir. Bu nedenle kötü niyetli bir saldırgan tarafından tehdit gelme olasılığı kriptolojide her zaman vardır. Steganografi, hedeflenen alıcı dışında kimsenin gizli mesajın varlığı hakkında herhangi bir şey bilmediği şekilde iletişimi sağlama tekniğidir. Başarılı bir steganografi, taşıyıcı ortamın şüphe uyandırmamasına bağlıdır. Bu çalışmada, web sayfalarının URL adreslerini kullanan bir steganografik yaklaşım önerilmiştir. Görüntüler, taşıyıcı olarak kullanılmıştır. LZW, Aritmetik kodlama ile DES, AES, Blowfish, RSA şifreleme algoritmaları önerilen yaklaşımı karmaşıklaştırmak, kapasiteyi artırmak ve güvenliği desteklemek için kullanılmıştır. Deneysel sonuçlar, önerilen yöntemin, iki taraf arasındaki iletişim için uygun olduğunu göstermiştir. İletişim sadece bir web adresi üzerinden yapıldığından, herhangi bir gözlem durumunda şüphe uyandırıcı bir durumla karşılaşılmamaktadır.

Anahtar sözcükler: Asimetrik şifreleme, Simetrik şifreleme, Steganografi, URL, Veri sıkıştırma

ABSTRACT

A NEW STEGANOGRAPHIC APPROACH OVER WEB ADDRESSES FOR A SECURE CYBER COMMUNICATION

Oğuzhan KENDİRLİ
Duzce University
Graduate School of Natural and Applied Sciences, Department of Computer Engineering
Master of Science Thesis
Supervisor: Assist. Prof. Dr. Esra ŞATIR
December 2013, 100 pages

With the rapid growth of information technology and internet, security has become a critical issue. Therefore, data hiding gained importance for delivering secret messages. Data hiding techniques hide messages such as images, videos, texts, etc. A data hiding technique is different from cryptology. A cryptographic scheme encrypts the message and then the message is sent, which is more secure and unpredictable, to the receiver's side. Since the message has a meaningless and uncommon content, the communication makes the observer aware of the exchange, so there is always a threat from a malicious attacker. Steganography is the technique of providing communication where no one except the intended receiver knows about the existence of secret data. Successful steganography depends on the carrier medium not to raise attention. In this study, a steganographic scheme that employs URL of web pages, has been proposed. Images have been used as the carriers. LZW, Arithmetic coding and AES, DES, Blowfish, RSA encryption algorithms have been used to increase complexity, capacity of the proposed approach and to support the security. Experimental results showed that the proposed method is feasible for any communication between two parties. Since the communication is performed via only a web address, it does not raise suspicion in case of an observation.

Keywords: Asymmetric encryption, Data compression, Steganography, Symmetric encryption, URL

EXTENDED ABSTRACT

A NEW STEGANOGRAPHIC APPROACH OVER WEB ADDRESSES FOR A SECURE CYBER COMMUNICATION

Oğuzhan KENDİRLİ
Duzce University
Graduate School of Natural and Applied Sciences, Department of Computer Engineering
Master of Science Thesis
Supervisor: Assist. Prof. Dr. Esra ŞATIR
December 2013, 100 pages

1. INTRODUCTION:

With the rapid growth of internet and information technology, information security has become a critical issue. As a sub-branch of information security, data hiding is an important subject that has gain a wide attention in recent years. Data hiding techniques consist of three sub-branches: Watermarking, cryptology and steganography. Watermarking focuses on the issues like copyright protection, tracking and etc. Its primary aim is to protect the cover medium from any kind of attacks or modification. Cryptology converts a secret message into a cipher text that has a meaningless format. This meaningless content naturally raises the attention of an observer during the secret communication. Steganography hides the secret message such that no one except the intended recipient knows about the existence of the secret message.

There are many steganographic schemes which use the cover mediums like images, sounds, videos and texts. In this study, a new steganographic approach that employs the URL information of web pages has been proposed. The primary aim of this method is to reduce the size of cover medium while rendering it imperceptible.

2. MATERIAL AND METHODS:

The goal of a steganographic algorithms is to be more statistical undetectable. Nowadays, the detectability of secret messages is mostly influenced by two factors:

1. The selection rule used to choose the imperceptible parts of cover object that can be modified during embedding the secret bits.

2. It is better to embed as many bits of secret message as possible by changing the least number of the cover object.

By considering the second item, we preferred to camouflage secret message, instead of embedding it into a multimedia object as in traditional multimedia steganography. Here, the purpose is to leave the cover image unchanged by only using it as a platform that has the all possible coordinates where the characters of secret message are mapped. After mapping the characters of secret message to the cover image, we obtain an array that holds the coordinate information of the secret characters (characters of secret message). Then we process this array via DES, AES, Blowfish and RSA encryption algorithms to increase security and LZW and Arithmetic coding algorithms to increase capacity and complexity. After these operations we obtain a random array in order to use in a URL of a web page that has he cover image.

Namely, we use two major elements here: One of them is the cover image (our map) while the other is the web address (coordinates) of the web page where the used cover image is demonstrated. Thus, firstly we aim to provide an unsuspicious medium for communication by using only a web address between the two parties. Secondly, we aim to render the stego-medium (cover image) resilliant to any kind of attack by making no change.

3. RESULTS AND DISCUSSIONS:

Experiments of the proposed method have been conducted by employing the software written in *C#* programming language. Here we used the randomly generated Lorem Ipsum patterns as the secret messages. Lengths of each *S* have been changed from 10 to 100, by incrementing the length ten at a time. Thus, we aim to perform an unbiased evaluation.

Generally, size of the part that is aimed to be embedded into the URL has reduced by means of compression. Thus, length of *T* is kept shortened as much as possible to make the URL of the web page seem innocent and unsuspicious.

As an example, the secret message and the constructed URL for this secret message has been given below, consequently:

Secret message: "*Lorem ipsu*"

URL:
http://s16.postimg.org/maz31aefp/00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuwnSaF8z+hrLw==/Baboon.png

This web address is the only part that will be sent to the recipient via any communication channel like Facebook, Skype and etc. Once the recipient get this URL, he/she can obtain the secret message by applying the extracting procedure of the proposed method.

4. CONCLUSION AND OUTLOOK:

With the rapid growth of information technology, nowadays, people can easily retrieve multimedia contents with their own computers or mobile phones over the Internet or mobile channels. Accordingly, data hiding has become one of the useful schemes for delivering secret messages. Data hiding technique hides messages such as images, videos, texts, etc. in the digital media imperceptibly. Here, steganography is the art of writing secret data in such a way that no one accept the intended receiver knows about the existence of secret data. So it is different from a cryptographic scheme since a cryptographic scheme renders the message meaningless and suspicious.

In this study, an unsuspicious and a covert communication has been targeted by means of the proposed method. Experimental results show that the implementation of the proposed method successfully performed in terms of embedding and extracting. Bu still there are some issues which need to be handled for a faster and efficient application. For instance, here imperceptibility has been provided but the capacity issue still needs to be tackled for a more efficient and faster communication. Besides, the proposed method is targeted to be a standard algorithm in case of applying it any kind of secret message.

1. GİRİŞ

İnternetin yaygın kullanımı ve bilgisayar endüstrisinin büyümesi ile günümüzde insanlar multimedya içeriklerine her yerden bilgi işlem ortamlarında, internet veya mobil kanallar üzerinden kendi bilgisayarları veya cep telefonları ile kolayca ulaşabilmektedirler. Multimedya ile ilgili araştırma ve uygulamalar son yirmi yılda büyük ölçüde artmıştır. Multimedya sinyal işlemlerine ek olarak, telif hakkı ile ilgili sorunları korumayı amaçlayan veri gizleme tekniklerine akademik sektör ve sanayi sektöründe ciddi ilgi vardır. Genel olarak, multimedya verisinin internet ya da kablosuz ağlar üzerinden iletildiği düşünülmekte ve her yerde bulunan bilgi işlem ortamları üzerinden dağıtım kolaylığı, multimedya içeriklerini her zaman ihlâl etme eğilimindedir [1].

Bu nedenle, gizli iletişim kurma yıllardır ilgi çeken, sıcak bir konu olmuştur. İnternetin büyük ölçekte genişlemesi ile her gün büyük miktarda web tabanlı bilgi transferi gerçekleşmektedir. Güvenlik nedenlerinden dolayı, veri gizleme alanında birçok farklı yöntem uygulanmakta ve her geçen gün yeni yöntemler geliştirilmektedir. Kriptografi, steganografi ve filigranlama en bilinen bilgi güvenliği tekniklerindendir ancak hepsi farklı mekanizmalar altında işlemektedir. Kriptografi gizli bir kod içine yazarak ve şifreleyerek verileri okunamaz hale getirmektedir. Ayrıca kimlik doğrulama, gizlilik ve veri bütünlüğünü sağlamaktadır. Filigranlama tekniğinin amacı, verideki bazı bilgileri gizleyerek belirli bir içerik üzerinde fikri mülkiyet hakları vb. için kanıt sağlamaktır. Steganografi ise verinin varlığını gizlemekteyken şeffaflık, sağlamlık ve kapasite sağlamaktadır [1,2].

Veri gizleme tekniği, mesajları görüntüler, videolar, haritalar, metin vb. gibi dijital medya ortamları içine gizlemektedir. Veri gizleme tekniği kriptolojinden farklıdır. Kriptolojide, mesajı şifrelemek için bir şifreleme düzeni kullanılmakta ve alıcıya daha güvenli ve tahmin edilemeyen bir mesaj gönderilmektedir. Ancak içeriği dolayısıyla herkes iletişim sırasında mesaj değişiminin farkında olduğundan, kriptolojide saldırıları çekme bakımından her zaman bir tehdit vardır [3].

Veriyi diğer bir verinin içine gizleyerek veya gömerek görünmez yapma tekniğine steganografi denilmektedir. Veriyi gizleme amacıyla kullanılan bu diğer veri parçasına örten ortam ya da taşıyıcı denir. Gizlenmiş veriyi içeren bu örten ortama ise stego nesne denilmektedir ve stego nesne, saklanabilmekte ya da iletilebilmektedir. Gizli veri değişik çeşitlerde örten ortamlara gömülebilmektedir. Stego görüntü veya örten görüntü, verinin bir görüntü dosyasına gömülmesi sonucunda oluşan nesnedir. Örten metin, stego görüntü, örten ses, stego ses, örten video, stego video vb. isimlendirmelerde benzer şekilde oluşmaktadır. I. Uluslararası Bilgi Gizleme Semineri'nde bu terminoloji kabul görmüştür [4].

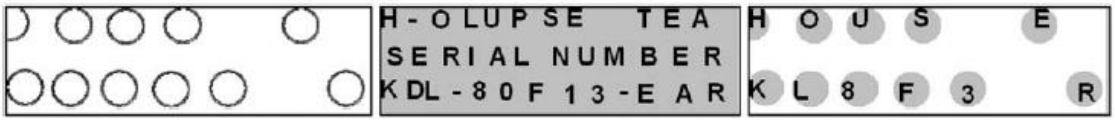
Çizelge 1.1. Steganografi, Filigran ve Kriptolojnin Karşılaştırılması [1].

Kriter/metot	Steganografi	Filigran	Kriptoloji
Taşıyıcı	Herhangi bir sanal ortam	Çoğunlukla görüntü ya da ses dosyaları	Genelde metin bazlı, bazı eklemeler ile görüntü dosyaları
Gizli veri	Yük	Filigran	Düz metin
Anahtar	İsteğe bağlı	-	Şart
Giriş dosyaları	Kendi içinde gömülü bir dosya yoksa, en az iki tane	-	Bir
Algılama	Kör	Genelde aydınlatıcı (yani geri dönüşüm için orijinal filigran ya da örten nesne gerekli)	Kör
Doğrulama	Verinin tamamı	Genelde çapraz bağıntı ile	Verinin tamamı
Amaç	Gizli iletişim	Telif hakkı koruma	Veri koruma
Sonuç	Stego dosyası (stego-file)	Filigranlanmış dosya (watermarked-file)	Şifreli-metin (chiper-text)
Amaç	Algılanabilirlik\ kapasite	Sağlamlık	Sağlamlık
Saldırı şekilleri	Steganaliz	Görüntü işleme	Kriptal analiz
Görünürlük	Asla	Bazen	Daima
Zayıflık	Fark edildiğinde	Değiştirildiğinde ya da kaldırıldığında	Şifre kırıldığında
Örtü ile bağıntı	Örten nesne ile zorunlu bir ilgisi bulunmamaktadır. Gizli mesaj örten nesneden çok daha önemli.	Genelde örten görüntünün bir niteliğini haline gelir. Örten nesne mesajdan fazla önem taşır.	Mevcut değil
Esneklik	İstenen taşıyıcı seçilebilir	Taşıyıcı nesne seçimi kısıtlıdır.	Mevcut değil
Geçmiş	Çok eski, sayısal hali hariç.	Modern çağ	Modern çağ

Steganografide gizli bilgi öyle bir şekilde gömülmelidir ki, gönderici ve hedeflenen alıcı dışında hiç kimse gizli bir mesajın varlığını fark edememelidir. Ayrıca, gizli mesajın varlığı neredeyse herhangi bir üçüncü şahıs tarafından tespit edilmemelidir. Yani güvenli steganografi, gizli verinin hedefe tespit edilmeden ulaştırılmasını gerektirmektedir. Çizelge 1.1.'de steganografi, filigranlama ve kriptolojinin karşılaştırılması görülmektedir [1,5].

Steganografi sözcüğü köken olarak "gizli yazı " anlamına gelen Yunanca kelimelerden türetilmiştir. Binlerce yıldır çeşitli şekillerde kullanılmaktadır. M.Ö. 5. yüzyılda Histaiacuss, bir kölenin başını tıraş ettirmiş, kafatasına bir mesajı dövme yaptırmış ve saçları tekrar uzadıktan sonra köle, kafasındaki mesajla birlikte gönderilmiştir.

Beş yüz yıl önce, İtalyan matematikçi Jerome Cardanre eski bir Çin gizli yazı yazma metodu keşfetmiştir. Buna göre, iki taraf arasında delikli bir kâğıt maske paylaşılmıştır. Bu maske boş bir kağıt üzerine yerleştirilmiş ve gönderici gizli mesajı deliklerin içinden yazmıştır. Daha sonra, Şekil 1.1'de görüldüğü gibi gizlenmiş metnin şüphe uyandırmaması amacıyla maskeyi kaldırılmış ve boşluklar doldurulmuştur. Bu yöntem, Cardan'a atfedilmiştir ve Cardan Grille olarak adlandırılmaktadır.



Şekil 1.1. Cardan Grille [6].

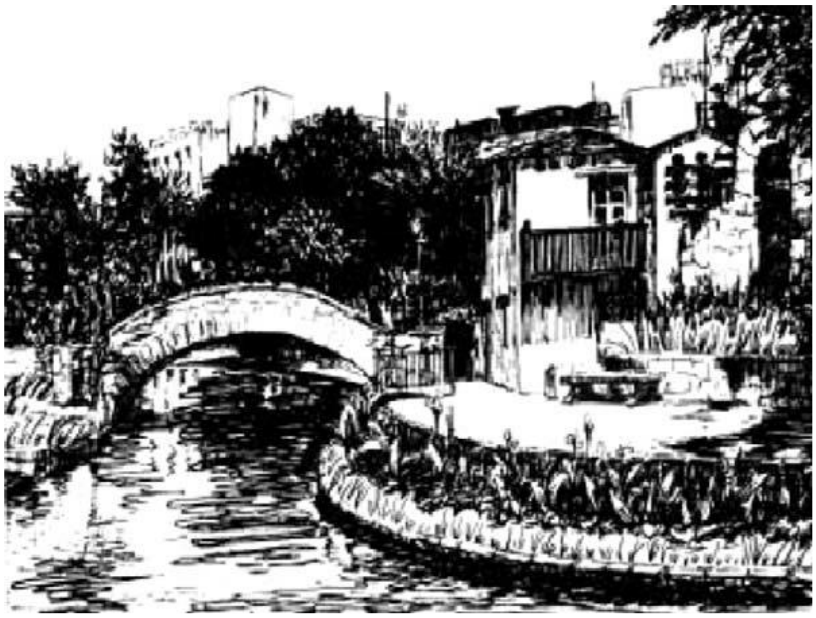
Ayrıca II. Dünya Savaşı sırasında Almanların mikro noktalar gibi çeşitli steganografik yöntemler icat ettikleri bilinmektedir. Bununla birlikte görünmez mürekkep ve boş şifreleri tekrar kullandıkları rapor edilmiştir. Örnek olarak, bir Alman casusu tarafından II. Dünya Savaşı sırasında gönderilen bir mesaj aşağıdaki gibidir:

"Apparently neutral's protest is thoroughly discounted and ignored. Isman hard hit. Blockade issue affects pretext for embargo on by-products, ejecting susets and vegetable oils."

Bu metinde her bir kelimenin ikinci harfini ele alarak gizli mesajı çıkarmak mümkündür:

"Pershing sails from NY June 1" [4].

1945 yılında, Morse kodu Şekil 1.2' de görülen bir çizimde gizlenmiştir. Gizli bilgiler nehir kıyısında uzanan çimler üzerinde kodlanmıştır. Burada uzun çim bir çizgi, kısa çim bir noktayı göstermektedir. Buna göre gizlenen mesaj: *''Compliments of CPSA MA to our chief Col Harold R. Shaw on his visit to San Antonio May 11th 1945''* .



Şekil 1.2. Mors kodunun görüntü içerisine gizlenmesi [6].

Bilgisayar ve internet gücündeki gelişmelerle birlikte dijital sinyal işleme (DSP), bilgi ve kodlama teorilerinin gelişimi ile steganografi dijital platformda uygulanabilir hale gelmiştir. Bu dijital dünya alanında steganografi, çeşitli ilginç uygulamaları ortaya çıkararak dikkatleri üzerine çekmiş ve böylece devam eden evrimini garantilemiştir. Müzik dosyaları, Hyper Text Mark-Up Language (HTML) dosyaları, çalıştırılabilir dosyalar (.exe) ve genişleyebilir işaretleme dili (XML) dosyaları gibi daha basit formdaki dosyaların içine bile veri gizleme işleminin yapıldığı örnekler mevcuttur [6].

Genel olarak, popülerlik nedeniyle internet üzerinden görüntü, ses ve video transferi, taşıyıcı olarak kullanılmaktadır. Bununla birlikte, internetin hızlı büyümesi nedeniyle çeşitli yeni medya uygulamaları geliştirilmiştir. Örneğin, kullanıcılar, arkadaşları ile online oyun oynayabilmekte, arkadaşlarının mobil cihazlarına bulmaca gönderebilmektedirler. Niwayama ve ark. tarafından gerçekleştirilen çalışmada veri gizlemek için taşıyıcı ortam olarak labirent oyunları seçilmiştir. Ancak bu yöntemin iki dezavantajı vardır. Biri çok az miktardaki gömme kapasitesi, diğeri stego labirentin

mükemmel olmamasıdır [7]. Lee ve ark tarafından, bu metodun dezavantajları üzerinde çalışılmıştır. Gömme kapasitesinin artırıldığı ve mükemmel labirentin oluşturulduğu yeni bir metot önermişlerdir.

Jing Yang ve Chen değişik animasyonlarla PowerPoint dosyası içine gizli bir mesajı gömmek için bir yöntem önermişlerdir. Önerilen metotta, farklı mesajları temsil etmek için animasyonlar kullanılmıştır. Bu amaca ulaşmak için, ilk olarak animasyonlar ve mesaj parçaları arasındaki ilişkiyi kaydedecek bir kod defteri tasarlanmıştır. Buna ek olarak, bir PowerPoint dosyasındaki animasyonların otomatik olarak uygulanması imkânsızdır ve pratik değildir. Daha sonra bu oluşturulan kod defterine göre, gizli mesajın yarı otomatik olarak animasyonlara çevrilmesi amacıyla interaktif bir sistem tasarlanmıştır. Burada herhangi bir animasyon, PowerPoint dosyasının içeriğini değiştirmediğinden, bu dosyanın gerçek içeriği bozulmadan muhafaza edilebilmektedir. Ayrıca, önerilen metot format dönüştürme saldırılarına karşı dirençlidir. Ancak kullanılan animasyon sayısı fazla olmadığından bu metodun kapasitesi düşüktür [8].

Web tabanlı iletişim büyük bir miktarda bant genişliğine sahiptir ve bu nedenle gizli iletişim için kullanılabilmektedir. HTML ve XML, web geliştirme için iki önemli temel ve evrensel araçlardır. Script dilleri de, dinamik web geliştirme için kullanılabilir ancak sonunda tüm tarayıcılar script kodunu HTML formatında komut dosyasına çevirmek zorundadır. Mir ve Hussain birkaç metin steganografi metodunu, XML dosyalarına uygulayarak veri gizleme gerçekleştiren bir metot önermişlerdir. XML sayfası, taşıyıcı ortam olarak kullanılmıştır ve veri gizlenmeden önce güvenliğin artırılması amacıyla Gelişmiş şifreleme standardı (AES) algoritması ile şifrelenmiştir [2].

Castiglione tarafından gerçekleştirilen çalışmada, istenmeyen (spam) e-postalar, internet üzerinden etkili, esnek ve eş zamanlı olmayan gizli iletişim kanalları uygulamak için ilginç bir fırsat sunmaktadır. İlk olarak, spam mailler birer e-postadır ve yıllarca taşıdıkları protokollere dayanmakta ve hala bu protokolleri taşımaktadırlar. Tüm e-posta yaşam döngüsü ile ilgili protokoller gizli verileri kodlamak için pek çok boşluk öneren Bilgi dönüşümü için Amerikan standart kodlama sistemi (ASCII) kontrol dizgileri üzerine inşa edilmiştir. İkincisi, spam mailler rutin olarak çıkarılmakta ve bunun yanı sıra onları spam olarak sınıflandırmak için herhangi bir itina gerektirmemektedir Bu, gelişmiş güvenlik duvarları ve trafik analizörleri tarafından bile algılanabilir olmayan sağlam, gizli bir iletişim imkanı sunmaktadır. Önerilen steganografik sistem, ne ilgili taşıma

protokolleri ve mekanizmalarını etkilemekte ne de örten e-posta iletisi içeriğini değiştirmektedir. Buna ek olarak, son kullanıcılara kadar fark edilebilir herhangi bir performans düşüşü veya veri kaybına neden olmamaktadır [9].

Sohbet odaları vasıtasıyla iletişimin, insanların yaşamında oldukça popüler bir hale geldiğini gören Wang ve Chang 2009 yılında yeni bir metin steganografi metodu önermişlerdir. Bu metotta gizli bilgi, sohbet odalarında internet üzerinden iletişim esnasında yüz mimiklerini ifade eden küçük boyutlu görüntüler yani ikonlar (emoticon) içerisine gömülmektedir. Metotta öncelikle, göndericinin ve alıcının ikon tablosunun aynı olması gerekmektedir. Daha sonra, gönderici bu ikonları anlamlarına göre (gülümseme, gülme, ağlama vb.) farklı kümelere ayırmaktadır. Her bir ikon yalnızca bir kümeye ait olabilmektedir. Sıfırdan başlayarak bir ikonun kendi kümesindeki sıra numarası, gömülecek gizli bitleri göstermektedir. Bu nedenle önerilen steganografik metot, her bir kümedeki ikon sırasını kontrol etmek amacıyla gizli bir anahtar kullanmaktadır. Gizli anahtar sadece alıcı ve gönderici tarafından tutulmaktadır. Sohbet odalarında kullanılan çok fazla sayıda ikon olduğundan bu metotla kapasite oldukça artırılsa da, bu artış büyük ölçüde önceden paylaşılan ikon tablosuna ve her bir kümedeki ikon sayısına bağlı olacaktır [10].

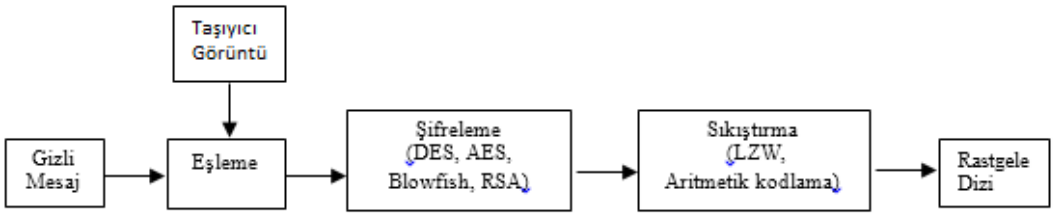
Samphaiboon, 2009 yılında yeni bir steganografik metot geliştirmiştir. Geliştirilen metotta, gizli mesaj, televizyon ve web siteleri gibi medya ekranlarında kısa bir metin dizisi içerisinde çoklu alıcıya gönderilmektedir. Ancak bu yöntemde, uygun optik karakter tanıma (OCR) biriminin kod çözücüde bulunduğu varsayılmaktadır. Metot, Tayland dili üzerinde uygulanmıştır ve gömme aşamasında, etkili birkaç metinden bite dönüşüm metodu önerilmiştir. Bu metodun en büyük avantajı, gizli mesajın aynı anda farklı yerlerdeki çoklu alıcılara yayınlanabilmesidir. Metodun dezavantajlarından birisi ise yazar tarafından, göndericinin kısa metnin iletildiği kanal üzerinde kontrole sahip olduğu varsayılmaktadır. Bir diğer kaçırılmaması gereken nokta ise görüntülenen metni tanıyabilen ve bunu makine tarafından okunabilir forma çeviren metinsel görüntü okuma biriminin var olduğunun kabulüdür [4].

Steganografik bir algoritmanın amacı, istatistiksel olarak algılanamaz olmaktır. Günümüzde, gizli mesajın tespit edilirliği, çoğunlukla şu iki faktöre bağlıdır:

1. Gizli bitleri gömme işlemi sırasında örten metinde modifiye edilebilecek algılanamaz kısımların seçim kuralı.

2. Örten nesneyi mümkün olduğunca az değişikliğe uğratarak, mümkün olduğunca fazla miktarda veri gömme [11].

Bu tez çalışmasında, ikinci madde dikkate alınarak, klasik multimedya steganografisinde olduğu gibi gizli bir mesajı bir multimedya nesnesine gömme yerine, bu gizli mesajı kamufle etme yoluna gidilmiştir. Buradaki amaç, gizli mesajın karakterlerinin eşlendiği tüm koordinat bilgilerine sahip örten görüntünün yalnızca bir platform olarak kullanılması, yani değiştirilmemesidir. Gizli mesajın karakterleri, örten görüntüye eşlendikten sonra, gizlenen karakterlerin koordinat bilgisini tutan bir dizi elde edilmektedir. Daha sonra bu dizi güvenliğin artırılması amacıyla veri şifreleme standardı (DES), AES, Blowfish ve Rivest, Shamir, Adleman (RSA) şifreleme algoritmaları, kapasite ve karmaşıklığın artırılması amacıyla da Lempel–Ziv–Welch (LZW) ve Aritmetik kodlama sıkıştırma algoritmaları ile işlenmiştir. Bu işlemlerden sonra, örten görüntünün upload edildiği web sayfasının tek tip kaynak konumlayıcı (URL) adresine eklenebilecek rastgele bir dizi elde edilmektedir. Şekil 1.1' de önerilen metodun blok diyagramı gösterilmektedir.



Şekil 1.3. Önerilen metodun blok diyagramı.

Burada, iki ana öge kullanmaktayız. Bunlardan birisi, bir nevi harita olarak kullanılan örten metin, bir diğeri ise bu görüntünün yüklendiği web sayfasıdır. Bu web sayfasının URL bilgisi, gizlenen karakterlerin şifrelenmiş ve sıkıştırma yoluyla karmaşıklaştırılmış ve indirgenmiş koordinat bilgilerini içermektedir. Böylelikle, ilk olarak iki şahıs arasında iletişim amacıyla sadece bir web adresi kullanarak dikkat çekmeyen bir ortamın oluşturulması amaçlanmaktadır. İkinci olarak ise, görüntüde herhangi bir değişikliğe sebebiyet verilmemesi neticesinde oluşturulan bu stego ortamın görüntüye karşı gerçekleştirilecek saldırılara dirençli olması amaçlanmaktadır.

2. MATERYAL VE YÖNTEM

Ele alınan örten nesne görüntü olduğundan ve önerilen metot esas olarak görüntü üzerinde hiçbir değişiklik yapmama prensibine dayandığından, bu bölümde temel görüntü steganografi teknikleri sınıflandırılarak kısaca özetlenmiştir. Böylelikle önerilen metot ile literatürdeki görüntü steganografisi metotları arasındaki farkın daha da somutlaştırılması amaçlanmaktadır.

2.1. UZAMSAL ALANDA STEGANOGRAFİ

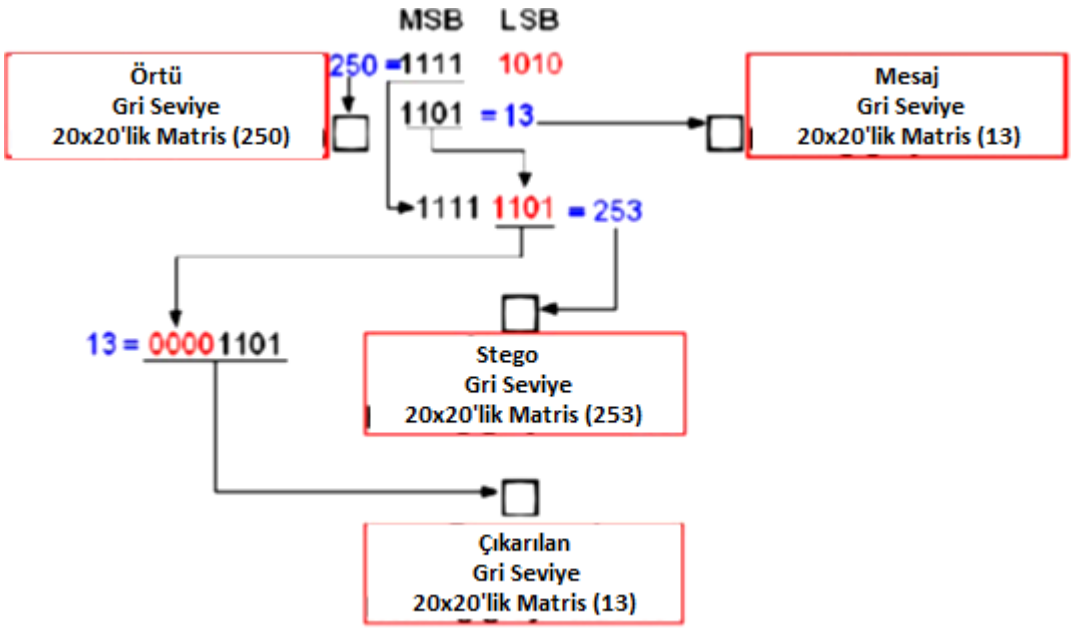
Genellikle internet üzerindeki en popüler görüntü biçimleri grafik değişim biçimi (GIF), birleşik fotoğraf uzmanları grubu (JPEG) ve daha az kapsamlı olan taşınabilir ağ grafiği (PNG)'dir. Geliştirilen tekniklerin çoğu, literatürde basit veri yapısından ötürü Bitmap (BMP) formatını kullanan ve bazı istisnalarla birlikte bu format yapılarını kullanan metotlar üzerinde kuruludur. Gömme sürecini aşağıdaki gibi tanımlanabilir:

$$E_m: C \oplus K \oplus M \rightarrow C' \tag{2.1}$$

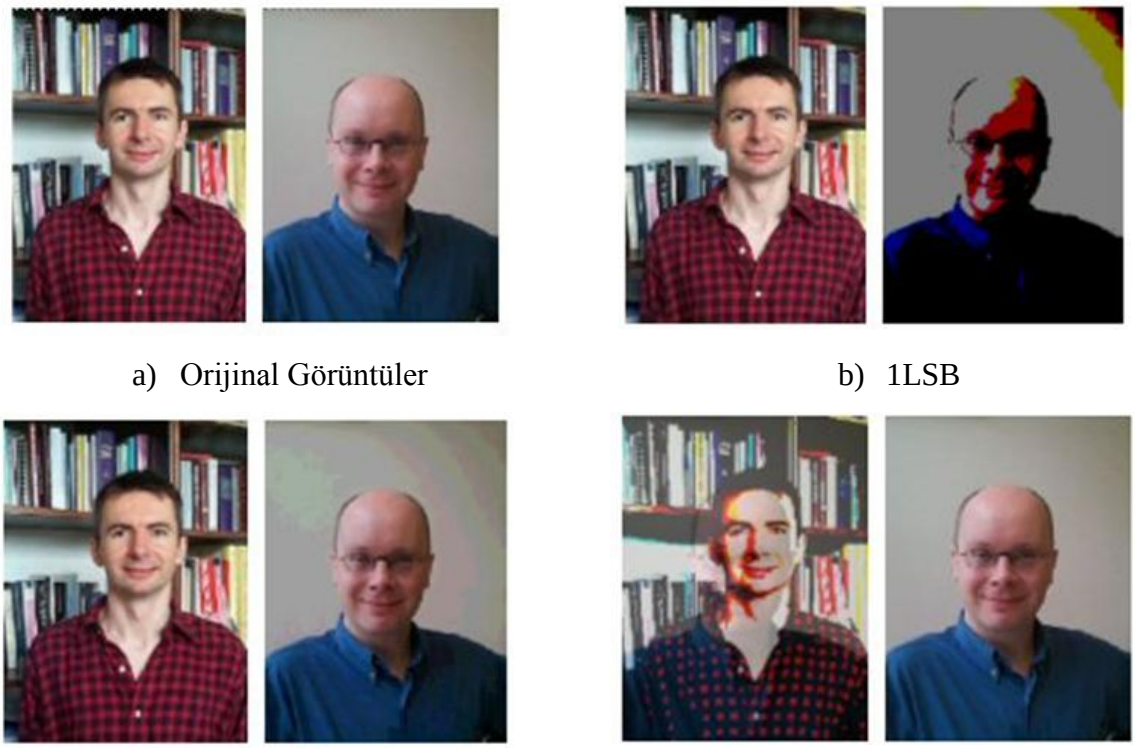
$$E_x\,(E_m(c,k,m)) \approx m,\ \forall c \in C,\ k \in K,\ m \in M \tag{2.2}$$

Burada C; taşıyıcı nesneyi (A görüntüsü) göstermekte, C' ise stego-görüntüyü göstermektedir. K; gizlenecek mesajı şifrelemek için kullanılan seçimlik bir anahtar ve M' de gönderilecek mesajı (B görüntüsü) ifade etmektedir. E_m; gömme için, E_x ise çıkarım için kısaltmadır [6].

En az anlamlı bit (LSB) steganografi algoritmaları, doğrudan gizlenecek veri bitleri ile örten görüntünün LSB' leri değiştirilerek LSB düzlemlerini işlenmektedir. Şekil 2.1'de bu metot için genel yapı gösterilmektedir. Şekil 2.2'de ise veri gizleme için kullanılan LSB sayıları artırıldıkça oluşan bozulmalar gösterilmektedir.



Şekil 2.1. LSB yer değiştirmesinin genel yapısı [6].



a) Orijinal Görüntüler b) 1LSB

c) 4LSB d) 7 LSB

Şekil 2.2. Kullanılan LSB sayılarına göre görüntülerde meydana gelen bozulmalar a)Orijinal Görüntüler b)1LSB c) 4LSB d) 7LSB [12].

Şekil 2.2'de görüldüğü gibi kapasite ve bozulma arasında bir ödünleşim mevcuttur. Şekil 2.2'de sol taraftaki görüntü örten, sağ taraftaki görüntü ise saklanacak görüntüdür. Saklama amacıyla kullanılan bit sayılarına göre örten görüntüde oluşan bozulmalar ve örten görüntüden çıkarılan gizlenmiş görüntüler gösterilmektedir. Buna göre, her iki görüntüye de eşit ağırlık verildiğinde, yani kullanılan LSB sayıları 4 olduğunda bu metot iyi çalışmaktadır [4,12].

2.2. FREKANS ALANINDA STEGANOGRAFİ

Ayrık Kosinüs Dönüşümü (DCT) teknikleri, JPEG görüntülerinin yaygın kullanımı nedeniyle popüler olmaktadır. JPEG, steganografide ele alınan en yaygın görüntü biçimidir. LSB tekniği her ne kadar büyük bir adım olarak görülse de ve algılanamazlık bakımından mükemmele yakın olsa da saldırılara karşı direnci düşüktür. F giriş görüntüsü ve T çıkış görüntüsü için iki boyutlu kosinüs dönüşümü aşağıda verilmiştir [5,6].

$$T_{pq} = \propto_p \propto_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} F_{mn} \cos\frac{\pi(2m+1)p}{2M} \cos\frac{\pi(2n+1)q}{2M} \tag{2.3}$$

$$0 \leq p \leq M\text{-}1 \tag{2.4}$$

$$0 \leq q \leq N\text{-}1 \tag{2.5}$$

ve

$$\propto_p = \begin{cases} \frac{1}{\sqrt{M}}, & p = 0 \\ \sqrt{2}/M, & 1 \leq p \leq M-1 \end{cases} \tag{2.6}$$

$$\propto_p = \begin{cases} \frac{1}{\sqrt{N}}, & q = 0 \\ \sqrt{2}/N, & 1 \leq q \leq N-1 \end{cases} \tag{2.7}$$

Burada M, N giriş görüntü boyutları, m ve n, sırasıyla 0' dan M-1' e ve 0' dan N-1'e kadar değişen değişkenlerdir. DCT, video ve görüntü sıkıştırmasında yaygın olarak kullanılmaktadır.

Her blok için Denklem 2.3' ten elde edilen. DCT katsayıları, kuantalama tablosu kullanılarak kuantalanmaktadır. JPEG standardı tarafından kullanılan bu tablo Çizelge 2.1'de verilmiştir:

Çizelge 2.1. DCT kayıplı sıkıştırmada kullanılan JPEG parlaklık kuantalama çizelgesi. 16 değeri, DC (iki boyutta 0 frekanslı olan katsayı) katsayısını; diğer değerler AC (iki boyuta kalan 0'dan farklı 63 katsayı) katsayısını ifade eder [6].

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Bu değerleri barındıran bir tablo seçmenin arkasındaki mantık, görüntü sıkıştırma ve kalite arasındaki dengeyi kurmaktır. Kuantalamanın amacı, DCT ile sıkıştırılan hassas değerleri çözümlemektir. Kuantalama adımı şu şekilde tanımlanmaktadır:

$$f'(w_x, w_y) = \left\lfloor \frac{f'(w_x,w_y)}{\Gamma'(w_x,w_y)} + \frac{1}{2} \right\rfloor, (w_x, w_y) \in 0,1, \dots ,7 \qquad (2.8)$$

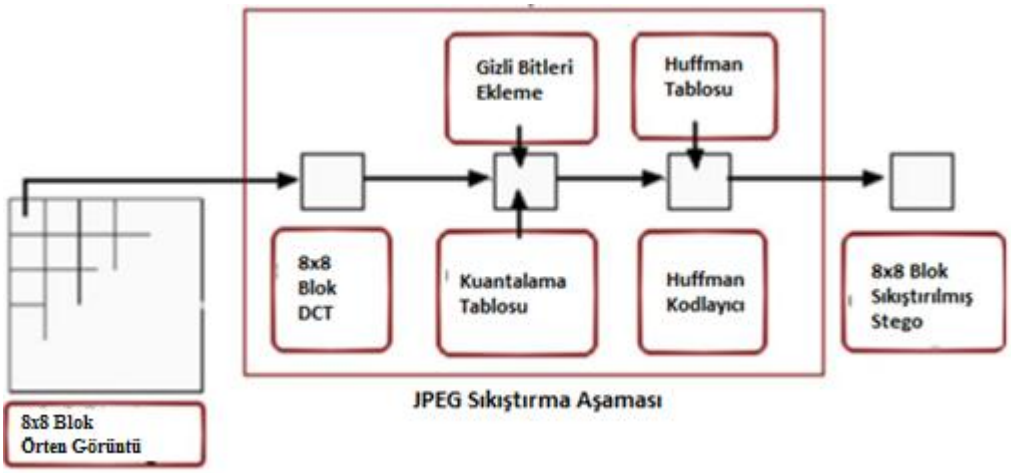
x ve y; görüntü koordinatları, $f'(w_x,w_y)$; sonuç fonksiyonu, $f(w_x, w_y)$; örtüşmeyen 8×8'lik görüntü yoğunluk bloğudur. $\lfloor . \rfloor$ yuvarlama operatörü, $\Gamma(w_x,w_y)$ ise JPEG kalitesiyle alakalı olan kuantalama adımıdır:

$$\Gamma(w_x, w_y) = \begin{cases} max\left(\left\lfloor \frac{200-2Q}{100} QT(w_x, w_y) + \frac{1}{2} \right\rfloor, 1\right), & 50 \le Q \le 100 \\ \left\lfloor \frac{50}{Q} QT\Gamma(w_x, w_y) + \frac{1}{2} \right\rfloor, & 0 \le Q \le 50 \end{cases} \qquad (2.9)$$

Burada Q, Çizelge 2.1'de gösterilen kuantalama tablosudur. Q, kalite faktörüdür. Daha sonra, elde edilen $\Gamma(w_x, w_y)$ sıkıştırılması için Huffman kodlama algoritması kullanılmaktadır. Gereksiz ve gürültülü verinin çoğu bu aşamada kaybolmaktadır.

Buradaki tekniklerin çoğu veri gömmek için araç olarak JPEG görüntülerini

kullanmaktadır. JPEG sıkıştırması, ardışık alt görüntü bloklarını (8×8), 64 DCT katsayılarına çevirmek için DCT kullanmaktadır. Veriler bu katsayılardaki önemsiz bitlere yerleştirilmektedir. Ancak, bir adet herhangi bir katsayıyı değiştirmek, tüm 64 blok pikselleri etkilemektedir. Değişiklik, uzamsal bölge yerine frekans bölgesinde gerçekleştiğinden ve katsayılar özenle işlendiğinde verilen örten görüntüde görünür hiçbir değişiklik oluşmamaktadır. Şekil 2.3' te frekans alanında gömmenin genel sürecini gösteren veri akış şeması verilmektedir.



Şekil 2.3. Frekans alanında gömme genel prosesini gösteren veri akış şeması [6].

En ufak bir veriyi değiştirmek görüntüdeki tüm 8×8' lik bloğu etkileyeceği için, 8×8' lik DCT katsayı bloğunda hangi değerlerin değiştirileceği çok önemlidir. Şekil 2.4'te dikkatle seçilmiş DCT katsayılarıyla gerçekleştirilmiş bir uygulama gösterilmektedir [6].

Şekil 2.4. DCT düzeyinde gömme çok başarılı ve güçlü bir araçtır ancak katsayılar dikkatle seçilmez ise bozulmalar fark edilebilir olacaktır [6].

2.3. ADAPTİF STEGANOGRAFİ

Adaptif steganografi, görüntü dosya formatı kullanılarak uygulanan steganografi ve uzamsal bölge steganografisinin özel bir durumudur. "Ayrıca, Maskeleme" ya da "Model tabanlı" olarak da bilinmektedir. Bu metot, görüntünün LSB/DCT katsayılarını kullanmadan önce istatistiksel özelliklerini dikkate almaktadır. Bu istatistiksel özellikler, değişiklik yapılabilecek alanları göstermektedir. Örten görüntüye göre piksellerin rastgele adaptif seçimi ve büyük standart sapmaya sahip bir bloktaki piksellerin seçimi ile karakterize edilmektedir. İkinci durum, alanların tek bir renkte olmasını engellemektedir. Bu özellik, adaptif steganografinin, kasten eklenmiş ya da hali hazırda gürültülü görüntüleri veya renk karmaşıklığı gösteren görüntüleri aramasına sebep olmaktadır [6].

2.4. KRİPTOLOJİ NEDİR?

Kriptoloji; Yunanca, "kryptos + logos" (gizli + bilim) kelimelerinin birleşimden meydana gelmektedir. Basit anlamda şifreli belgeler, gizli yazılar bilimidir diyebiliriz. Kriptoloji, "Kriptografi" ve "Kripto analiz" diye iki ana dala ayrılmaktadır.

Kriptografi; Yunanca, "kryptos + graphein" (gizli + yazmak) kelimelerinin birleşiminden meydana gelmektedir. Şifreleme ve şifre açmakta kullanılan tekniklerin tümünü inceleyen bilim dalıdır.

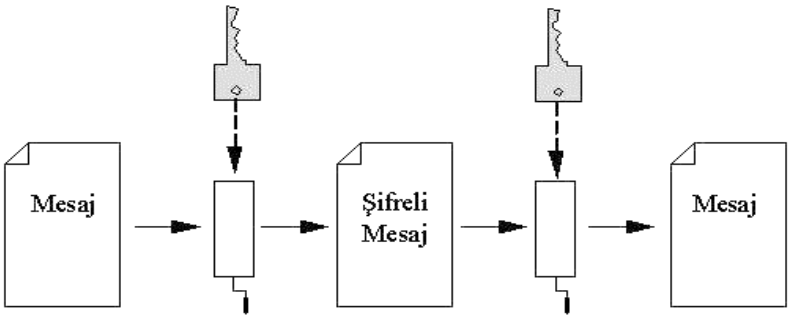
Bir şifreleme sistemini veya sadece şifreli mesajı inceleyerek, şifreli mesajın açık halini elde etmeye çalışan kriptoloji alanına Kripto analiz denilmektedir. Kısaca kriptografi ile şifrelenen metinler, kripto analiz ile elde edilmeye çalışılır.

Kriptoloji, bilgi gizliliğinin önem kazandığı andan itibaren hep ön planda olmuştur. Binlerce yıl önce; devletler, imparatorluklar gizli ve önemli bilgileri düşmanın eline geçmeden iletebilmek için özel yetiştirilmiş ulaklar ve güvercinler kullanmışlardır. Fakat bu yöntemler çoğunlukla mesajın başkalarının eline geçmesine engel olamamıştır. Bu nedenlerden ötürü mesajın anlamını da gizleme gereği doğmuştur ve kriptolojiyi ortaya çıkarmıştır. Binlerce yıl önce, bilgileri kodlama ile başlayan kriptoloji; yani sözcüklerin veya cümlenin başka bir sözcük, sayı ya da sembol ile yerini değiştirerek göndermek, o

dönem için mesajın başkaları tarafından anlaşılmasını engellemek için yeterli olmuştur. Gelişen teknoloji ile birlikte bu yöntemlerde hızla güvenirliğini kaybetmiştir. Günümüzde kriptolojide; matematik temellerine, bilgisayarların işlem güçlerine dayalı sistemler ön plana çıkmıştır [13].

2.4.1.Kriptografi

Kriptografi, veriyi yalnızca okuması istenen kişilerin okuyabileceği bir şekilde saklamak veya güvenli olmayan ortamlardan iletilmesini sağlamak amacıyla kullanılan bir bilimdir (Şekil.2.5). Bilgi güvenliği; başkası tarafından dinlenme, bilginin değiştirilmesi, kimlik taklidi gibi tehditlerin ortadan kaldırılması ile sağlanır. Bu güvenliğin sağlanması için veri, bilgi vb. matematiksel yöntemler kullanılarak kodlanır başkalarının okuyamayacağı hale getirilir ve bu matematiksel kodlama "kripto algoritması" olarak adlandırılır.



Şekil 2.5. Kriptografinin çalışma şekli.

Kripto algoritmasının güçlü olması tehditlere karşı dirençli olması istenen en önemli özelliktir. Şifreleme veya şifre açma anahtarlarından biri üçüncü şahıslar tarafından ele geçirebilir bu durumda sistem güvenliği sağlayan tüm kısımların çözülememesi gerekmektedir.

"Bu dünyada kriptografinin iki türü vardır: Kardeşinizin belgelerinizi okumasını engelleyen kriptografi, ve hükümetlerin belgelerinizi okumasını engelleyen kriptografi." Bruce Scheier (Counterpane Internet Security şirketinin kurucusu).

Kriptografinin gücü; şifreli metni, açık metne çevirmek için gerekli zamanın ve araçların kapasitesiyle ölçülür. Güçlü kripto algoritmalarının her türlü tehdide karşı sıkı güvenlik prensiplerini karşılaması istenilir [13].

2.4.2.Kriptografi Çeşitleri

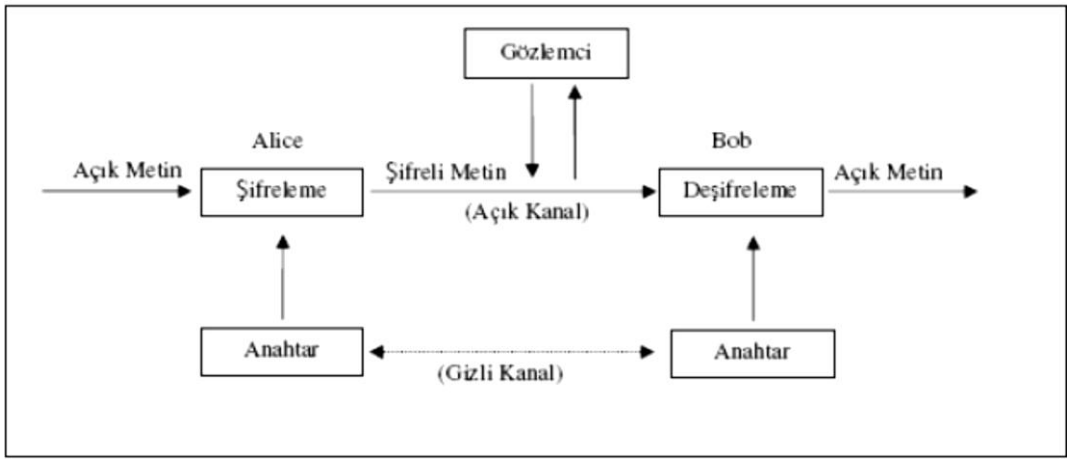
Özellikle II. Dünya savaşından sonra kriptoloji biliminde bilgisayar ve matematiğin etkin bir şekilde kullanılmaya başlanmasının ardından şifreleme sistemleri yeni bir boyut kazandı. Kripto sistemler simetrik ve asimetrik sistemler olarak ikiye ayrılmaktadır.

2.4.3. Simetrik Şifreleme

Geleneksel veya gizli anahtarlı şifreleme olarak da adlandırılan simetrik şifrelemede, şifreleme ve şifre açma için tek bir anahtar kullanılmaktadır. Mesajı gönderen taraf, mesajı bir anahtarla şifrelerken, alıcı taraf da aynı anahtarı kullanarak şifreyi açmaktadır (Şekil 2.6).

Alıcı ve göndericinin simetrik şifreleme kullanarak güvenli bir şekilde haberleşmesi için, bir anahtar üzerinde anlaşmaları ve bu anahtarı gizli tutmaları gerekmektedir. Gönderici ve alıcı ayrı konumlarda bulunuyorsa, taşıma ortamlarının (kanal) özel anahtarın saklanabilmesi açısından yeterli güvenilirlikte olması gerekmektedir. Çünkü anahtarı ele geçiren kişi şifreyi kolaylıkla çözebilir. Anahtarların üretimi, iletimi ve saklanması anahtar yönetimi olarak adlandırılır. Anahtar yönetimi sorunları tüm şifreleme sistemlerinde uğraşılması zorunlu bir durumdur. Anahtarların gizli kalması gerektiğinden dolayı, simetrik şifreleme yöntemi, özel anahtar yönetiminde oldukça problem yaşamaktadır.

DES, Blowfish, Twofish, AES, Carlisle Adams Stafford Tavares-128 (CAST128), Rivest cipher 5 (RC5) bazı simetrik şifreleme algoritmalardır. Bahsi geçen simetrik algoritmaların en büyük avantajı basit ve kolay uygulanabilir olmasıdır. Ancak, şifreleme ve şifre çözme için aynı anahtarın kullanılıyor olması güvenlik açısından büyük bir dezavantaj oluşturmaktadır. Diğer şahıslara bu anahtarın güvenli olarak gönderilmesi sorunuyla birlikte, anahtara sahip olan şahısların anahtarı ne kadar gizli tutacağı da başka bir problem teşkil etmektedir. O nedenle, bu tür algoritmalar, bilgisayar dosyaları gibi paylaşımın olmadığı durumlar için daha uygundur.



Şekil 2.6. Simetrik şifreleme [13].

Simetrik sistemlerde anahtar dağıtma büyük bir problem oluşturmaktadır. Bu problem temel olarak gönderici ve alıcının her ikisinin de anahtarın bir kopyasına sahip olmalarından kaynaklanır, bu ikili bir başkasının bu anahtarın bir kopyasını elde etmesini önlemelidir.

Örneğin; Bob ve Alice, aralarında bilgi alışverişini sağlamak için, simetrik şifreleme sistemi kullanmak istiyorlar ise bu işlem için her ikisi de verinin şifrelenmesinde kullanılacak bir gizli anahtarı edinmelidir. Bununla birlikte, iletişim ortamı güvenilir olmadığından birbirleri ile yüz yüze görüşmelidirler. Bu yapılabilirse kullanıcılar gizli anahtarla şifrelenmiş, üçüncü kişilere anlamsız görünen bilgiyi güvenli bir şekilde değiştirebilirler. Fakat Bob ve Alice kendilerine ait olan bu anahtarın bir kripto analizcinin ele geçmesi tehlikesine karşı korumalıdır.

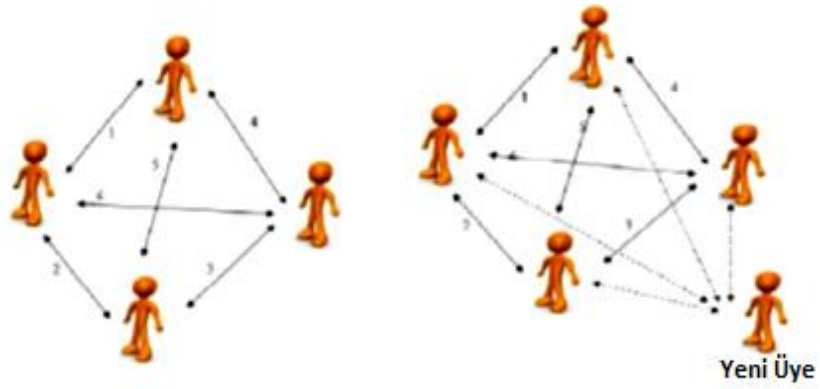
Bob ve Alice başka biriyle, örneğin Jack ile yazışmak isterlerse, Jack'a anahtarı verirlerse onunla da tam bir uzlaşma içinde olmalıdırlar çünkü anahtarı bir kripto analizci ele geçirirse yeni bir kaynağa daha sahip olmuş olacaktır. Tarafların birbirlerine gönderdiği her bir mesajın çözümlenmediğinden emin olmaları için Bob ve Alice, Jack ile iletişimde farklı anahtarlara sahip olmalıdırlar. Böylece her birinde iki anahtar bulunur. Bob, Jack'ın mesajını başka şekilde alır, Alice'e başka türlü iletir ve iletişim böyle sürer.

100 üyeli bir sistem düşünülürse, üyelerin tümü bir diğeri ile gizli bir iletişim kurmak istesin. Bu halde her bireyin iletişim kurduğu herkes için bir anahtara ihtiyacı doğacaktır. Farklı bir deyişle herkes 99 anahtara sahip olacaktır. Her üye de bu 99 anahtarı korumakla zorunludur. Bu şartlar dâhilinde *n* kullanıcılı sisteme yeni üye olanlara n-1 tane anahtar

verilmelidir (Şekil 2.7). Sistemde saklı tutulması gereken anahtar sayısı Denklem 2.10'daki gibi hesaplanır:

Anahtar sayısı = [n * (n - 1)] / 2] (2.10)

Böyle bir sistemde saklanması gereken çok sayıda anahtar olacağından çoklu kullanıcılı ortamlar da asimetrik kripto sistemleri kullanılması daha uygundur [13].



Şekil 2.7. Simetrik sistemde anahtar problemi (a) mevcut sistem(b) yeni üye katılımı[13].

2.4.3.1. DES Algoritması

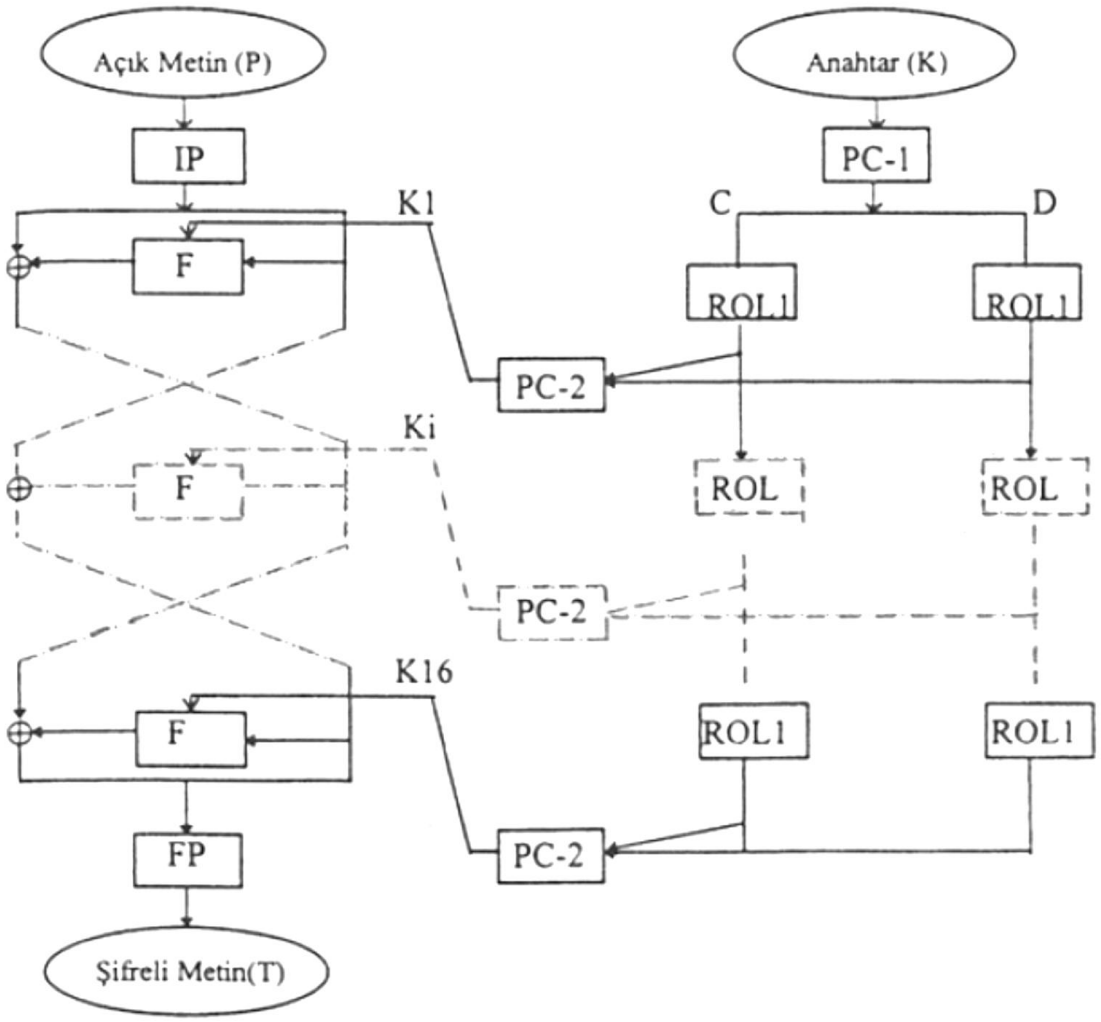
DES algoritması, IBM tarafından bazı hassas verilerin güvenliğini sağlamak için geliştirilmiştir ve 1976'dan beri kullanılmaktadır. Bu algoritma, 64 bit açık metni, 56 bit anahtar altında, 64 bit şifreli metne çeviren blok şifredir. DES tanımında, bit sıralaması soldan sağa 1'den 64'e kadar yapılmıştır.

Algoritmanın ilk kısmında açık metnin bit sıralamasının değiştirildiği giriş (başlangıç) permütasyonu (Initial Permutation, IP), sonunda da şifreli metnin bit sıralamasının değiştirildiği ve giriş permütasyonunun tersi olan çıkış permütasyonu (Final Permutation, FP) bulunur. İki permütasyon arasında yinelemeli çevrimlerden (round) oluşan, algoritmanın ana gövdesi yer almaktadır. Algoritmanın ana gövdesi, veriyi 32 bitlik sağ ve sol yan veri olarak ikiye ayırmaktadır. Algoritmanın temel işlemi "çevrim" olarak adlandırılır. Her bir çevrimde, sağ ve sol yan veriler ile anahtar düzenleme algoritmasından elde edilen 48 bit anahtarlar kullanılarak yeni sağ ve sol yan veriler bulunur.

Genel olarak DES 16 çevrimden oluşur. Her çevrimde verinin sağ yarısı ve her çevrim için farklı değere sahip olan 48 bit anahtar kullanılarak F fonksiyonu hesaplanır. Verinin sol yarısı bu F fonksiyonunun çıkışı ile özel veya işlemine tabi tutulur (XOR). İki çevrim arasında verinin iki yarısı yer değiştirilir ama bu işlem birinci çevrimden önce ve son çevrimden sonra yapılmaz. Algoritmanın 3 çeşidi vardır. Bunlar, *Düz permütasyon, Genişletme permütasyonu* ve *Permütasyonlu seçenekler*'dir. DES'in algoritma yapısı Şekil 2.8'de, giriş permütasyonu Çizelge 2.2'de gösterilmiştir [14,15,16].

Çizelge 2.2. DES giriş permütasyonu [14].

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7



Şekil 2.8. DES ve anahtar düzenleme algoritması [15].

DES'in değiştirme kutuları (S-Box), 6 bitten 4 bite düzenleme tabloları şeklindedir. Her bir S-Box 64 mümkün giriş değerini (6-bit), 16 çıkış değerine (4-bit) karşı düşürür. DES'in standart tanımlamasında S-Box'ları, 0,1,2,…,15 değerlerinin dört ayrı permütasyonu olarak tanımlanır. E genişletmesi ve P permütasyonu Çizelge 2.4'te görülmektedir. 6 bitlik giriş verisinin birinci ve altıncı bitlerinin oluşturduğu 2 bitlik sayı, satır numarasını verirken geriye kalan aradaki 4 bitlik sayı sütun numarasını vermektedir. Satır ve sütun numarasının S1-Box'taki (Çizelge 2.3) kesişme noktasındaki veri, çıkışta görülecek olan 4 bitlik veriyi göstermektedir. Örneğin S bloğuna girecek olan veri 010001 ise satır numarası 01, yani 1'dir. Sütun numarası ise 1000, yani 8'dir. Birinci satır ve sekizinci sütundaki veri 8'dir. Dolayısı ile çıkışta 1000 verisi görülmektedir [15, 16].

Çizelge 2.3. S1-Box [17].

Satırlar \ Sütunlar	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

Çizelge 2.4. E genişletmesi ve P permütasyonu [14].

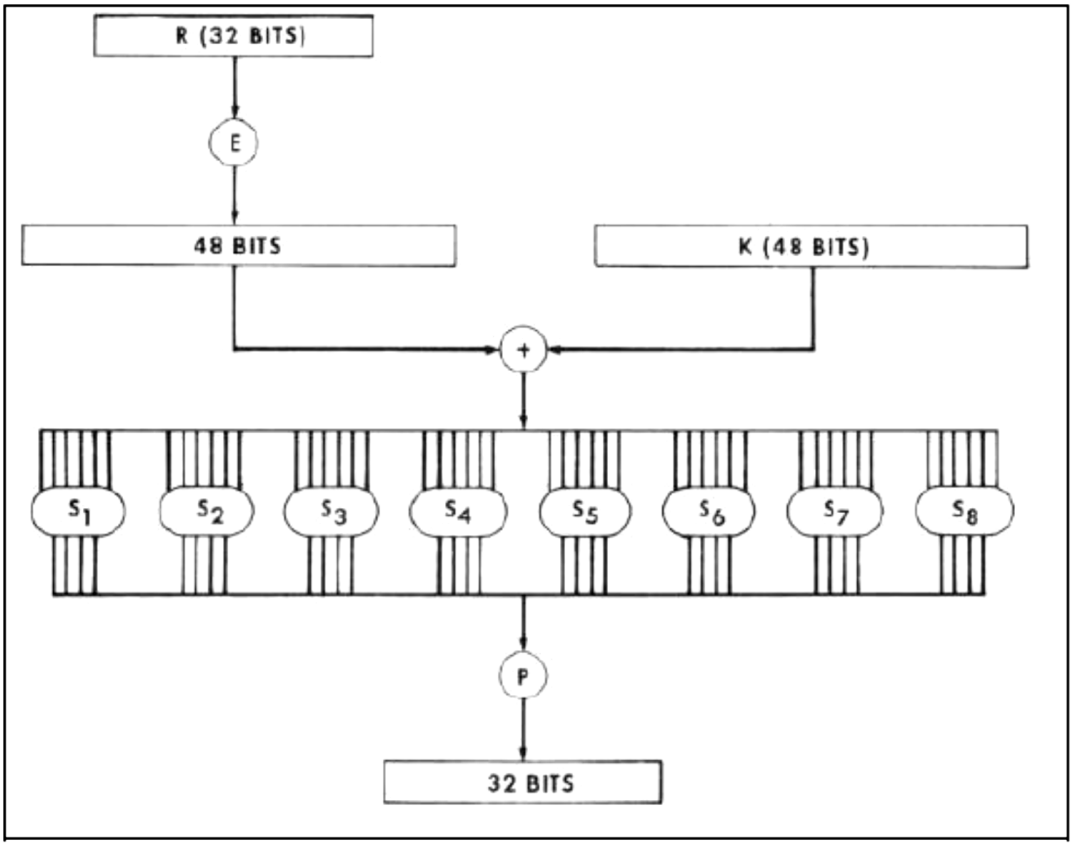
E genişletmesi						P permütasyonu			
32	1	2	3	4	5	16	7	20	21
4	5	6	7	8	9	29	12	28	17
8	9	10	11	12	13	1	15	23	26
12	13	14	15	16	17	5	18	31	10
16	17	18	19	20	21	2	8	24	14
20	21	22	23	24	25	32	27	3	9
24	25	26	27	28	29	19	13	30	6
28	29	30	31	32	1	22	11	4	25

Anahtar düzenleme algoritması, 56 bit anahtardan 16 tane 48 bit *K1, K2, ..., K16* anahtar değerlerini hesaplar. Bu anahtarları DES'in çevrimlerinde *F* fonksiyonu girişi olarak kullanmaktadır (Şekil 2.9). Başlangıçtaki anahtar bitleri, bit numarası sekize bölünenler (8,16,...) eşlik biti olacak şekilde l'den 64'e kadar numaralandırılır. Elde edilen anahtar bitleri Çizelge 2.5'te görülen PC-1 permütasyonu kullanılarak yeniden sıralanır ve 28 bitlik *L* ve *B* kaydedicilerine yüklenir. *L* kaydedicisinin bitleri anahtarın 57, 49, 41,...., 36. bitleri, *B* kaydedicisinin bitleri de anahtarın 63, 55, 47,...., 4. bitleridir. Her bir çevrimde *L* ve *B* kaydedicileri bir ya da iki bit sola kaydırılır. Elde edilen *L* ve *B* kaydedicisi değerleri birleştirilip l'den 56'ya kadar numaralandırılır ve Çizelge 2.5'te gösterilen PC-2 permütasyonu kullanılarak 48 bit anahtar elde edilir. Anahtar düzenleme algoritması Sekil 2.9'da gösterilmiştir [15].

Çizelge 2.5. PC-1 Permütasyon PC-2 Permütasyon [14].

PC-1 Permütasyon						
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

PC-2 Permütasyon					
14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32



Şekil 2.9. DES'in F fonksiyonu [17].

2.4.3.2. AES Şifreleme Algoritması

DES algoritması Ulusal Standartlar ve Teknoloji Enstitüsü (NIST) tarafından Temmuz 1977 yılında veri şifreleme standardı olarak yayınlanmış ve 20 yılı aşkın bir süre yaygın olarak kullanılmıştır. Ancak gelişen teknoloji ve buna bağlı olarak üretilen hızlı işlem

kapasiteli bilgisayarlar DES algoritmasının güvenliği için tehdit oluşturmaya başlamıştır. DES'in 56 bitlik anahtar uzunluğu, gelişen teknoloji ve artan işlemci hızları karşısında güvenilirliğini yitirmeye başlamıştır. Bunun üzerine 1997'de NIST, yeni bir şifreleme standardının geliştirilmesi için bir çalışma başlatmıştır. Geliştirilecek yeni şifreleme standardının mevcut standart olan DES'in yerini alması düşünülmüştür [18,19].

NIST, DES'in yerini alacak yeni şifreleme standardını belirlemek için bir yarışma düzenlenmiş ve Eylül 1997'de algoritmalar için resmi çağrıda bulunmuştur. 1998'de 15 aday algoritmanın değerlendirmeye alındığı duyurulmuş ve 1999'da gerçekleşen seçimlerde algoritma sayısı 5'e indirilmiştir. Seçilen bu 5 finalist algoritma aşağıda sıralanmıştır:

- Çarpma, toplama, rotasyon ve değişiklik (MARS),
- Rivest cipher 6 (RC6),
- Rijndael,
- Serpent,
- Twofish.

Birkaç yıl boyunca süren değerlendirme ve eleme süreçlerinin ardından, NIST 2000 yılında sonucu açıklanmıştır. Joan Daemen ve Vincent Rijmen tarafından tasarlanan, Rijndael algoritmasının AES olarak kullanılacağını ilan etmiştir [19].

NIST tarafından uzun süren bir standardizasyon ve doğrulama sürecinin ardından 26 Kasım 2001 tarihinde AES Federal bilgi işleme standartları (FIPS) 197 standardı olarak yayınlanmıştır [20].

2000 yılında, NIST, bir rapor yayınlamış ve söz konusu raporda ön elemeyi geçen 5 tane algoritmayı çeşitli kategorilerde karşılaştırılmıştır. Çizelge 2.6'da bu beş algoritmanın aldığı puanlar gösterilmektedir (1=düşük, 3=yüksek).

Çizelge 2.6. Finalist 5 algoritmanın farklı kategorilerde karşılaştırılması [19].

	Algoritma				
Kategori	**MARS**	**RC6**	**Rijndael**	**Serpent**	**Twofish**
Genel güvenlik	3	2	2	3	3
Uygulama güvenliği	1	1	3	3	2
Yazılım performansı	2	2	3	1	1
Akıllı kart performansı	1	1	3	3	2
Donanım performansı	1	2	3	3	2
Tasarım özellikleri	2	1	2	1	3

Rijndael algoritmasında blok ve anahtar uzunlukları 128 bitten 256 bite kadar 32 bit aralıklarla birbirinden bağımsız olarak değişebilmekte olmasına rağmen; standart olarak AES algoritması, 128 bit blok uzunluğu ve 128, 192 ve 256 bit anahtar uzunluklarıyla kullanılmaktadır [20].

AES Algoritması genel olarak *Tur İşlemleri* ve tur işlemleri içerisinde gerçekleştirilen *Tur Dönüşüm İşlemleri* basamaklarından oluşmaktadır [18].

AES Algoritması Tur İşlemleri:

Tur: Bir algoritma içerisinde tekrar tekrar yürütülen işlemlerin oluşturduğu yapıya 'Tur' veya 'Döngü' adı verilir.

Durum: 16 baytlık ara şifre değerlerini ifade eder.

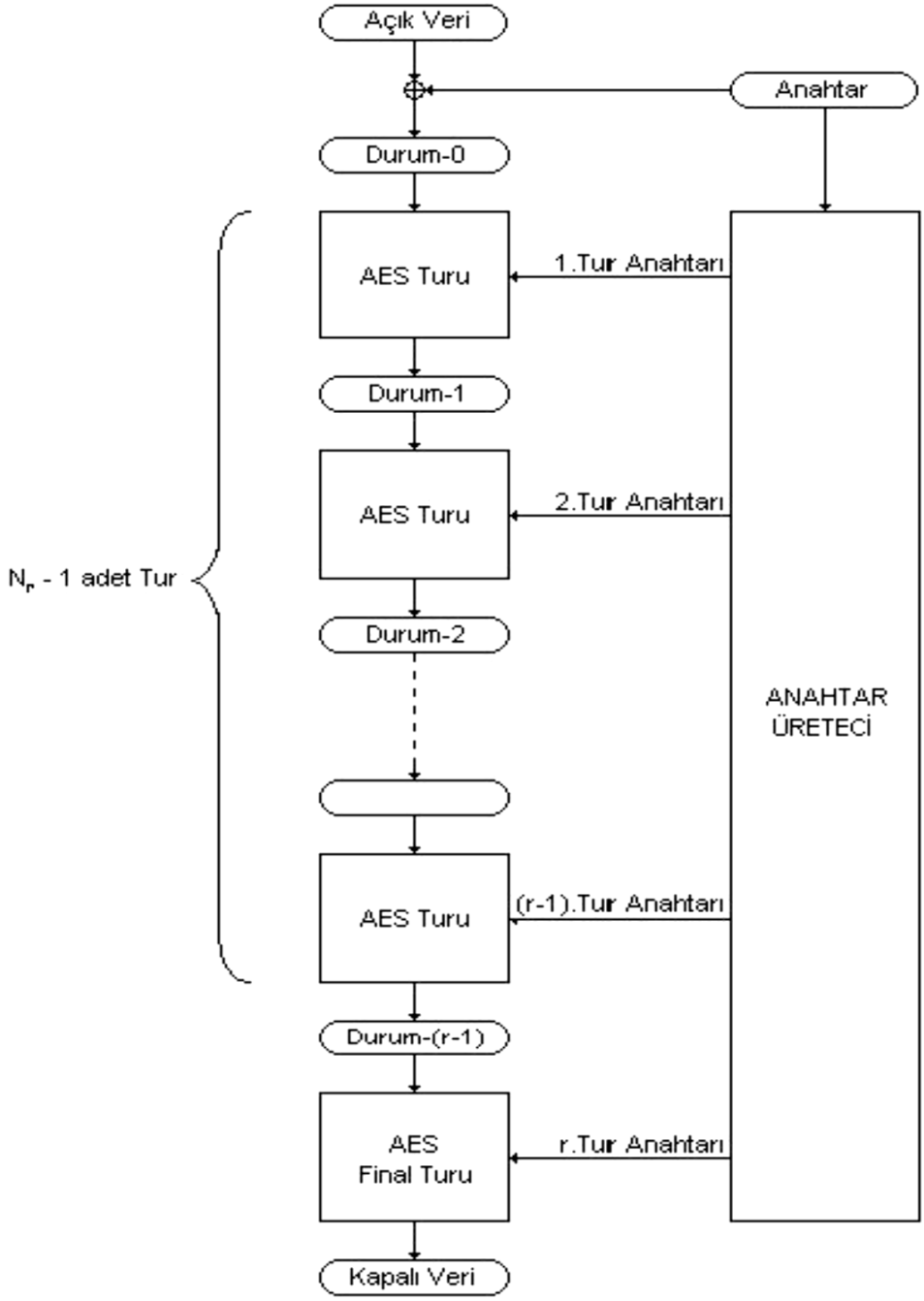
AES, standart olarak 128 bit veri bloklarını işlerken anahtar uzunluğu farklı uzunlukta olabilmektedir. Algoritma içerisinde gerçekleşecek tur sayısı da anahtar uzunluğuna bağlı olarak Çizelge 2.7'deki gibi bir değişme göstermektedir [18].

Çizelge 2.7. Anahtar uzunluğuna göre tur sayıları [20].

AES TİPİ	*Anahtar Uzunluğu (bit)*	*Blok Uzunluğu*	*Tur Sayısı*
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

AES Tur'u 16 baytlık, 'Durum' olarak adlandıran veri bloklarını işler [19].

AES genel olarak iki bloktan oluşur; ilk blok tur dönüşüm, ikinci blok ise anahtar üretim bloğudur. Bu iki blok birbirine paralel olarak çalışmaktadır ve anahtar üretim bloğu her bir tur dönüşüm bloğu için anahtar üretimini gerçekleştirmektedir [18].



Şekil 2.10. AES algoritması blok diyagramı [21].

AES Algoritması Tur Dönüşüm İşlemleri:

AES algoritması tekrarlı bir yapıdan oluşur. Tur dönüşüm işlemi anahtar uzunluğuna bağlı olarak (10,12 veya 14) çok defa tekrarlanır. Tur dönüşüm işlemleri içerisinde; *Bayt Değiştirme, Satırları Kaydırma, Sütunları Karıştırma* ve *Tur Anahtarını Ekleme* işlemleri gerçekleştirilmektedir. Bu işlemlerin her birine '*Adım'* olarak isim verilir. Final turu haricindeki tüm turlarda bu dört adım sırası ile tekrar edilir ve final turunda ise *Sütunları Karıştırma* işlemi gerçekleştirilmez.

Kodlama işleminin yapılabilmesi için öncelikle 16 baytlık kodlanacak verinin *Durum Tanımlama* işlemi yapılır. *Durum Tanımlama*, 16 bayt uzunluğundaki veriyi 4x4'lük matris haline getirme işlemidir. Bu işlem Şekil 4.2'de gösterildiği gibi yapılmaktadır ve bu işlem yapıldıktan sonra diğer tur dönüşüm işlemleri yapılacaktır.

Çizelge 2.8. Giriş verisi için durum tanımlama işlemi

D_0	D_4	D_8	D_{12}
D_1	D_5	D_9	D_{13}
D_2	D_6	D_{10}	D_{14}
D_3	D_7	D_{11}	D_{15}

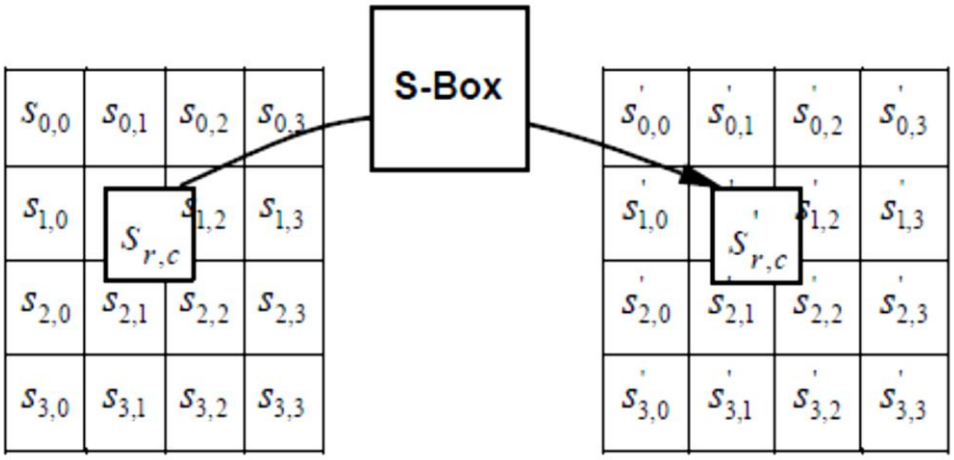
Bayt Değiştirme:

Bayt Değiştirme dönüşümü, giriş verisindeki durumun her bir baytını, Şekil 11'de görülen *S-Box* adı verilen bir değiştirme tablosu kullanarak başka bir bayta dönüştürür [18].

		y														
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Şekil 2.11. S-Box [22].

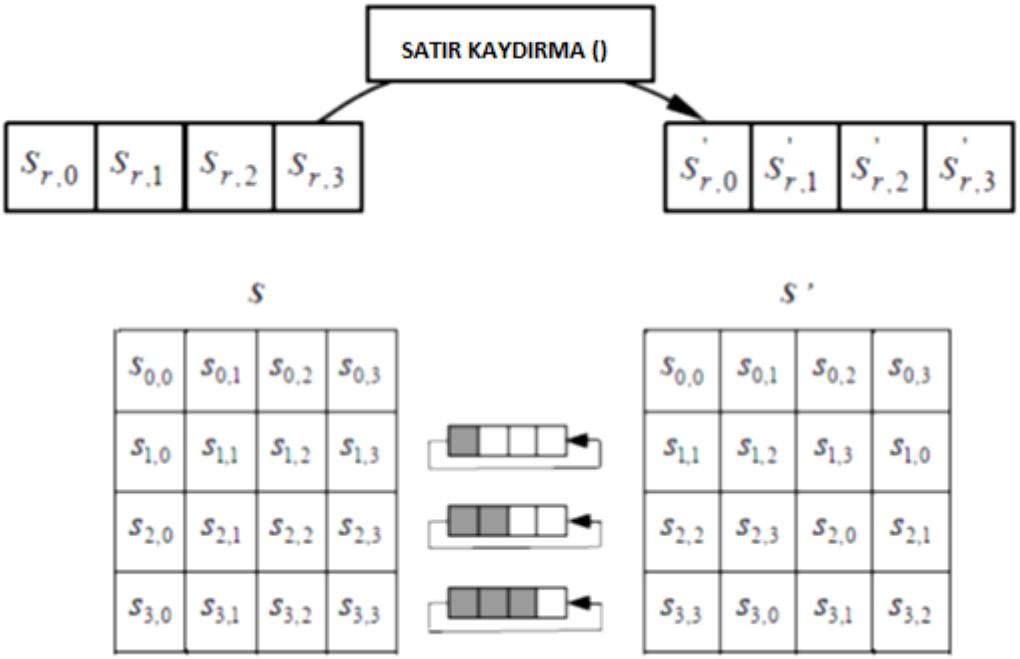
Bayt Değiştirme işlemi gerçekleştirilecek bayt hexadecimal olarak ifade edilir. Elde edilen sonucun en önemli 4 bit kısmı S-Box tablosunda bakılacak olan satır değerini ve en önemsiz 4 bit kısmı sütun değerini gösterir. S-Box üzerinde, bakılan satır ve sütun değerlerinin kesiştiği hücre içerisindeki bulunan değer Bayt Değiştirme (Şekil 2.12) işleminin sonucu olarak alınır ve bayt yer değiştirme işlemine giren sayının yerine yerleştirilir.



Şekil 2.12. Bayt değiştirme işlemi [22].

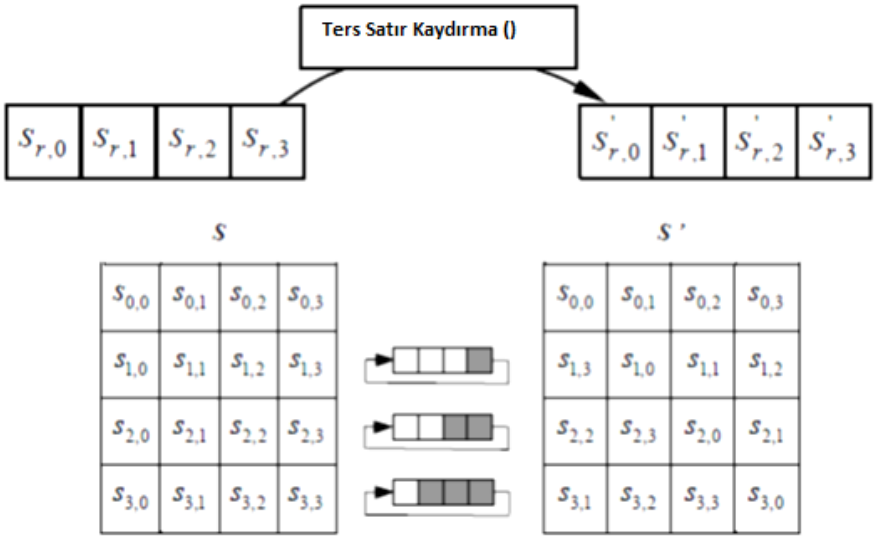
Satırları Kaydırma:

Bayt Değiştirme işlemi sonucunda elde edilen çıkış verisi *Satırları Kaydırma* işleminin giriş verisini oluşturmaktadır. Satır kaydırma işleminde ilk satır aynı bırakılır. İkinci satır sağdan sola doğru bir pozisyon, değiştirecek şekilde, döngüsel olarak, kaydırılır. Döngüsel kaydırma nedeniyle, 1. sütuna gelen eleman kaydırıldığında 4.sütuna geçer. Üçüncü satır benzer şekilde iki pozisyon, dördüncü satır da üç pozisyon döngüsel olarak kaydırılır. Satırları kaydırma dönüşümünün nasıl değiştirdiği Şekil 2.13'te gösterilmiştir [18].



Şekil 2.13. Satırları kaydırma [22].

Ters Satır kaydırma, işleminde ise Şekil 2.14'te gösterildiği gibi yapılmaktadır.

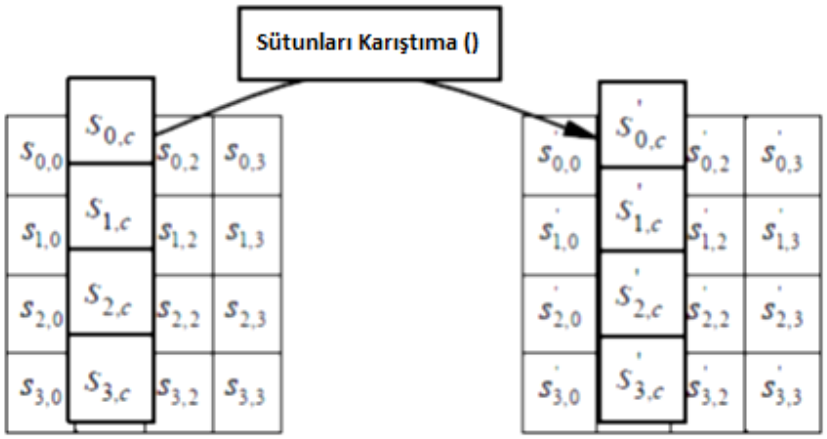


Şekil 2.14. Ters satırları kaydırma [22].

Sütunları Karıştırma:

Sütunları Karıştırma işlemi (Şekil 2.15) Durum matrisindeki her bir sütun üzerinde bağımsız olarak gerçekleştirilir ve bu işlem gerçekleştirilirken her bir satır $GF^*(2^8)$'de bir polinom olarak düşünülür. Her bir satır üzerinde bir $A(x)$ polinomu ile modülo(x^4+1)'de çarpma işlemi gerçekleştirilir. Çarpma için kullanılan polinom:

$$A(x)=\{03\}x_3+\{01\}x_2+\{01\}x_1+\{02\} \tag{2.11}$$

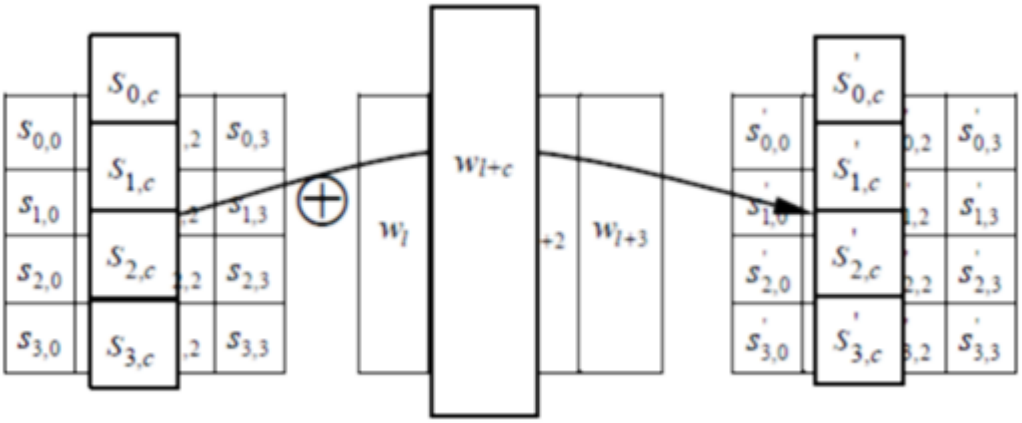


Şekil 2.15. Sütunları karıştırma işlemi [22].

* GF(2^n) Galois alanı hesabı için bakınız: Öztemür, M., AES algoritmasının bir gerçeklemesine güç analizi saldırıları, *Yüksek Lisans Tezi* Yıldız Teknik Üniversitesi, (**2012**).

Tur Anahtarı Ekleme İşlemi:

Tur Anahtarını Ekleme işleminin yapılabilmesi için öncelikle *Tur Anahtarı Matrisi*'nin Şekilde gösterildiği gibi oluşturulması gerekir. Bu adımdan sonra *Sütunları Karıştırma İşlemi* sonunda elde edilen durum matrisi ile anahtar matrisi XOR işlemine tabi tutulur. Şekil 2.16'da *Anahtar Ekleme* gösterilmektedir [18].



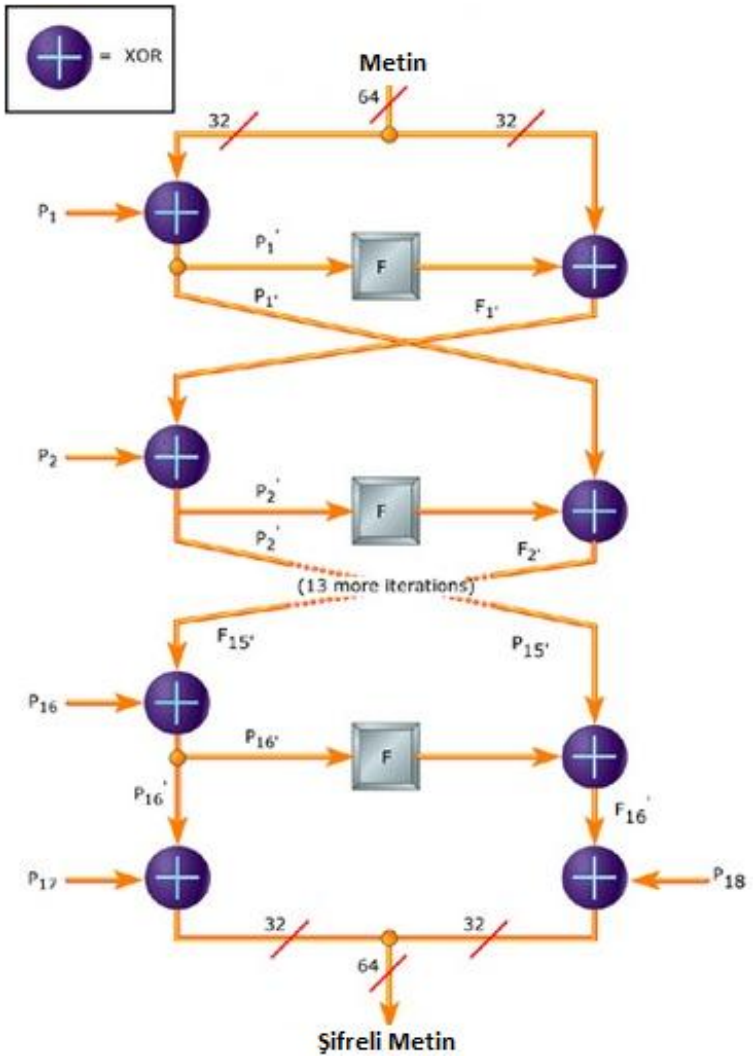
Şekil 2.16. Tur Anahtarı Ekleme İşlemi [22].

2.4.3.3. Blowfish Algoritması

Blowfish algoritması, IBM tarafından geliştirilen Feistel Ağını kullanan bir blok şifreleme yöntemidir. Bu algoritma, Bruce Schneier tarafından 1993 yılında tasarlanmış, simetrik bir blok şifreleyicidir. Blowfish ile ilgili olarak şu ana kadar etkin bir şifre çözme analizi mevcut olmasa da, günümüzde AES ya da Twofish gibi daha büyük ebatlı blok şifreleyicilerine daha fazla önem verilmektedir. Schneier; Blowfish'i bir genel kullanım algoritması olarak, eskiyen DES'in yerini alması için ve diğer algoritmalarla yaşanan sorunlara çözüm olarak tasarlamıştır. Tasarımın belirgin özellikleri anahtar-bağımlı S-Box ve oldukça karmaşık anahtar çizelgesini içermesidir. Aynı anahtarı hem şifreleme hem de şifre çözmede kullandığı için simetrik algoritmadır. Blowfish şifrelemesinde 16 adımdan oluşan Feistel Ağı kullanılmaktadır. Bu ağdaki mesaj boyutu 64 bit ve anahtar boyutu 32 ile 448 bit arasında değişkendir. Blowfish algoritması, sabit S-boxes kullanan CAST-128 yapısına benzemektedir [15,16, 23,24].

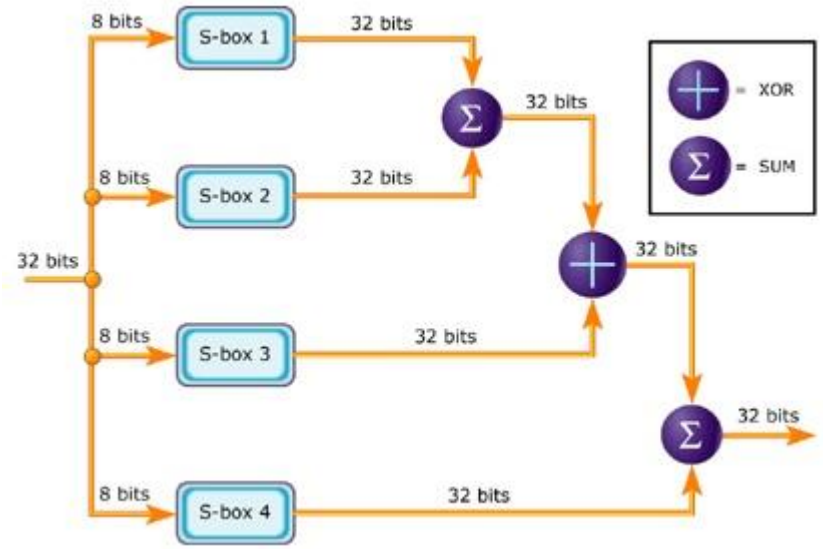
Blowfish' in anahtar çizelgesi P-sıralarını ve S-Box'ları, hiç bir belirgin şablon içermeyen; pi sayısının hexadecimal dijitlerinden türetilen değerlerle başlangıç konumuna getirerek başlar. Pi sayısının kullanılma nedeni, basamaklarını oluşturan

sayıların rastgelesellik özelliğine sahip olmasıdır. Şekilde toplam 16 adımdan oluşan Blowfish algoritmasının blok diyagramı görülmektedir. Algoritmaya göre her adımda 32 bitlik işlem yapılmaktadır. Algoritma iki alt anahtar sırası tutar: 18-girisli P-sırası ve dört 256-girisli S-Box bulunmaktadır. S-Box 8-bit girdi kabul eder ve 32-bit çıktı oluşturur. P-sırasının bir girişi her turda kullanılır ve son turdan sonra veri öbeğinin her bir yarısı geri kalan kullanılmamış iki P-girişinden biri tarafından XOR'lanır. Her adımda yer değiştirme dizilerinden bir tanesi kullanılmaktadır. Son adımdan sonra veri bloğunun her iki yarısının XOR'lanmış hali alınmaktadır ve bu işlem sırasında arta kalan iki yerine koyma dizisi de kullanılmaktadır. 16 adımın her birisi için birer yerine koyma dizisi ve son adımda da 2 yerine koyma dizisi kullanıldığı için toplam 18 adet dizi bulunur. Algoritmanın blok diyagramı Şekil 2.17'de görülmektedir [15,16,24].



Şekil 2.17. Blowfish algoritması bloğu [25].

Şekil 2.18'de her F-Fonksiyonu için kullanılan yöntem verilmiştir. Buna göre mesajın yarısı olan 32 bit uzunluğundaki veri 4 adet 8 bitlik parçaya ayrılarak, S-Box'lara yerleştirilmekte ve her S-Box'tan çıkan sonuç Şekildeki gibi uygulanmaktadır [15].



Şekil 2.18. Blowfish F Fonksiyonu şeması [25].

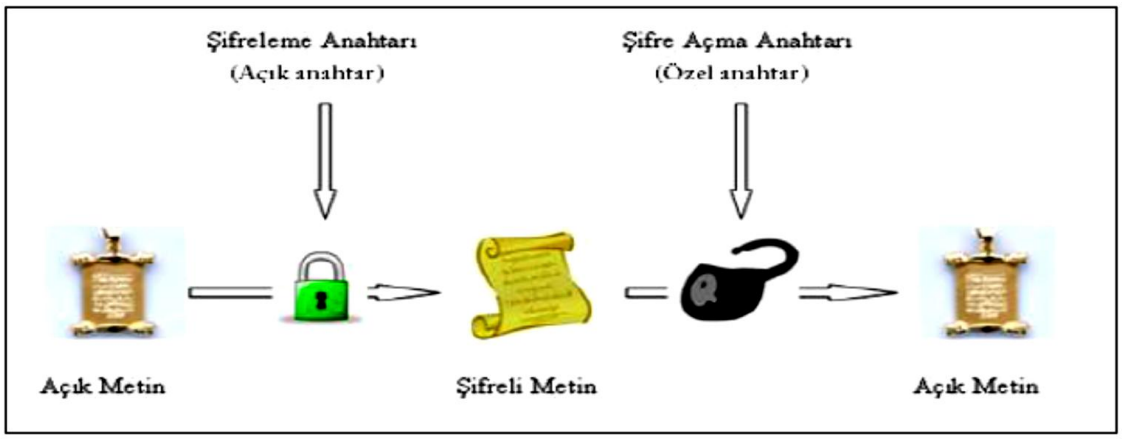
Sonuçlar 2^{32} değerinde modulo işlemi alınarak yapılmakta ve 32 bitlik sonucu üretmek için XOR işlemine tâbî tutulmaktadır. 17. ve 18. adımlarda kullanılan permütasyonun terse çevrilmesi ve her adımda bulunan permütasyonun sırasıyla geriye dönülmesi ile şifreleme yönteminin açılması için mümkündür [15,23].

2.4.4. Asimetrik Şifreleme

Asimetrik şifreleme algoritmaları, simetrik şifreleme algoritmalarından radikal farklılıklarla ayrılmaktadır. Çizelge 2.9'da bu farklılıklar gösterilmektedir. Asimetrik şifreleme algoritmalarında açık (public) ve özel (private) anahtar olmak üzere iki ayrı anahtar kullanılmaktadır.

Asimetrik algoritmalar da şifreleme için kullanılan anahtar ile şifre çözme için kullanılan anahtar birbirinden farklıdır. Asimetrik şifreleme algoritmalarının anahtarının halka (genel kullanıma/kamuya) açık olmasından ötürü; bu algoritmaya açık anahtarlı algoritmalarda denilmektedir. Anahtar çiftlerini üreten algoritmaların matematiksel özelliklerinden dolayı açık-gizli anahtar çiftleri her kişi için farklıdır. Bu algoritmada, her kullanıcının açık-gizli anahtar çifti yalnızca o kullanıcıya özel şekilde üretilir.

Bir kullanıcıya ait gizli anahtarı, yalnızca kendi kullanımı içindir ve başkalarının eline geçmemesi gerekir. Bu kullanıcının açık anahtarı ise, bu şahsa mesaj göndermek isteyen herhangi biri tarafından kullanılabilir. Gönderici, mesajı alıcının açık anahtarı ile şifreler. Alıcı, gelen mesajı kendi özel anahtarı ile açar ve bu sistem Şekil 2.19'da gösterilmektedir.



Şekil 2.19. Asimetrik şifreleme [13].

Asimetrik anahtarlı algoritmalarda önemli bir nokta da şifre çözüm anahtarının (en azından makul bir zaman dilimi içerisinde) şifre anahtarından hesaplanamaz olmasıdır.

Çizelge 2.9. Simetrik ve asimetrik şifreleme algoritmalarının özellikleri [26].

Simetrik şifreleme algoritmaları	**Asimetrik şifreleme algoritmaları**
Aynı algoritma ve aynı şifreleme anahtarı hem şifreleme hem de şifre çözmede kullanılır.	*Şifreleme ve şifre çözmek için bir algoritma fakat şifreleme ve şifre çözme için farklı anahtarlar kullanılır.*
Gönderici ve alıcı aynı algoritmayı ve aynı anahtarı kullanır.	*Gönderici alıcının açık anahtarını bilmelidir. Gönderici ile alıcının anahtar çiftleri birbirinden farklıdır.*
Şifreleme için kullanılan algoritma gizli tutulmalı	*İki anahtardan biri gizli tutulmalı diğeri erişime açık olmalıdır.*
Algoritma bilgisi ve şifreli metin örnekleri anahtarı belirlemede yeterli olmamalı.	*Algoritma bilgisi, anahtarlardan birinin ve şifreli metin örnekleri, diğer anahtarı belirlemede yeterli olmamalı.*

Asimetrik algoritmalar, simetrik algoritmalara göre daha güvenli ve kırılması zor algoritmalardır. Bununla birlikte, performansları simetrik algoritmalara göre oldukça düşüktür [13].

2.4.4.1. RSA Algoritması

RSA algoritması, 1977 yılında Ron Rivest**,** Adi Shamir ve Len Adleman tarafından bulunmuştur ve soy isimlerinin ilk harfleri ile isimlendirilmiştir. RSA, çok büyük tamsayıları oluşturma ve bu sayıları işlemenin zorluğu üzerine düşünülmüştür. Anahtar oluşturma işlemi için asal sayılar kullanılarak daha güvenli bir yapı oluşturulmaya çalışılmıştır. RSA açık anahtarlı şifreleme yöntemlerindendir. Hem mesaj şifreleme hem de elektronik imza için kullanılmaktadır [15,16, 27].

Günümüzde, RSA şifreleme algoritması halen güvenilirliğini korumaktadır. Bunun nedeni modüler matematik üstüne kurulmuş, kripto analizi asal sayılara, çarpanlara ayırmaya dayalı anlaşılması kolay ama çözülmesi zor bir algoritma olmasından kaynaklanmaktadır [13].

Son derece basit matematiksel ilişkilerle çalışan bu yöntem de iki ayrı anahtar bulunmaktadır. Anahtarlardan birisi gizli diğeri açık anahtardır. Herkes açık anahtarını yayınlar ve kendisine şifreli bir mesaj göndermek isteyen birisi bu anahtarı kullanarak mesajı şifreler ve gönderir. Ancak mesajı sadece gizli anahtar kimde ise o çözebilir. Gizli anahtar ise sadece sahibinde bulunur. Böylece, herkes çözüm için gerekli anahtarı bilmeden, şifreyle mesajlarını gizleyebilmektedir.

Daha önce hiç karşılaşmamış, birbirini tanımayan kişiler bile birbirlerine gizli mesajlar gönderebilir. Örneğin internetten alışveriş yapmak isteyen bir müşteri, kendisini hiçbir şekilde tanımayan bir web sitesine girerek, sitenin açık anahtarını alır, kredi kartı numarasını bu anahtarla şifreleyerek gönderir. Şifreli bilgiyi gönderen dahil hiç kimse bu mesajı çözemez, sadece web sitesinde bulunan gizli anahtarla gelen kart numarasını web sitesi çözebilir. Böylece müşteri kredi kart numarasının başkaları tarafından okunmayacağından emin olacaktır [15].

RSA şifreleme algoritması sayısal imza içinde kullanılabilmektedir. Mesajı imzalayan taraf öncelikle açık ve özel anahtarları üretir. Daha sonra mesajı kendi özel anahtarı ile şifreleyerek orijinal mesaj ile birlikte karşı tarafa gönderir. Şifreli mesajı alan taraf mesajı karşı tarafın açık anahtarı ile mesajı deşifre ederek orijinal mesaj ile karşılaştırır eğer mesajlar eşit ise imza doğrulanmış olur [27].

Aşağıda açıklanan RSA şifreleme algoritmasının çalışması Şekil 3.1.'de gösterilmiştir.

Anahtar üretimi:

- Önce kullanıcı anahtarları oluşturulur.
  · *n/2* bit büyüklüğünde iki asal sayı *p ve q* seçilir.
  · İki asal sayının *(p ve q)* çarpımından *n bit* büyüklüğünde *N* sayısı hesaplanır.
  · Denklem 2.12 ile hesaplanan *Φ (n)* ile ortak böleni bir olan ve $1 < e < \Phi(n)$ koşulunu sağlayan açık anahtar *e* sayısı seçilir.

$$\Phi(n) = (p-1)(q-1) \tag{2.12}$$

  · $1 < d < \Phi(n)$ koşulunu sağlayan özel anahtar *d*, *k* keyfi bir tamsayı olmak üzere 2.13 denkleminden hesaplanır.

$$ed = k\,\Phi(n) + 1 \tag{2.13}$$

- Bu değerlerden *(e,n)* açık anahtar olarak yayınlanır.
- Bu değerlerden *(d,n)* özel anahtar olarak elde tutulur. *d* değeri kimse ile paylaşılmaz.

Şifreleme:

- Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür.
- Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin *(M)* bloklarını şifreler ve elde ettiği şifreli değerleri *(C)* karşıya gönderir.

$$C = M^{e} \bmod N \tag{2.14}$$

Şifre açma:

- Şifreli mesajı alan kullanıcı özel anahtarı ile şifreli değeri açar ve gerçek mesaja ulaşır.

$$M = C^{d} \bmod n \tag{2.15}$$

Örneğin; “Steganografi” yazısının RSA ile şifrelenmesi Şekil 2.20’de gösterilmiştir. Bu işlemin basamakları şu şekildedir:

Anahtar üretimi:

- Önce kullanıcı anahtarları oluşturulur.
  · İki asal sayı *p=73* ve *q=151* seçilir.
  · İki asal sayının *(p ve q)* çarpımından *N=7x151=11023* sayısı hesaplanır.
  · Denklem 2.12’den $\Phi(n) = 72x150 = 10800$. $\Phi(n)$ ile ortak böleni bir olan açık anahtar *e=11* sayısı seçilir.

$$1 < e < \Phi(n) \tag{2.16}$$

$$d \equiv e^{-1} \bmod \Phi(n) \tag{2.17}$$

  · Denklem 2.17’den özel anahtar *d=5891* hesaplanır.

$$1 < d < \Phi(n) \tag{2.18}$$

- *(e=11,N=11023)* açık anahtar olarak yayınlanır.
- *(d=5891,N=11203)* özel anahtar olarak elde tutulur. *d* değeri kimse ile paylaşılmaz.

Şifreleme:

- Şifrelenecek olan açık metin öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür.

ST	EG	AN	OG	RA	Fİ
1320	1119	2317	2111	2123	0513

- Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin *(M)* bloklarını şifreler ve elde etiği şifreli değerleri *(C)* karşıya gönderir.

$$C = M^{e} \bmod n \tag{2.19}$$

$C1 = 1320^{11} \bmod 11023 \rightarrow 10124$

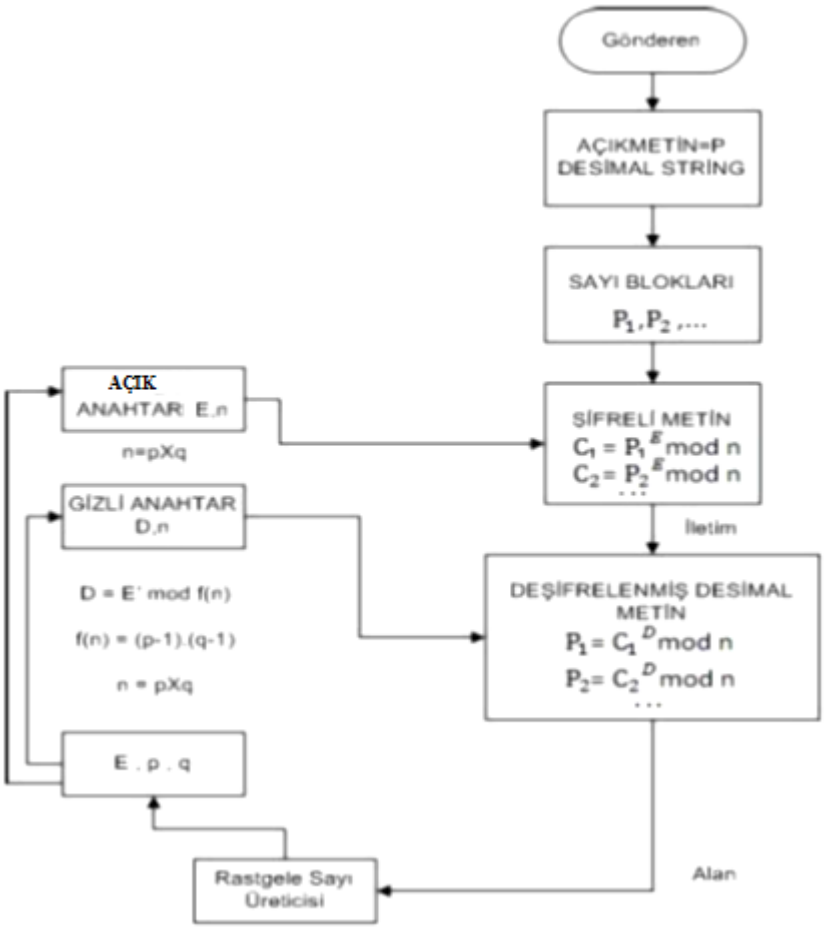
$C2 = 1119^{11} \bmod 11023 \rightarrow 5618$; …

Şifre açma:

- Şifreli mesajı alan kullanıcı özel anahtarı ile şifreli değeri açar ve gerçek mesaja ulaşır [13,28].

$$M = C^{d} \bmod n \tag{2.20}$$

$M_1 = 10124^{11} \bmod 11023 \rightarrow 1320 \rightarrow ST$; …..



Şekil 2.20. RSA algoritması blok diyagramı [15].

2.5. VERİ SIKIŞTIRMA

Sıkıştırma işlemi, verinin fiziksel boyutunu azaltmak için kullanılan bir işlemdir. Sıkıştırma işleminin ana amaçları arasında; aynı ortam üzerinde daha fazla bilgi depolayabilmek, bir ağ üzerindeki disk boyutunu veya gönderim zamanını ve bant genişliğini azaltabilmek ve veriyi daha sonra tekrar kullanabilmek sıralanabilir.

Sıkıştırma işleminde amaç, verilen türdeki verinin fazlalıklarını azaltmaktır. Veri sıkıştırma algoritmaları genellikle kayıplı ya da kayıpsız olarak sınıflandırılırlar. Kayıpsız veri sıkıştırma işlemi, orijinal veri setinin dönüşümü sonrasında gerçekleştirilen çözme işlemi sonucu orijinal verinin birebir üretiminin mümkün olmasını gerektirmektedir. Kayıpsız sıkıştırma, orijinal verinin önemli olduğu ve orijinal veri ve çözülen veri dosyalarının özdeş olması gerektiği durumlarda kullanılmaktadır. Kayıplı veri sıkıştırma algoritması, orijinal veri setinden verinin tekrar üretiminin mümkün olmadığı bir dönüşümdür ve çözme işlemi ile ancak orijinal veriye yakın bir veri seti oluşturulabilir. Bu türdeki sıkıştırma internet ve özellikle duraklamasız medya ortamları ve mobil ortam uygulamalarında kullanılmaktadır. Bu noktada, şu üç tanımlama gerçekleştirilebilir:

TANIM 1. Sıkıştırma, verilen bir D kaynak bilgisinden, daha kısa bir $\Delta(D)$ bilgisini üreten bir süreçtir.

TANIM 2. Kayıpsız sıkıştırma, $\Delta(D)$ bilgisinden D kaynak bilgisinin birebir çıkarılabildiği bir işlemdir. Kayıplı sıkıştırma ise, orijinal verinin yaklaşık olarak kodlanabildiği bir metottur.

Kayıplı ve kayıpsız sıkıştırma teknikleri arasındaki ayırım önemlidir çünkü kayıplı sıkıştırma metotlarında sıkıştırma oranı kayıpsız sıkıştırma metotlarına göre daha fazladır. Kayıpsız sıkıştırma metotları genellikle 2:1 oranından 8:1 oranına kadar sıkıştırma oranı elde edebilirler. Kayıplı sıkıştırma metotları ise, 100:1 oranından 200:1 oranına kadar sıkıştırma sağlayabilirler. Buna ilaveten, orijinal veride hatalara ne kadar fazla tolerans gösterilirse, o kadar fazla sıkıştırma oranı elde edilir. Ayrıca özellikle kayıplı sıkıştırma metotlarında, sıkıştırma verimi, kaynak bilginin karakteristiğinden önemli derecede etkilenmektedir.

Veri sıkıştırma tekniklerinin üç ana modeli mevcuttur: İstatistiksel, Sözlük tabanlı sıkıştırma ve Yer değiştirme sıkıştırması. İstatistiksel sıkıştırma teknikleri, karakterlerin

hesaplanan olasılıklarına göre en kısa ortalama kod uzunluğunun üretimini içermektedir. Bu tür sıkıştırma tekniklerinde, kaynak dosyadaki karakterler ikili koda dönüştürülür. Dosyadaki en genel karakterler en kısa ikili kodu alırken, en az genel olan karakterler en uzun ikili kodu almaktadır. Sözlük tabanlı sıkıştırma metotlarında ise metindeki karakterler, oluşturulan bir sözlüğe göre indis veya işaretçi kodu ile gösterilirler (LZW). Yer değiştirme sıkıştırması, tekrar eden karakterlerin tümü için daha kısa bir ifade kullanılmasını içermektedir.

Çoğu sıkıştırma algoritması, sıkıştırma oranını artırmak için farklı veri sıkıştırma tekniklerinin kombinasyonlarını kullanmaktadır. Sıkıştırma miktarı, sıkıştırma oranı olarak bilinen bir C faktörü ile ölçülmektedir:

$$C = \frac{So}{Sc} \tag{2.21}$$

Burada *Sc* ise sıkıştırılan dosya boyutu, *So*; orijinal dosya boyutudur. Sıkıştırma oranı *C:1* şeklinde ifade edilmektedir. Sıkıştırma miktarı aynı zamanda orijinal veri miktarındaki azalma ile de ölçülebilmektedir:

$$R = \frac{So - Sc}{So} \tag{2.22}$$

R genellikle yüzde ile ifade edilmektedir [4].

2.5.1. LZW Algoritması

LZW algoritmasını açıklamadan önce dizi eşleşme problemine (string matching problem) bakalım:

TANIM 3. Verilen bir $P = p_1, \ldots, p_m$ örüntüsü ve $T = t_1, \ldots, t_u$ metni için *T* içinde mevcut olan tüm *P* örüntülerini bul ve $\{|x|, T = xPy\}$ kümesini döndür.

Veri; metin, görüntü, ses gibi değişik formatlarda olabilir. Veri sıkıştırma açısından, bunların arasında önemli bir farklar bulunmaktadır. Görüntü ve ya ses dosyalarında orijinal bilginin yakın bir tahmini yeterli olmaktadır. Metinsel bilgide sıkıştırma/çözme işlemi gerçekleştirme durumunda orijinal verinin birebir olarak geri elde edilmesi gerekmektedir. Metin sıkıştırmasında daha az yer kullanılarak verinin temsili amacıyla

fazlalıklardan yararlanılmaktadır. Ziv–Lempel ailesi başarılı sıkıştırma oranı ile verimli sıkıştırma ve çözme zamanı sebebiyle günümüzde en yaygın olarak kullanılan sıkıştırma tekniklerinden birisidir.

LZW kodlama tekniğinde hazır halde bir sözlük bulunmamaktadır. LZW sıkıştırma algoritması öncelikle veriyi okur ve sözlükte kodlanan bir diziden yararlanarak mümkün olduğunca büyük veri biti serisi ile eşleşme yapmaya çalışır. Eşleşen veri sırası ve bunu izleyen karakter sonraki veri serilerinin kodlanması amacıyla birlikte gruplandırılarak sözlüğe eklenir. Daha küçük, sıkıştırılmış bir kod daha yüksek sıkıştırma oranıyla sonuçlanırken, bu durum sözlük boyutunu da sınırlandırmaktadır. Algoritmanın işleme şekli aşağıda gösterilmiştir:

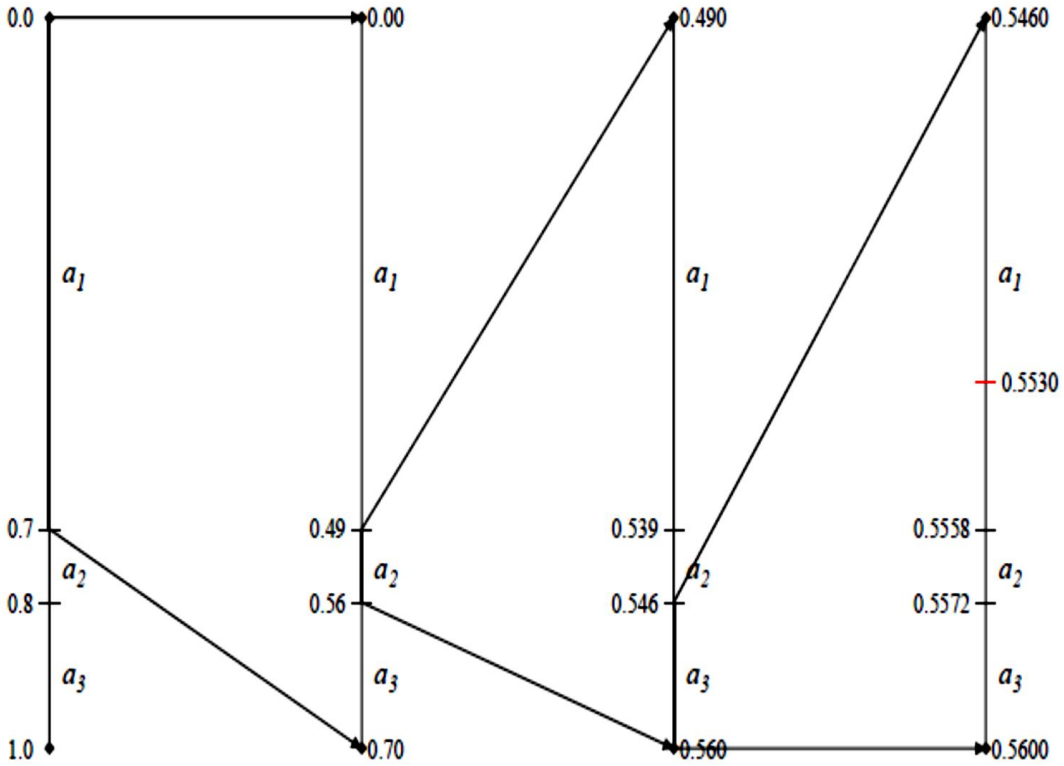
1. *C* veri dizisindeki bir sonraki karakter olsun.
2. *P* + *C* dizisi sözlükte var mı?
    2.1. Eğer var ise, *P* ← *P* + *C* (*P*'yi *C* ile genişlet)
    2.2. Yok ise
        2.2.1. *P* kod kelimesini, kod dizisine çıkış olarak ver
        2.2.2. *P* + *C* dizisini sözlüğe ekle
        2.2.3. *P* ←*C* (*P* bu durumda sadece *C* karakterini içermektedir.)
3. Veri dizisinin sonuna gelindi mi?
    3.1. Hayır ise 2. Adıma git
    3.2. Evet ise *P* kod kelimesini, kod dizisine çıkış olarak ver [4].

2.5.2. Aritmetik Kodlama

Alfabenin küçük olduğu ve karakterlerin belirme olasılığında büyük farklar olduğu durumlarda, belirli sıralamaya sahip sembollere kod atayarak aynı uzunluktaki tüm olası sıralamalara kod oluşturma zorunluluğu getirmeyen bir yönteme ihtiyaç duyulduğunda ilk akla gelecek olan kodlama Aritmetik kodlamadır diyebiliriz. Aritmetik kodlama teorisinin ispatına çok yakın bir kavramdan Shannon 1948'deki makalesinde bahsetmiştir. Bu teori, Abramson'un bilişim teorisi üzerine yayınladığı kitabında da yer almaktadır. Daha sonra Jelinek tarafından yazılan bir başka kitapta ise, aritmetik kodlama fikri değişken uzunluklu kodlamanın bir örneği olarak yer almıştır. Sonlu duyarlık sorununun çözülmesi, modern aritmetik kodlamanın başlangıcı olarak kabul edilebilir [29,30,31].

Aritmetik kodlamanın temel fikri, *n* adet mesajın her olası serisini temsil etmek için 0 ile 1 arasındaki bir sayı aralığını (örneğin 0,3 ile 0,6 aralığı gibi) kullanmaktır. Alfabenin küçük olduğu ve karakterlerin belirme olasılığında büyük farklar olduğu durumlarda; Aritmetik kodlamanın, Huffman kodlamasına göre daha başarılı olduğu görülmüştür. Aritmetik kodlamada, belirli bir sembol serisini diğer sembol serilerinden ayırmak için, her serinin tekil bir belirleyicisi ile etiketlenmesi gerekmektedir. Bu etiket, genellikle 0 ile 1 arasında bir sayı şeklinde belirlenmektedir [29,30].

Örneğin, üç harfli bir $A = \{ a_1, a_2, a_3 \}$ alfabesini ele alalım. Harflerin olasılık dağılımları $P(a_1) = 0.7$, $P(a_2) = 0.1$ ve $P(a_3) = 0.2$ olsun. Eğer a_1, a_2, ve a_3 harflerinin ardı ardına geldiği bir seri için etiketin bulunması istenirse, öncelikle 0 ile 1 sayıları aralığı olasılık dağılımları baz alınarak 3 farklı parçaya bölünür. 0 ile 0.7 arası a_1, 0.7 ile 0.8 arası a_2, ve 0.8 ile 1 arası da a_3 için tahsis edilir. Serinin ilk karakteri a_1 olduğu için, ikinci karaktere geçilirken, a_1'e tahsis edilen aralık yani 0 ile 0.7 aralığı genişletilir. Yeni aralık tekrar olasılık dağılımları baz alınarak 3 farklı parçaya bölünür. 0 ile 0.49 arası a_1, 0.49 ile 0.56 arası a_2, ve 0.56 ile 0.7 arası da a_3 için tahsis edilir. Serinin ikinci karakteri a_2 olduğu için 0.49 ile 0.56 aralığı seçilerek işleme devam edilir. Serinin üçüncü karakteri olan a_3 karakteri 0.546 ile 0.56 arasını tahsis edecektir ve serinin etiketi olarak bu aralıkta yer alan herhangi bir sayısal değer seçilebilir.



Şekil.2.21. A alfabesinin a_1, a_2, a_3 sırası için aritmetik kodlamada etiketin bulunması[29].

Şekil 2.21'de bu işlemin adımları soldan sağa doğru gösterilmektedir. Şekilde görüldüğü gibi bu örnekte en son aralığın orta noktası olan 0.553 değeri etiket olarak seçilmiştir. İkili sayı sistemindeki 0.100011 sayısı, onluk tabanda 0.546875 sayısına karşılık gelir, ve en son aralıkta (0.546 ile 0.56 aralığı) yer alan bu değer tam orta nokta olmasa bile etiket olarak kodlanabilir. Sadece noktadan sonraki basamakların (100011) alıcıya gönderilmesi yeterlidir. Etiket olarak son aralığın orta noktasının alındığı bir aritmetik kodlama sistemi için, $M(x_i)$ etiketinin matematiksel olarak gösterimi aşağıdaki gibi yapılabilir:

$$M(x_i) = \sum_{y<xi} P(y) + \frac{1}{2}P(x_i) \qquad (2.23)$$

Bu durumu günlük hayattan bir örnekle ifade etmeye çalışırsak; bir zarı iki defa attığımız zaman önce 1 sonra 2 gelmesi olasılığı için etiket oluşturmak istersek. Öncelikle her zarın gelme ihtimalini eşit kabul edelim, 0 ile 1 arasını 36 eşit parçaya bölebiliriz. Sonra ilk parçayı 1 ve 1 gelme olasılığına *($P(x_{11})$),* ikinci parçayı 1 ve 2 gelme olasılığına *($P(x_{12})$),…,* vererek bu 36 eşit parçayı paylaştırabiliriz. 1'den sonra 2 gelmesi olasılığı için etiket değerinin hesaplanması Denklem 2.24 ve 2.25'te görülmektedir [29].

$$M(x_{12}) = P(x_{11}) + \frac{1}{2}P(x_{12}) \tag{2.24}$$

$$M(x_{12}) = \frac{1}{36} + \frac{1}{2}\left(\frac{1}{36}\right) = \frac{3}{72} \tag{2.25}$$

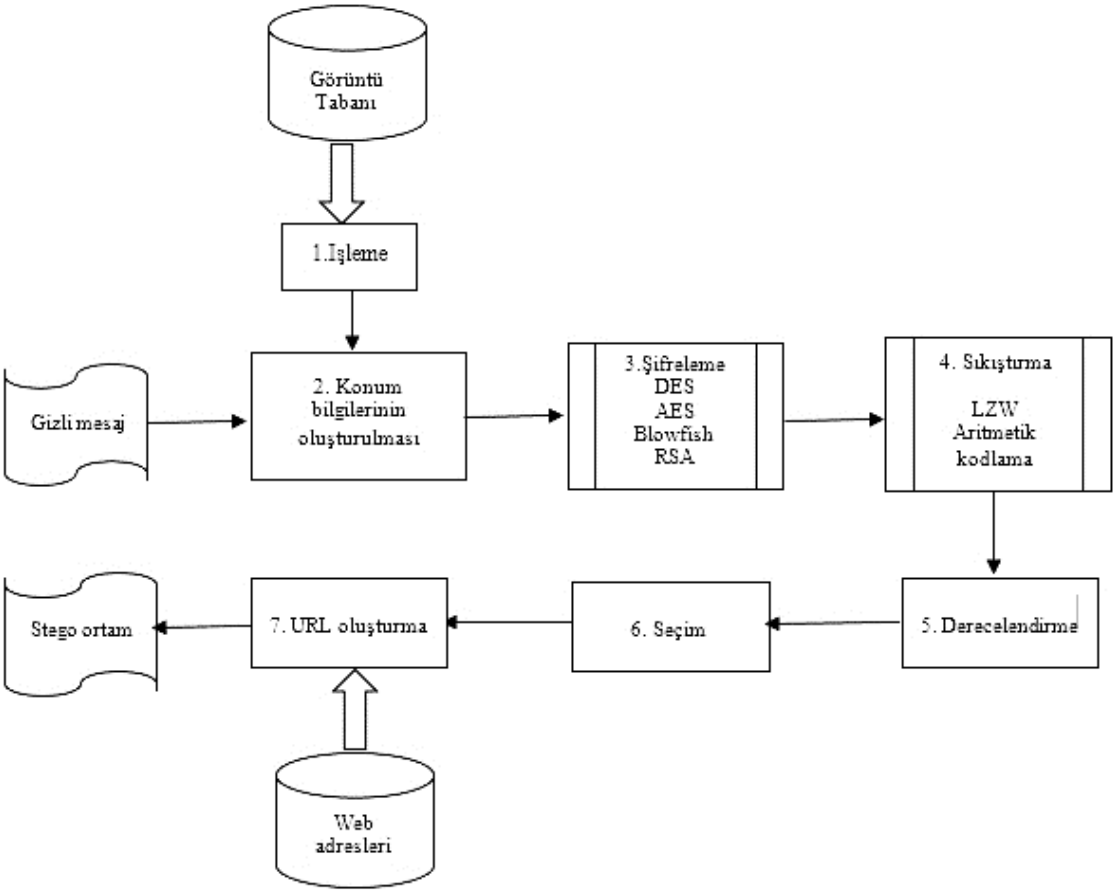
2.6. ÖNERİLEN YÖNTEM

Bu bölümde, önerilen metot, LZW sıkıştırma algoritması ve Aritmetik kodlama tekniklerinin her biri ile birlikte kullanılan AES, DES, Blowfish ve RSA şifreleme algoritmaları ile birlikte hem alıcı hem de gönderici tarafı detaylandırılarak açıklanmaktadır.

2.6.1. Gönderici Tarafı- Gömme Aşaması

Görüntü Tabanı: İçerisinde görüntülerin bulunduğu bir dizindir. Bu görüntüler rastgele seçilebilir, renkli ya da gri seviye olabilir. Bu görüntüler, taşıyıcı olarak kullanılacaktır. Önerilen metodun deneysel aşamasında, görüntü tabanında, literatürde sıkça rastlanan test görüntüleri kullanılmıştır. Görüntü tabanında bulunan görüntülerin sayısı makul bir düzeyde tutulmalıdır. Çok az sayıda görüntü, ard arda benzer örüntülerin saklanması durumunda sıklıkla aynı taşıyıcı görüntünün kullanılmasına sebep olacağından şüphe uyandıracaktır. Çok fazla sayıda görüntü kullanılması ise işlem yükü ve harcanan zamanı artıracaktır.

Web adresleri: Taşıyıcı görüntünün upload edileceği aday site adreslerinden oluşmaktadır.



Şekil 2.22. Gönderici tarafı - gömme aşaması.

1. ADIM: İşleme

Öncelikle, görüntü tabanındaki her görüntü gri seviyeye çevrilmektedir. Görüntü tabanındaki her bir görüntü *I* olsun. Bu durumda her bir görüntü:

$$I_{m\times n} = \begin{bmatrix} i_{1,1} & \cdots & i_{n,1} \\ \vdots & \ddots & \vdots \\ i_{1,m} & \cdots & i_{m,n} \end{bmatrix} \tag{2.26}$$

şeklinde temsil edilebilir. Bu adımda her bir görüntü, matrislerden dizilere dönüştürülmektedir. Bu durumda ise aşağıdaki gibi bir gösterim söz konusu olmaktadır:

$$l = m \times n \tag{2.27}$$

$$i_l = (i_1, i_2, \ldots, i_l) \tag{2.28}$$

2. ADIM: Konum bilgilerinin belirlenmesi

Bu adımda görüntü tabanındaki her bir görüntü ve gizli mesaj kullanılarak, gizli mesajın görüntü içindeki konu bilgilerini tutan *C* dizisi hesaplanmaktadır. *S*, *x* uzunluğunda bir gizli mesaj olsun:

$$S_x = (s_1, s_2, \dots, s_x) \tag{2.29}$$

S dizisindeki her bir karakterin ASCII kodu bulunarak:

$$ASCII(S)=A_x=(a_1,a_2,\dots,a_x) \tag{2.30}$$

dizisi elde edilir. Bu işlemden sonra *A* dizisinin her bir elemanı *i* dizisinde aranarak elemanlar arası göreli uzaklık değerleri hesaplanır:

$$\begin{aligned} a_1 = i_1 &\rightarrow a_1 - i_1 = 0 \\ a_1 = i_2 &\rightarrow a_1 - i_2 = 0 \\ a_1 = i_3 &\rightarrow a_1 - i_3 = 0 \\ &\vdots \\ a_1 = i_l &\rightarrow a_1 - i_l = 0 \end{aligned} \tag{2.31}$$

$a_1 = i_3$ olduğunu farz edelim. Bu durumda *C* dizisinin ilk elemanı, *i'* nin indeksi olan 3 değeridir. Daha sonra kalınan *A* ve *i* elemanlardan devam edilerek:

$$\begin{aligned} a_2 = i_4 &\rightarrow a_2 - i_4 = 0 \\ a_2 = i_5 &\rightarrow a_2 - i_5 = 0 \\ &\vdots \\ a_2 = i_n &\rightarrow a_2 - i_n = 0 \end{aligned} \tag{2.32}$$

$a_2 = i_4$ olduğunu farz edelim Bu durumda *C* dizisinin elemanı; *i*'nin şimdiki indeksi ve bir önceki indeksi arasındaki fark olacaktır: *4 - 2 = 2*. Bu işlem, *A* dizisinin elemanları bitinceye kadar görüntü tabanındaki her bir görüntü işlenerek elde edilen *i* dizisi için

gerçekleştirilmektedir. İşlemin sonucunda elimizde görüntü tabanındaki her bir görüntü adedince her birinin uzunluğu *x* kadar olan konum bilgileri dizileri vardır. Bu diziler ise:

$$C_{y\times x} = \begin{bmatrix} c_{1,1} & \cdots & c_{1,x} \\ \vdots & \ddots & \vdots \\ c_{y,1} & \cdots & c_{y,x} \end{bmatrix} \tag{2.33}$$

matrisini oluşturmaktadırlar.

3. ADIM: Şifreleme

Elde edilen *C* matrisinin her bir satırı ayrı ayrı, isteğe bağlı olarak seçilen simetrik (DES, AES, Blowfish) ve asimetrik (RSA) şifreleme tekniklerinden biri ile şifrelenmektedir. Böylece elimizde her bir satırındaki eleman sayısı, *C*'nin sütun değerinden farklı (muhtemelen daha fazla) yeni bir matris bulunmaktadır:

$$z \geq x \tag{2.34}$$

$$E_{y\times z} = \begin{bmatrix} e_{1,1} & \cdots & e_{1,z} \\ \vdots & \ddots & \vdots \\ e_{y,1} & \cdots & e_{y,z} \end{bmatrix} \tag{2.35}$$

4. ADIM: Sıkıştırma

Şifreleme sonucu elde edilen *E* matrisinin her bir satırı ayrı ayrı, isteğe bağlı olarak seçilen LZW ve Aritmetik kodlama algoritmalarından biri ile sıkıştırılmaktadır. Böylece elimizde her bir satırının uzunluğu farklı ve en fazla *E*' nin sütun değeri kadar olan yeni bir matris bulunmaktadır:

$$U_{y\times z} = \begin{bmatrix} u_{1,1} & \cdots & u_{1,z} \\ \vdots & \ddots & \vdots \\ u_{y,1} & \cdots & u_{y,z} \end{bmatrix} \tag{2.36}$$

U matrisinde boş kalan yerlere ise *"null"* değeri atanmaktadır.

5. ADIM: Derecelendirme

Elde edilen *U* matrisinin her bir satırındaki eleman sayısı hesaplanır:

$$F = \begin{bmatrix} f_1 \\ \vdots \\ f_y \end{bmatrix} \tag{2.37}$$

6. ADIM: Seçim

F içindeki minimum elemanın indeksi bulunur:

$$Index=Min(F) \tag{2.38}$$

Buradaki amaç en fazla sıkıştırmanın gerçekleştiği *U* matrisinin ilgili satırını tespit edip, görüntü tabanında buna karşılık gelen görüntünün bulunmasıdır. Böylece bu adım sonunda öncelikle kullanılacak taşıyıcı görüntü tespit edilecektir. Bununla birlikte elimizde:

$$U = (u_1, u_2, \ldots, u_v) \tag{2.39}$$

olan ve bir sonraki adımda seçilecek URL adresini modifiye etmede kullanılan *U* dizisi bulunacaktır.

7. ADIM: URL oluşturma

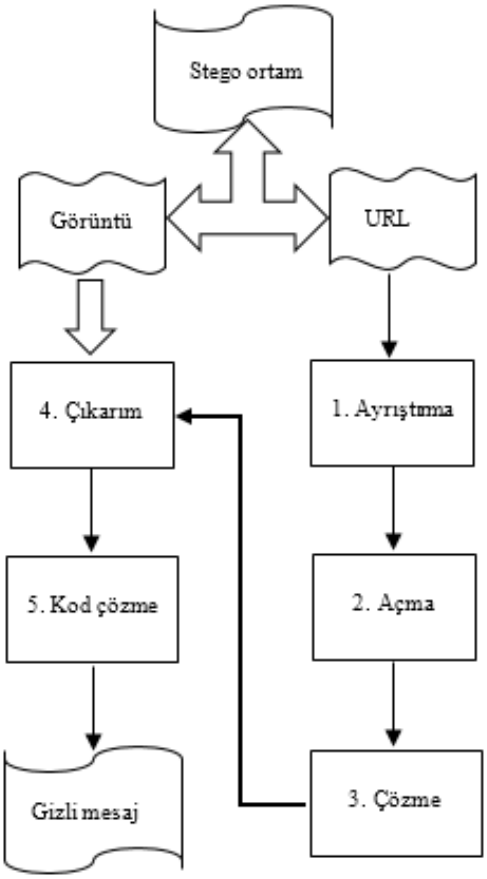
U matrisinde en az elemana sahip satır seçilerek, taşıyıcı görüntünün upload edileceği web sayfası belirlenmektedir. Taşıyıcı görüntü ile birlikte konum bilgilerinin de alıcıya iletilmesi gerekmektedir. Web adresleri içinden seçilen bir adres, şifrelenen ve sıkıştırılan konum bilgileri dizisinin eklenmesi yoluyla modifiye edilmektedir. Böylece elimizde taşıyıcı görüntü ve bu görüntünün yüklenmiş olduğu web adresi, yani stego ortam, bulunmaktadır.

Örnek olan bir URL aşağıda verilmektedir:

“http://s16.postimg.org/maz31aefp/00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuwnSaF8z+hrLw==/Baboon.png”

2.6.2. Alıcı Tarafı-Çıkarım Aşaması

Alıcıya iletilen stego ortam, *"www..."* şeklinde bir web adresidir. Gizli mesajın çıkarılabilmesi için alıcının bu URL içindeki *U* dizisine ve taşıyıcı görüntüye ihtiyacı vardır.



Şekil 2.23. Alıcı tarafı - çıkarım aşaması.

Bu bilgiler kullanılarak alıcı, gömme aşamasındaki süreçlerin tersini uygulayarak gizli mesajı çıkarabilmektedir. Şekil 2.23'te bu aşamalar görülmektedir.

1. ADIM: Ayrıştırma

Bu adımda, alınan URL içinden modifiye ile eklenen kısım çıkarılarak *U* dizisi elde edilmektedir.

2. ADIM: Açma

U dizisi, gömme aşamasında konum bilgisini tutan dizinin sıkıştırılmış ve şifrelenmiş

haliydi. Bu adımda *U* dizisi kullanılan sıkıştırma algoritması tekrar işletilerek çözülmektedir. Böylece, şifreli *E* dizisine ulaşılmaktadır.

$$E=Decompess(U) \tag{2.40}$$

3. ADIM: Çözme

Bu adımda, konum bilgisine ulaşmak için, şifrelenmiş *E* dizisi çözülmektedir. Böylece asıl *C* dizisine ulaşılmaktadır.

$$C=Decrypt(E) \tag{2.41}$$

4. ADIM: Çıkarım

Bu adımda, upload edilen görüntü (I dizisi) ve bu görüntü içine gizlenen karakterlerin konum bilgilerini tutan *C* dizisi kullanılarak, bu karakterlerin ASCII kodlarını tutan *A* dizisine ulaşılmaktadır.

$$i_1 = \text{index } (0+c_1) \tag{2.42}$$

$$i_2 = \text{index} \tag{2.43}$$

$$\begin{aligned} a_1 &= I[0+c_1] \\ a_2 &= I[c_1+c_2] \\ a_3 &= I[c_2+c_3] \\ &\vdots \\ a_n &= I[c_{n-1}+c_n] \end{aligned} \tag{2.44}$$

5. ADIM: Kod Çözme

Görüntü içine gizlenen karakterlerin ASCII kodlarını tutan *A* dizisinin kodu çözülmektedir. Böylece gizli mesajı tutan *S* dizisine ulaşılmaktadır.

$$S=TersASCII\ (A) \tag{2.45}$$

3. BULGULAR VE TARTIŞMA

Bu bölümde deneylerin sonuçları sunulacaktır. Deneyler C Sharp (C#) programlama dili ile yazılan bir yazılım kullanılarak yapılmıştır. Gizli mesajlar "*Lorem Ipsum…"* (http://www.tr.lipsum.com/feed/html) kullanarak üretilmiştir [32]. Çizelge 3.1' de görüldüğü üzere, *S* uzunlukları her defasında 10'ar karakter uzatılarak 10 karakterden 100 karaktere çıkartılmıştır. Böylelikle tarafsız bir değerlendirme amaçlanmaktadır. Çizelgelerde her deney için gözlemlediğimiz parametreler; *S* (gizli mesaj), *A* (*S* içindeki karakterlerin ASCII kodlarını), *C* (*S*'nin gizlenmesinden sonraki matris koordinatı), *C'* (şifrelenmiş *C*) ve son olarak *U* (seçilmiş URL'ye eklenen sıkıştırılmış *C*') olmaktadır.

3.1. KAPASİTE ANALİZİ

Çizelge 3.1' de önerilen metodun hem LZW hem de Aritmetik kodlama için DES, AES, Blowfish ve RSA şifreleme algoritmaları kullanılarak uygulanmasıyla elde edilen kapasite değerleri görülmektedir. Buradaki kapasite değerleri yüzde (%) cinsinden verilmektedir. Çizelgeden görüldüğü gibi, gizlenecek mesajdaki (*S*) karakter sayısı arttıkça, kapasite değerleri de artmaktadır. En yüksek kapasite değerleri, 100 karakterlik gizli mesaj için elde edilmiştir. Buna göre 100 karakter saklandığında elde edilen en yüksek kapasite değeri % 34.60 ile RSA-LZW sıkıştırma kombinasyonundan, en düşük kapasite değeri ise %12.45 ile Blowfish-Aritmetik kodlama kombinasyonundan elde edilmiştir. Sonuçlar incelendiğinde LZW sıkıştırma algoritmasının, aritmetik kodlamaya göre daha iyi performans verdiği gözlemlenmiştir. DES, AES, Blowfish ve RSA şifreleme algoritmaları için kullanılan anahtar uzunlukları 128 bittir. Önerilen yöntem için yapılan örnek uygulamalar EKLER'de çizelgeler halinde (**Bkz.** 65-86) gösterilmektedir.

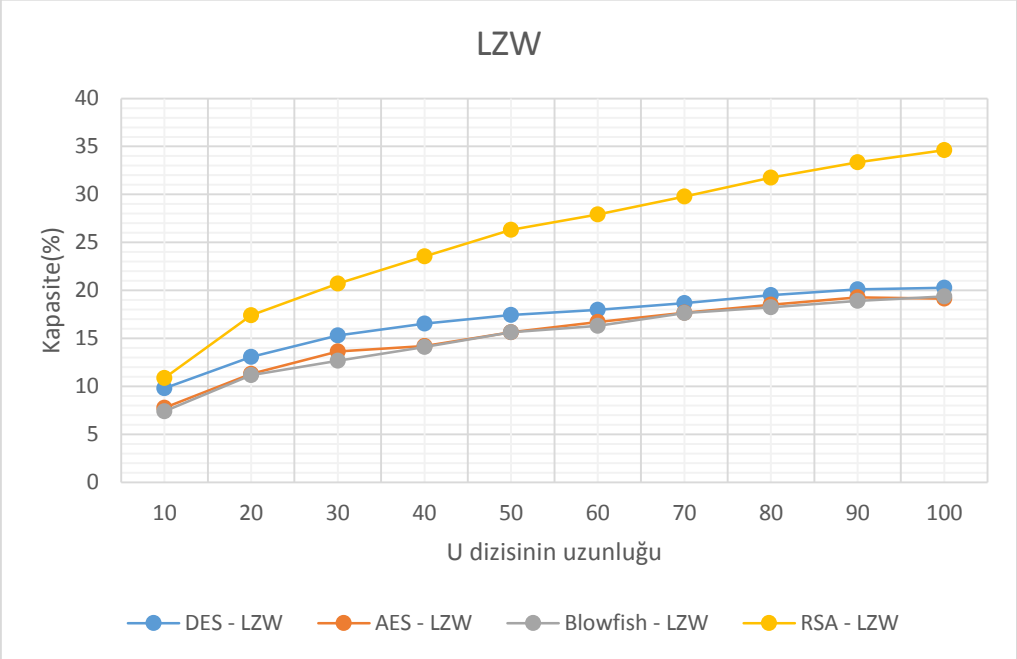
Kapasite, gizli mesajın boyutunun, stego ortamın boyutuna oranı olarak ifade edilmektedir [4]:

$$C = \frac{Gizli\ mesajın\ boyutu}{Stego\ ortamın\ boyutu} \tag{3.1}$$

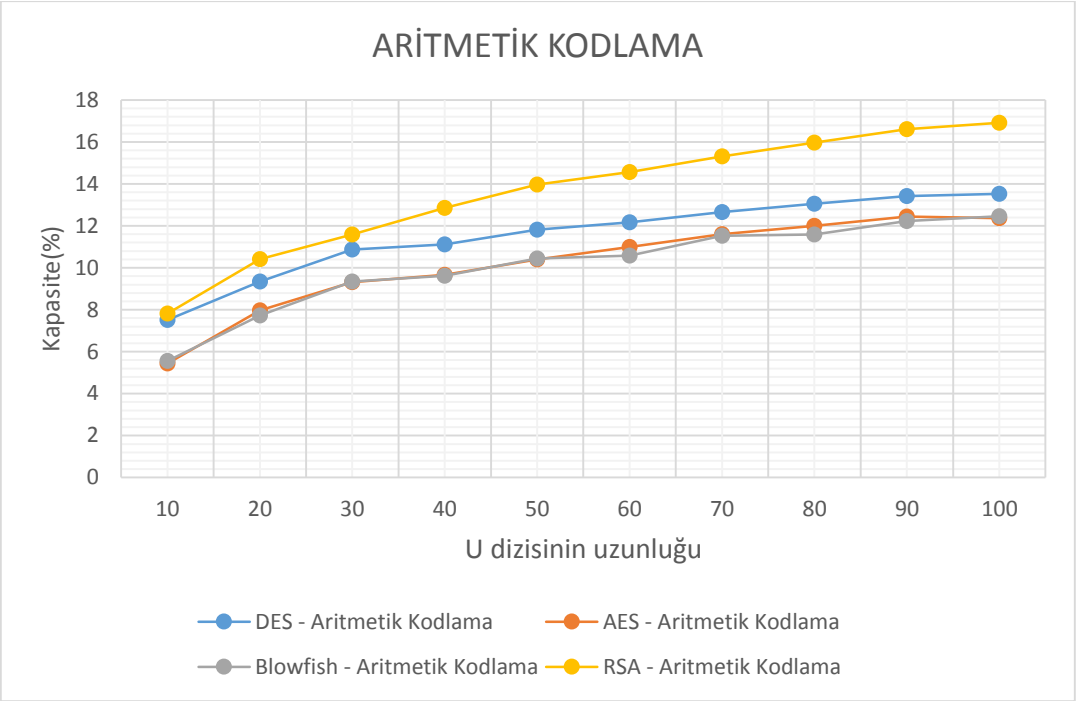
Çizelge 3.1. Kullanılan sıkıştırma ve şifreleme algoritmalarına göre kapasite sonuçları(%).

	LZW				ARİTMETİK KODLAMA			
Karakter Sayısı	**DES**	**AES**	**Blowfish**	**RSA**	**DES**	**AES**	**Blowfish**	**RSA**
10	9,803922	7,751938	7,407407	10,86957	7,518797	5,434783	5,555556	7,8125
20	13,0719	11,29944	11,17318	17,3913	9,345794	7,968127	7,722008	10,41667
30	15,30612	13,63636	12,65823	20,68966	10,86957	9,31677	9,345794	11,58301
40	16,52893	14,1844	14,08451	23,52941	11,11111	9,661836	9,615385	12,86174
50	17,4216	15,625	15,625	26,31579	11,82033	10,39501	10,43841	13,96648
60	17,96407	16,71309	16,30435	27,90698	12,17039	10,98901	10,58201	14,56311
70	18,66667	17,67677	17,63224	29,78723	12,65823	11,60862	11,53213	15,31729
80	19,5122	18,51852	18,22323	31,74603	13,05057	11,994	11,5942	15,96806
90	20,08929	19,27195	18,90756	33,33333	13,41282	12,44813	12,22826	16,60517
100	20,28398	19,12046	19,37984	34,60208	13,5318	12,36094	12,4533	16,92047

Çizelge 3.1 görüldüğü üzere *U*, sıkıştırma ile azaltılmıştır. Böylelikle, web sayfa URL'sinin sıradan ve masum görünmesi için *U*'nun uzunluğu mümkün oldukça küçük tutulmuş olur. Şekil 3.1 ve Şekil 3.2'de; Çizelge 3.1' den elde edilen sonuçların grafiksel gösterimi yer almaktadır.



Şekil 3.1. LZW sıkıştırma algoritmasına göre DES, AES, Blowfish, RSA için kapasite grafiği.



Şekil 3.2. Aritmetik kodlama algoritmasına göre DES, AES, Blowfish, RSA için kapasite grafiği.

3.2. GÜVENLİK VE ALGILANAMAZLIK ANALİZİ

Önerilen metot ile gizlenecek bilgi iki ortamda temsil edilmektedir. Bunlardan birisi taşıyıcı görüntü; gizlenecek bilginin koordinatlarına sahip bir nevi harita, diğeri ise gizlenecek bilginin bu haritadaki koordinatlarını tutan dizidir. Bu dizi, kapasite verimi için önce sıkıştırılmakta (LZW, Aritmetik kodlama), güvenlik ve sağlamlık içinde şifrelenmektedir (DES, AES, Blowfish, RSA). En yüksek kapasite değeri veren RSA sağlamlık ve güvenilirlik açısından en iyi sonuçları vermektedir. Ayrıca, kullanılan LZW ve Aritmetik kodlama algoritmaları da, metodun çıkarım aşamasını karmaşıklaştırarak, çözülmeyi zorlaştırmaktadır.

Önerilen metot kullanılarak, verilen gizli mesaj ve bu mesaj için elde edilen URL aşağıda, ekran çıktısı ise Şekil 3.3'te gösterilmektedir. İlgili URL' nin tarayıcıya yazılarak çalıştırılması durumunda web sayfasına ulaşmada herhangi bir sorunla ya da hata mesajı ile karşılaşılmamaktadır.

S=(L,o,r,e,m, ,i,p,s,u)

URL:

http://s16.postimg.org/maz31aefp/00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuwnSaF8z+hrLw==/Baboon.png

Şekil 3.3. Önerilen metot ile oluşturulan stego ortam

Bu web sayfasındaki görüntü, bahsedildiği gibi, URL'ye eklenen kısımdaki koordinat bilgileri çıkarıldıktan sonra, gizli mesajın bulunması için kullanılacaktır. Gizlenecek bilgiler, ilgili görüntüde herhangi bir değişiklik gerçekleştirilmeden yalnızca kamufle edilerek saklanmıştır. Dolayısıyla, görüntü kalitesinde herhangi bir değişiklik, bozulma vb. durum oluşmamaktadır. Bu durumu matematiksel olarak kanıtlamak için tepe sinyal gürültü oranı (PSNR) ölçümü kullanılmıştır. Buna göre orijinal görüntü X ve gömme aşamasından sonra elde edilen görüntü X' olsun. PSNR için öncelikle karesel hata (MSE) hesaplaması gerekmektedir:

$$MSE = \frac{1}{MxN}\sum_{i=1}^{M}\sum_{j=1}^{N}(X'(i,j) - X(i,j))^2 \quad (3.2)$$

Denklem 3.1.'e göre *db* birimi ile ölçülen PSNR değeri:

$$PSNR = 10 \log\left(\frac{255^2}{MSE}\right) \tag{3.3}$$

şeklinde hesaplanmaktadır [1,6].

Önerilen metotta *X' = X* olduğundan MSE *"0"* çıkmaktadır:

$$MSE = \frac{1}{MxN}\sum_{i=1}^{M}\sum_{j=1}^{N}(X(i,j) - X(i,j))^2 = 0 \tag{3.4}$$

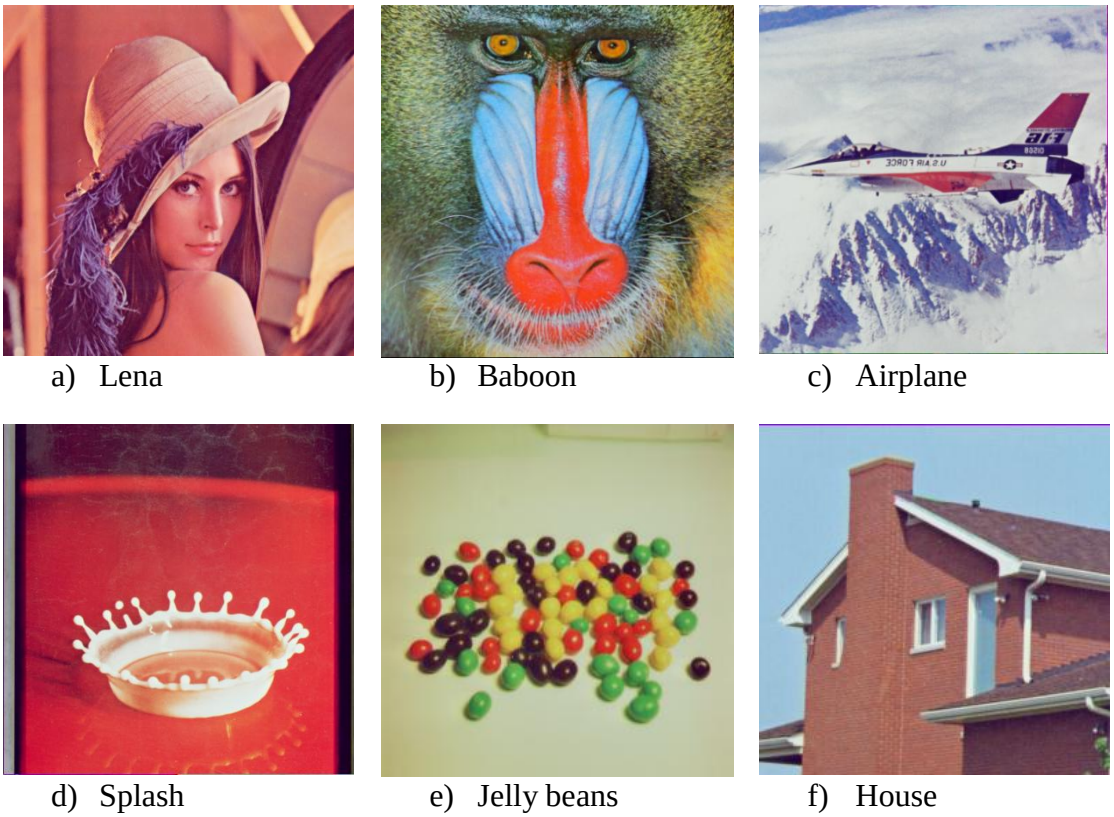
PSNR değeri ne kadar büyük çıkarsa, sonuç o kadar iyi olmaktadır [1,6].

Buna göre:

$$PSNR = 10 \log\left(\frac{255^2}{0}\right) = \infty \tag{3.4}$$

PSNR = ∞ olmaktadır.

Kullanılan tüm görüntüler Şekil 3.4-a, b, c, d, e ve f'de gösterilmiştir. Önerilen metoda göre en kısa *U* dizisini elde etmek için seçilen görüntü Baboon görüntüsü olmaktadır.



a) Lena b) Baboon c) Airplane

d) Splash e) Jelly beans f) House

Şekil 3.4. Seçilen görüntüler a) Lena b) Baboon c) Airplane d) Splash e) Jelly beans f) House

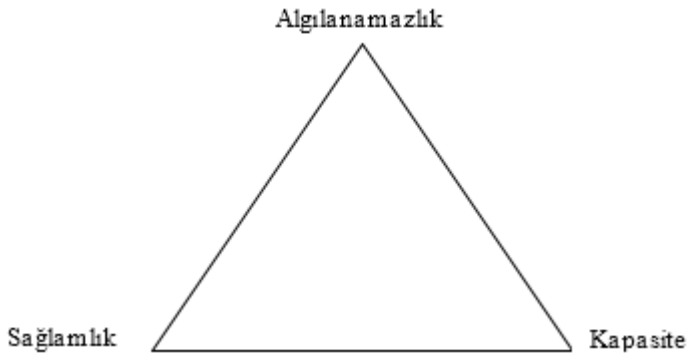
Bu web adresi Facebook, Skype ve benzeri iletişim kanalları vasıtasıyla alıcıya gönderilecek parçadır. Alıcı URL'yi aldığında, önerilen metodun çıkarım prosedürünü uygulayarak gizli mesajı elde edebilmektedir.

4. SONUÇLAR VE ÖNERİLER

Bilgi teknolojisinin hızlı büyümesi ile günümüzde, insanlar kolayca internet veya mobil kanallar üzerinden kendi bilgisayar veya cep telefonları ile multimedya içeriklerini alabilirler. Böylelikle veri gizleme, gizli mesajları iletmek için etkili tekniklerden biri haline gelmiştir. Görüntüler, videolar, metinler vb. gibi veri gizleme ortamları dijital medya mesajlarını gizleyebilmektedir. Steganografi, amaçlanan alıcı dışında kimsenin gizli verinin varlığı hakkında bir şey bilmediği şekilde veriyi gizleme tekniğidir. Bir şifreleme algoritması mesajı anlamsız ve şüpheli hale getirdiğinden steganografi, şifreleme algoritmasından farklıdır. Steganografi, amacı bakımından filigranlamadan da farklıdır. Filigranlama, esasen ortamın kimliklendirilmesiyle uğraşırken steganografi, veriyi gizleme ile uğraşmaktadır. Filigranlama tekniğinde gömülü veri, taşıyıcının herhangi bir niteliği ile alakalı olabilmektedir ve taşıyıcıya ilişkin ekstra bilgi ya da özellik iletmektedir. Steganografide genellikle gömülü bilginin, bilgiyi geçirmek amacıyla bir mekanizma olarak basitçe kullanılan taşıyıcıyla gerçekleştireceği herhangi bir etkileşim yoktur.

Steganografi teknikleri, taşıyıcı olarak görüntü, ses, video ve metin ortamlarını kullanarak bilgiyi gizlemektedirler. Bu tez çalışmasında önerilen metot ile gizli bilgi üç temel aşamada saklanmıştır. Birinci aşamada taşıyıcı olarak kullanılan görüntünün içeriği değiştirilmeden, yalnızca ilgili karakterlerin konum bilgileri tespit edilmektedir. İkinci aşamada, bu konum bilgileri şifrelenip sıkıştırılmaktadır. Üçüncü aşamada ise, elde edilen bu içerik, görüntünün upload edileceği web sayfasının URL bilgisine eklenmektedir. Bu ekleme sonucunda URL bilgisinin tarayıcıya yazılıp sınanması sonucu herhangi bir hata durumu veya farklı bir web sayfasına geçiş gerçekleşmemektedir. Dolayısıyla önerilen metot ile alıcıya yalnızca bu web adresi gönderilmektedir. Gizli bilgi, iki farklı ortamda saklandığından URL ve görüntüye ayrı ayrı gerçekleşecek saldırılar bir anlam ifade etmeyecektir. Bununla birlikte önerilen metotta görüntü tabanının alıcı tarafında olmasına gerek duyulmamaktadır. Önerilen metotta herkes kendi görüntü tabanını oluşturabilme özgürlüğüne de sahiptir.

Şekil 4.1'de bir veri gizleme algoritmasındaki üç ana gereksinimin birbiriyle olan ilişkilerini temsil eden sihirli üçgen görülmektedir.



Şekil 4.1. Veri gizleme sistemindeki sihirli üçgen [33].

Bu üç gereksinim arasından bir ödünleşim mevcuttur [33]. Herhangi birine yaklaşılması durumunda diğer ikisinden uzaklaşılmaktadır. Buna göre önerilen metot ile kapasite ve algılanamazlık arasındaki denge kurularak, kullanılan şifreleme algoritmaları ile de sağlamlık ve güvenlik desteklenmektedir. Bu noktada, saklanacak bilgi miktarı bakımından belli bir oranın üzerine çıkılması sağlamlık bakımından sorun teşkil etmemekte, kapasite bakımından da makul değerleri vermekte ancak algılanamazlık bakımından sorun teşkil etmektedir. Başka bir deyişle, gizlenecek bilgi miktarı arttıkça, karşı tarafa gönderilecek URL uzunluğu artmaktadır. Çok uzun olan URL adresleri, şüphe uyandırıcı olabilmektedir. Bunun önüne geçilmesi amacıyla ilgili içeriğin parçalanarak birden fazla web adresine eklenmesi şeklinde bir çözüme gidilebilir. Önerilen metodun başka bir dezavantajı ise; her web adresinin böyle bir eklentiyi desteklememesi ve oluşturulan URL'nin tarayıcıya yazılarak çalıştırılması sonucunda hata ile karşılaşılmasıdır. Bu durum, önerilen metotla uyumlu çalışabilecek web sayfalarının tespitini gerektirmektedir.

5. KAYNAKLAR

[1] Huang H., Fang W., Techniques and applications of intelligent multimedia data hiding, *Telecommunication Systems,* 44 (3-4) (**2010**) 241-251.

[2] Mir N., Hussain S.A., Secure web-based communication, Procedia Computer Science, 3 (2011) 556–562.

[3] Weng C., Tso H.,, Wang S., Steganographic data hiding in image processing using predictive differencing, *Opto−Electronics,* 20 (4) (**2012**) 379–379.

[4] ŞATIR E., Bilgi güvenliği için metin steganografisinde yeni bir yaklaşım, *Doktora Tezi,* Selçuk Üniversitesi, (**2013**).

[5] Sajedi H., Jamzad M., Using contourlet transform and cover selection for secure steganography, *International Journal of Information Security,* 9 (5) (**2010**) 337-352.

[6] Cheddad A., Condell J., Curran K., McKevitt P., Digital image steganography: Survey and analysis of current methods, *Signal Processing,* 90 (**2010**) 727–752.

[7] Lee H., Lee C., Chen L., A perfect maze based steganographic method, *Journal of Systems and Software,* 83 (12) (**2010**) 2528–2535.

[8] Yang W., Chen L., A steganographic method via various animations in PowerPoint files, *Multimedia Tools and Applications,* (**2010**).

[9] Castiglione A., De Santis A., Fiore U., Palmieri F., An asynchronous covert channel using spam, *Computers & Mathematics with Applications,* 63 (2) (**2012**) 437–447.

[10] Wang, Z.H., Kieu, T.D., Chang, C.C., Li, M.C., Emoticon-based text steganography in chat. *Proceedings of 2009 Asia-Pacific Conference on Computational Intelligence and Industrial Applications,* 2 (**2009**) 457–460.

[11] Fu, Y., Zhang, R., Ma, S., Qu, Z., Nıu, X., Yang, Y., Fast coding in digital steganography, *The Journal of China Universities of Posts and Telecommunications*, 16 (6) (**2009**) 92-96.

[12]Anonim,http://www.cs.bham.ac.uk/~mdr/teaching/modules03/security/students/SS5/Steganography.htm (Erişim Tarihi: 27 Aralık **2013**).

[13] Aksuoğlu A., RSA Algoritmasının İyileştirilmesi İçin Yeni Bir Yaklaşım, *Yüksek Lisans Tezi*, Anadolu Üniversitesi, (**2012**).

[14] Egemen T., Design and system implementation of a crypto processor for AES and DES algorithms, *Master's Thesis*, Middle East Technical University, (**2007**).

[15] Günden Ü., Şifreleme Algoritmalarının Performans Analizi, *Yüksek Lisans Tezi,* Sakarya Üniversitesi, (**2010**).

[16] Ciğer İ., Data Şifreleme Algoritmaları ve Performans Analizi, *Yüksek Lisans Tezi*, İstanbul Üniversitesi, (**2012**).

[17] Anonim, Data encryption standard (DES), *Federal Information Processing Standards Publication*, Sayı: 46-3, (**1999**).

[18] Öztemür, M., AES algoritmasının bir gerçeklemesine güç analizi saldırıları, *Yüksek Lisans Tezi* Yıldız Teknik Üniversitesi, (**2012**).

[19] Başkök M., D., AES şifreleme algoritmasının modellenmesi, *Yüksek Lisans Tezi*, Gazi Üniversitesi, (**2007**).

[20] Doğan A., Y., AES algoritmasının FPGA üzerinde düşük güçlü tasarımı, *Yüksek Lisans Tezi*, İstanbul Teknik Üniversitesi, (**2008**).

[21] Kula G., Ç., Gelişmiş şifreleme standardı blok şifreleme algoritmasının bir mikroişlemci üzerinde gerçeklemesine yan kanal saldırısı, *Yüksek Lisans Tezi*, İstanbul Teknik Üniversitesi, (**2009**).

[22] Anonim, Announcing the advanced encryption standard (AES), *Federal Information Processing Standards Publication*, Sayı: 197, (**2001**).

[23] Anonim, http://en.wikipedia.org/wiki/Feistel_network (Erişim Tarihi: 25 Aralık **2013**).

[24] Anonim, http://tr.opensuse.org/Blowfish#Hakk.C4.B1nda (Erişim Tarihi: 27 Aralık **2013**).

[25] Anonim , http://www.design-reuse.com/articles/5922/encrypting-data-with-the-blowfish-algorithm.html (Erişim Tarihi: 27 Aralık **2013**).

[26] Kodaz H., Botsalı M., F., Simetrik ve asimetrik şifreleme algoritmalarının karşılaştırılması, *Teknik-Online Dergi,* 9 (1) (**2010**).

[27] Okumuş İ., RSA kripto sisteminin hızını etkileyen faktörler, *Doktora Tezi*, Atatürk Üniversitesi, (**2012**).

[28] Anonim, http://en.wikipedia.org/wiki/RSA_(algorithm) (Erişim Tarihi: 28 Aralık **2013**).

[29] Mesut A., Veri sıkıştırmada yeni yöntemler, *Doktora Tezi*, Trakya Üniversitesi, (**2006**).

[30] Anonim, http://en.wikipedia.org/wiki/Arithmetic_coding (Erişim Tarihi: 28 Aralık **2013**).

[31] Akıncı A., Universal command generator for robotics and CNC Machinery, *Master's Thesis*, Middle East Technical University, (**2009**).

[32] Anonim, http://www.tr.lipsum.com/feed/html (Erişim Tarihi: 20 Aralık **2013**).

[33] Zaker, N., Hamzeh, A., A novel steganalysis for TPVD steganographic method based on differences of pixel difference histogram, *Multimed Tool Applications*, 58 (1) (**2012)** 147-166.

6. EKLER

EK-1. DES ŞİFRELEME – LZW SIKIŞTIRMA

	S	***A***	***C***	***DES***	***x***	***LZW***	***x'***
10	{L,o,r,e,m, ,i,p,s,u,}	76,111,114,101,109,32,105,112,115,117,	110,105,18,342,138,1358,19,91,391,246	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuwnSaF8z+hrLw==	56	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuwnSaF8z+hrLw==	56
20	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDXhLUgGrPTFs=	108	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxhVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDXhLUgGrPTFs=	107
30	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUemYmC60mhj4nw==	152	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxhVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDİbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUemYmC60mhj4nw==	150
40	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnDqxtz6it6icsTC/v20ML+mB7lGBPU+5fLMV5WmAz+1u5QKRmgPUVKCb4=	204	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxhVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDİbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnŅxtz6ithucsTC/v20ML+mB7lGBPU+5ṪMV5WmAz+1u5QKfgl̨VKCĹ=	196
50	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnDqxtz6it6icsTC/v20ML+mB7lGBPU+5ṪMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTloxYe/YqS3NlBJuMmLwjN48c1s2IZvyjKKg==	248	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxhVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDİbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnŅxtz6ithucsTC/v20ML+mB7lGBPU+5ṪMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTŋxYe/ĶS3NlBJuMmLwjN48c1s2IZvyjKKg==	241

60	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnDqxtz6it6icsTC/v20ML+mB7lGBPU+5fLMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbxYURSH/0bc9RSM/6J49qmnIkMcM934jCTFa+DW58CcjmG9/4V/ERZny4=	300	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxhVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDİbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnN̈xtz6ithucsTC/v20ML+mB7lGBPU+5ṪDMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTŋxYe/R̈S3NlBJuMmLwjN48dgVQZehfbİURSH/0ŠŸSM/6ā9qmnIkMcM934jCTFa+DW58Ccjmd/4V/ERZny4=	288
70	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,97,98,105,116,46,32,67,117,114,97,98,105,116,117,114,97,98,105,116,117,114,,114,32,101,103,101,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnDqxtz6it6icsTC/v20ML+mB7lGBPU+5fLMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbxYURSH/0bc9RSM/6J49qmnIkMcM934jCTFa+DW58CcjmGwPCN6LllxjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbxYURSH/0bc9RSM/6J49qmnIkMcM934jCTFa+DW58CcjmGwPCN6LllxjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbxYURSH/0bc9RSM/6J49qmnIkMcM934jCTFa+DW58CcjmGwPCN6LllxjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbİURSH/0ŠŸSM/6ā9qmnIkMcM934jCTFa+DW58CcjmGwPCN6LllxjoNJ53k6aePsGMnnolhuaX8ZoAQUŖJ8f4tGxmzn7uocdQ==	344	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxhVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDİbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnN̈xtz6ithucsTC/v20ML+mB7lGBPU+5ṪDMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTŋxYe/R̈S3NlBJuMmLwjN48dgVQZehfbİURSH/0ŠŸSM/6ā9qmnIkMcM934jCTFa+DW58CcjmGwPCN6LllxNJ53k6aePsGMnnolhuaX8ZoAQUŖJ8f4tGxmzn7uocdQ==	329
80	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,97,98,105,116,46,32,67,117,114,97,98,105,116,117,114,,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnDqxtz6it6icsTC/v20ML+mB7lGBPU+5fLMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbxYURSH/0bc9RSM/6J49qmnIkMcM934jCTFa+DW58CcjmGwPCN6LllxjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbİURSH/0ŠŸSM/6ā9qmnIkMcM934jCTFa+DW58CcjmGwPCN6LllxNJ53k6aePsGMnnolhuaX8ZoAQUŖJ8f4uĆMwDRO6HHĈĐĵ+YB+fpwR2igÖkuRvm8OyI+ufSF7rPhkWg5U	384	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxhVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDİbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnN̈xtz6ithucsTC/v20ML+mB7lGBPU+5ṪDMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTŋxYe/R̈S3NlBJuMmLwjN48dgVQZehfbİURSH/0ŠŸSM/6ā9qmnIkMcM934jCTFa+DW58CcjmGwPCN6LllxNJ53k6aePsGMnnolhuaX8ZoAQUŖJ8f4uĆMwDRO6HHĈĐĵ+YB+fpwR2igÖkuRvm8OyI+ufSF7rPhkWg5U	364
90	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,p,i,t, ,a,r,c,u,., ,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,97,98,105,116,46,32,67,117,114,97,98,105,116,117,114,,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,112,105,116,32,97,114,99,117,46,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnDqxtz6it6icsTC/v20ML+mB7lGBPU+5fLMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbxYURSH/0bc9RSM/6J49qmnIkMcM934jCTFa+DW58CcjmGwPCN6LllxjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbİURSH/0ŠŸSM/6ā9qmnIkMcM934jCTFa+DW58CcjmGwPCN6LllxNJ53k6aePsGMnnolhuaX8ZoAQUŖJ8f4uĆMwDRO6HHĈĐĵ+YB+fpwR2igÖkuRvm8OyI+ufSsJEcJ2cQJIŖaiFqş+LtUkCeu4ĶABhuzajŋVUoJhEKWEgqT/k=	428	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxhVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDİbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnN̈xtz6ithucsTC/v20ML+mB7lGBPU+5ṪDMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTŋxYe/R̈S3NlBJuMmLwjN48dgVQZehfbİURSH/0ŠŸSM/6ā9qmnIkMcM934jCTFa+DW58CcjmGwPCN6LllxNJ53k6aePsGMnnolhuaX8ZoAQUŖJ8f4uĆMwDRO6HHĈĐĵ+YB+fpwR2igÖkuRvm8OyI+ufSsJEcJ2cQJIŖaiFqş+LtUkCeu4ĶABhuzajŋVUoJhEKWEgqT/k=	402

100	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,p,i,t, ,a,r,c,u,., ,C,u,m, ,s,o,c,i,i,s,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,112,105,116,97,115,32,115,117,115,99,105,112,105,116,32,97,114,99,117,46,32,67,117,109,32,115,111,99,105,105,115,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281,500,284,87,13,75,85,68,86,55,306,33	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnDqxtz6it6icsTC/v20ML+mB7lGBPU+5fLMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbXURSH/0bc9RSM/6J49qmnIkMcM934jCTFa+DW58CcjmGwPCN6Lllxjo53k6aePsgVMnnol6iaX8ZoAQU7lJ8f4u3TMwDRO6HHikNGiOj+YB+fpwR2igDWkuRvm8OyI+ufSsJEcJ2cQQUI7laiFq6L+LtUkCeu4mEAB6izajU7VUoBOzvYuLLYo5InN82c3XoPAAXcc2KzxlkROKxMm1aAHbH821TVbudaz5kal2wqSkh	480	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxhVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDİbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnṆxtz6ithucsTC/v20ML+mB7lGBPU+5ʹDMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTŋxYe/ṚS3NlBJuMmLwjN48dgVQZehfbİURSH/0ŠŸSM/6ã9qmnIkMcM934jCTFa+DW58CcjmGwPCN6Lllx NJ53k6aePsGMnnolhuaX8ZoAQUṚJ8f4uĆMwDRO6HHĈĐƀj+YB+fpwR2igÖkuRvm8OyI+ufSsJEcJ2cQḭIṚaiFqș+LtUkCeu4Ķ ABhuzajŋVUoBOzvđLđo5InN8fj3XoPAAXcc2KzxlkȥKxŮ1aAκHȝ1TVbudaz5kal2wqSkh	447

EK-2. AES ŞİFRELEME – LZW SIKIŞTIRMA

	S	*A*	*C*	*AES*	*x*	*LZW*	*x'*
10	{L,o,r,e,m, ,i,p,s,u,}	76,111,114,101,109,32,105,112,115,117,	110,105,18,342,138,1358,19,91,391,246	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD847377713479A9E4E67EB7009274D784E0	96	6319E06677Ą53ċ975E6CBC61EF496F8AF5CFDFđ'BĐ202ë2B3ĞABF3BDD8473Ĉ713Ĺ9Aă4đĉB7009274D7κĄ	83
20	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAA67CF4718EC002F080F051FDC656E0DD	160	6319E06677Ą53ċ975E6CBC61EF496F8AF5CFDFđ'BĐ202ë2B3ĞABF3BDD84ĘFB9Ĕ44ę0C8BĒĴąŃ93Čę625101ļBA4ĩC1Fĥ8Ĺ73DĿDE83E1ACAAćĢ4718EC0ĪF080ŷŐġě56ĄĶ	131
30	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D540611607DA67B4A3573A22CCDA2D2868AF6F11	224	6319E06677Ą53ċ975E6CBC61EF496F8AF5CFDFđ'BĐ202ë2B3ĞABF3BDD84ĘFB9Ĕ44ę0C8BĒĴąŃ93Čę625101ļBA4ĩC1Fĥ8Ĺ73DĿDE83E1ACı858D7BţœAŒDĮ4ŴB17Ĩŝŋĕ8ćŝš7ĀDĜFŪĒ4D2ķ0Ů54ą11607DAćűAČşA22CCĿĦ2'BĞFĜhu	174
40	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,117,114,32,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D5406116552 3FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81CCF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FFE1BFD6152DB3B0657E3BFC644CF6211D	320	6319E06677Ą53ċ975E6CBC61EF496F8AF5CFDFđ'BĐ202ë2B3ĞABF3BDD84ĘFB9Ĕ44ę0C8BĒĴąŃ93Čę625101ļBA4ĩC1Fĥ8Ĺ73DĿDE83E1ACı858D7BţœAŒDĮ4ŴB17Ĩŝŋĕ8ćŝš7ĀDĜFŪĒ4D2ķ0Ů54ą1165523FŤ82ď'F0EE4ĚEBŋ50ķěŗ432ŴĤŠ3ÉĘA81CĢĉDāFÉB7A2lČD30Ш㸵1'CăĆŃ5Ļ.2ijĘκ8śtIJCĺŕÝŇŧĉĴĐ6ŁĢÕhuD	236
50	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,117,114,32,97,100,105,112,105,115,99,105,110,103,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D5406116552 3FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81CCF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF227637A1EB08514946293740DD1A370D93304FD1509E26136FDF877DE7015D2AE428D7C54830C8875AFCF716F7A5507A	384	6319E06677Ą53ċ975E6CBC61EF496F8AF5CFDFđ'BĐ202ë2B3ĞABF3BDD84ĘFB9Ĕ44ę0C8BĒĴąŃ93Čę625101ļBA4ĩC1Fĥ8Ĺ73DĿDE83E1ACı858D7BţœAŒDĮ4ŴB17Ĩŝŋĕ8ćŝš7ĀDĜFŪĒ4D2ķ0Ů54ą1165523FŤ82ď'F0EE4ĚEBŋ50ķěŗ432ŴĤŠ3ÉĘA81CĢĉDāFÉB7A2lČD30Ш㸵1'CăĆŃ5Ļ.2ijĘκ8ś2ΘĀ!ėBƐŐĺ'ŎĐ7ĠĶŨ3żsŋ0ŀD1ļă2ĖŹġĝČţżDZǫAP2Ů7Cƒťl88đ'ğ5lĥAK0!	274
60	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D5406116552 3FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81CCF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF227637A1EB08514946293740DD1A370D93304FD1509E26136FDF877DE7015D2A6B68F0F34E84DE3CD83AE158EE0099A317EA8162EB498077EDDBCFCE1040F623CD4EA804C762F39A1E58CA3D61519285	448	6319E06677Ą53ċ975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84ĘFB9Ĕ44ę0C8BĒĴąŃ93Čę625101ļBA4ĩC1Fĥ8Ĺ73DĿDE83E1ACı858D7BţœAŒDĮ4ŴB17Ĩŝŋĕ8ćŝš7ĀDĜFŪĒ4D2ķ0Ů54ą1165523FŤ82ď'F0EE4ĚEBŋ50ķěŗ432ŴĤŠ3ÉĘA81CĢĉDāFÉB7A2lČD30Ш㸵1'CăĆŃ5Ļ.2ijĘκ8ś2ΘĀ!ėBƐŐĺ'ŎĐ7ĠĶŨ3żsŋ0ŀD1ļă2ĖŹġĝČţżDZǫA6ŅŮNJ3Ĺκţ3Cķ3Atũoŋ0099Aāĉ₹l22Ė80ĈEĶĚĐtŽNJÕêšE₹ŽCbŮ3ŘėŭŪŠĖÓ92Ū	313

70	{L,o,r,e,m, ,i,p, s,u,m, ,d,o,l,o, r, ,s,i,t, ,a,m,e, t,,, ,c,o,n,s,e,c, t,e,t,u,r, ,a,d,i, p,i,s,c,i,n,g, ,e, l,i,t,., ,C,u,r,a,b ,i,t,u,r, ,e,g,e,}	76,111,114,101,109,32, 105,112,115,117,109,3 2,100,111,108,111,114, 32,115,105,116,32,97,1 09,101,116,44,32,99,11 1,110,115,101,99,116,1 01,116,117,114,32,97,1 00,105,112,105,115,99, 105,110,103,32,101,10 8,105,116,46,32,67,117 ,114,97,98,105,116,117 ,114,32,101,103,101,	110,105,18,342,138,13 58,19,91,391,246,383,3 98,89,60,23,9,533,2858 ,181,61,243,13043,449, 155,273,40,130,1011,7, 22,40,4,284,74,532,119 ,13,94,216,1150,17,72, 1,6,234,226,388,135,27 1,168,6699,101,40,230, 289,241,645,32,129,23 9,28,58,402,92,90,76,4 07,264,18,77	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E2029 62B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F46251 01B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01D B34A0B17028933C68678989763D6FFCA6C4D2D808D5406116 5523FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81C CF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF 227637A1EB08514946293740DD1A370D93304FD1509E26136 FDF877DE7015D2A6B68F0F34E84DE3CD83AE158EE0099A3 17EA8162EB498077EDDBCFCE1040F623346F8CA6CC79AB2 F98EE9FD166E0A9AE71FA33B5D685CF9341BF7AD4F7F829 EF9455387865D631C212F96789B190711E	512	6319E06677Ą53ċ975E6CBC61EF496F8AF5CF DFďBĐ202ě2B3ĞABF3BDD84ĘFB9Ě44ę0C8 BĒĴąŃ93Čę625101ļBA4īC1Fĥ8Ľ73DĿDE83E1 ACı858D7BŗœAŒDĮ4ŴB17Īŝŋĕ8ćŝŝ7ĀDĜFŪ Ē4D2ķ0Ů54ą1165523FŤ82ďF0EE4ĚEBη50ķěŗ 432ŴĤŠ3ĖĘA81CǤĉDăFĖB7A2lČD30Ɯğ5ĽC ăĆŃ5Ļ2ijĘĸ8ś2ƟĀ!ėBƐŐľŎŊ7ƓĶŨ3żȿŋ0ŀD1ʅ ă2ĖƵǵĝĈţżDZǫA6ŅŬNJ3ĹĸţƷCķ3Aŧŭƣ0099Aā ĉƩl2ƧĔ80ĈEĶĔĐŧǮNJŎŋ4Ĝ8ƊƵ7ȒBǕ98ƣ9ǵl6 ĄAȒE7ŚȓĴ5CƯŬĢŊ4œĥAšĥĝīĘĽƘį7ƁşĀʒDžȯƂ9 ź9ƷƕE	350
80	{L,o,r,e,m, ,i,p, s,u,m, ,d,o,l,o, r, ,s,i,t, ,a,m,e, t,,, ,c,o,n,s,e,c, t,e,t,u,r, ,a,d,i, p,i,s,c,i,n,g, ,e, l,i,t,., ,C,u,r,a,b ,i,t,u,r, ,e,g,e,s ,t,a,s, ,s,u,s,c,i ,}	76,111,114,101,109,32, 105,112,115,117,109,3 2,100,111,108,111,114, 32,115,105,116,32,97,1 09,101,116,44,32,99,11 1,110,115,101,99,116,1 01,116,117,114,32,97,1 00,105,112,105,115,99, 105,110,103,32,101,10 8,105,116,46,32,67,117 ,114,97,98,105,116,117 ,114,32,101,103,101,11 5,116,97,115,32,115,11 7,115,99,105,	110,105,18,342,138,13 58,19,91,391,246,383,3 98,89,60,23,9,533,2858 ,181,61,243,13043,449, 155,273,40,130,1011,7, 22,40,4,284,74,532,119 ,13,94,216,1150,17,72, 1,6,234,226,388,135,27 1,168,6699,101,40,230, 289,241,645,32,129,23 9,28,58,402,92,90,76,4 07,264,18,77,262,50,21 6,21,212,28,38,34,229, 250	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E2029 62B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F46251 01B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01D B34A0B17028933C68678989763D6FFCA6C4D2D808D5406116 5523FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81C CF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF 227637A1EB08514946293740DD1A370D93304FD1509E26136 FDF877DE7015D2A6B68F0F34E84DE3CD83AE158EE0099A3 17EA8162EB498077EDDBCFCE1040F623346F8CA6CC79AB2 F98EE9FD166E0A9AE71FA33B5D685CF9341BF7AD4F7F829 EF6117928A46399B94ADBAD3AE4F6D8E7F1C41E842E543F BF46088B628C59553A06EC6E4355E486E1799F26BB4807C04 03	576	6319E06677Ą53ċ975E6CBC61EF496F8AF5CF DFďBĐ202ě2B3ĞABF3BDD84ĘFB9Ě44ę0C8 BĒĴąŃ93Čę625101ļBA4īC1Fĥ8Ľ73DĿDE83E1 ACı858D7BŗœAŒDĮ4ŴB17Īŝŋĕ8ćŝŝ7ĀDĜFŪ Ē4D2ķ0Ů54ą1165523FŤ82ďF0EE4ĚEBη50ķěŗ 432ŴĤŠ3ĖĘA81CǤĉDăFĖB7A2lČD30Ɯğ5ĽC ăĆŃ5Ļ2ijĘĸ8ś2ƟĀ!ėBƐŐľŎŊ7ƓĶŨ3żȿŋ0ŀD1ʅ ă2ĖƵǵĝĈţżDZǫA6ŅŬNJ3ĹĸţƷCķ3Aŧŭƣ0099Aā ĉƩl2ƧĔ80ĈEĶĔĐŧǮNJŎŋ4Ĝ8ƊƵ7ȒBǕ98ƣ9ǵl6 ĄAȒE7ŚȓĴ5CƯŬĢŊ4œĥAšĥĝīĘĖŻ92Ğȧ3ȓļŸŶ¢Î Ƥ|ķȺFʒɄŤŗEƒƛIJȧƐńŎęǍƘÎąEĕPČĐ4ƁŧȬȳǴBű ţ7C83	386
90	{L,o,r,e,m, ,i,p, s,u,m, ,d,o,l,o, r, ,s,i,t, ,a,m,e, t,,, ,c,o,n,s,e,c, t,e,t,u,r, ,a,d,i, p,i,s,c,i,n,g, ,e, l,i,t,., ,C,u,r,a,b ,i,t,u,r, ,e,g,e,s ,t,a,s, ,s,u,s,c,i ,p,i,t, ,a,r,c,u,., ,}	76,111,114,101,109,32, 105,112,115,117,109,3 2,100,111,108,111,114, 32,115,105,116,32,97,1 09,101,116,44,32,99,11 1,110,115,101,99,116,1 01,116,117,114,32,97,1 00,105,112,105,115,99, 105,110,103,32,101,10 8,105,116,46,32,67,117 ,114,97,98,105,116,117 ,114,32,101,103,101,11 5,116,97,115,32,115,11 7,115,99,105,112,105,1 16,32,97,114,99,117,46 ,	110,105,18,342,138,13 58,19,91,391,246,383,3 98,89,60,23,9,533,2858 ,181,61,243,13043,449, 155,273,40,130,1011,7, 22,40,4,284,74,532,119 ,13,94,216,1150,17,72, 1,6,234,226,388,135,27 1,168,6699,101,40,230, 289,241,645,32,129,23 9,28,58,402,92,90,76,4 07,264,18,77,262,50,21 6,21,212,28,38,34,229, 250,131,37,3,171,473,2 27,67,573,281	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E2029 62B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F46251 01B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01D B34A0B17028933C68678989763D6FFCA6C4D2D808D5406116 5523FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81C CF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF 227637A1EB08514946293740DD1A370D93304FD1509E26136 FDF877DE7015D2A6B68F0F34E84DE3CD83AE158EE0099A3 17EA8162EB498077EDDBCFCE1040F623346F8CA6CC79AB2 F98EE9FD166E0A9AE71FA33B5D685CF9341BF7AD4F7F829 EF6117928A46399B94ADBAD3AE4F6D8E7F1C41E842E543F BF46088B628C59553A0374432D1EEA49A001DB68690F92880 5B3CBFC45487FBF836E000DE2B58D6B12BAB354C0A81E63 4A08BB43A55EF633BB0	640	6319E06677Ą53ċ975E6CBC61EF496F8AF5CF DFďBĐ202ě2B3ĞABF3BDD84ĘFB9Ě44ę0C8 BĒĴąŃ93Čę625101ļBA4īC1Fĥ8Ľ73DĿDE83E1 ACı858D7BŗœAŒDĮ4ŴB17Īŝŋĕ8ćŝŝ7ĀDĜFŪ Ē4D2ķ0Ů54ą1165523FŤ82ďF0EE4ĚEBη50ķěŗ 432ŴĤŠ3ĖĘA81CǤĉDăFĖB7A2lČD30Ɯğ5ĽC ăĆŃ5Ļ2ijĘĸ8ś2ƟĀ!ėBƐŐľŎŊ7ƓĶŨ3żȿŋ0ŀD1ʅ ă2ĖƵǵĝĈţżDZǫA6ŅŬNJ3ĹĸţƷCķ3Aŧŭƣ0099Aā ĉƩl2ƧĔ80ĈEĶĔĐŧǮNJŎŋ4Ĝ8ƊƵ7ȒBǕ98ƣ9ǵl6 ĄAȒE7ŚȓĴ5CƯŬĢŊ4œĥAšĥĝīĘĖŻ92Ğȧ3ȓļŸŶ¢Î Ƥ|ķȺFʒɄŤŗEƒƛIJȧƐńŎęǍƘÎ0ǪĿuǰƣŖȒô1Ŷ6Ɓ9 NJɏ8ţ5ĮēĐ4ƒ87ĻĝƵĄôţĭŭƇźĭıČ4C0ÛđâŴńűÎƘR ŋBǠ	421
100	{L,o,r,e,m, ,i,p, s,u,m, ,d,o,l,o, r, ,s,i,t, ,a,m,e, t,,, ,c,o,n,s,e,c, t,e,t,u,r, ,a,d,i, p,i,s,c,i,n,g, ,e, l,i,t,., ,C,u,r,a,b ,i,t,u,r, ,e,g,e,s ,t,a,s, ,s,u,s,c,i ,p,i,t, ,a,r,c,u,.,	76,111,114,101,109,32, 105,112,115,117,109,3 2,100,111,108,111,114, 32,115,105,116,32,97,1 09,101,116,44,32,99,11 1,110,115,101,99,116,1 01,116,117,114,32,97,1 00,105,112,105,115,99, 105,110,103,32,101,10 8,105,116,46,32,67,117	110,105,18,342,138,13 58,19,91,391,246,383,3 98,89,60,23,9,533,2858 ,181,61,243,13043,449, 155,273,40,130,1011,7, 22,40,4,284,74,532,119 ,13,94,216,1150,17,72, 1,6,234,226,388,135,27 1,168,6699,101,40,230, 289,241,645,32,129,23	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E2029 62B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F46251 01B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01D B34A0B17028933C68678989763D6FFCA6C4D2D808D5406116 5523FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81C CF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF 227637A1EB08514946293740DD1A370D93304FD1509E26136 FDF877DE7015D2A6B68F0F34E84DE3CD83AE158EE0099A3 17EA8162EB498077EDDBCFCE1040F623346F8CA6CC79AB2 F98EE9FD166E0A9AE71FA33B5D685CF9341BF7AD4F7F829	736	6319E06677Ą53ċ975E6CBC61EF496F8AF5CF DFďBĐ202ě2B3ĞABF3BDD84ĘFB9Ě44ę0C8 BĒĴąŃ93Čę625101ļBA4īC1Fĥ8Ľ73DĿDE83E1 ACı858D7BŗœAŒDĮ4ŴB17Īŝŋĕ8ćŝŝ7ĀDĜFŪ Ē4D2ķ0Ů54ą1165523FŤ82ďF0EE4ĚEBη50ķěŗ 432ŴĤŠ3ĖĘA81CǤĉDăFĖB7A2lČD30Ɯğ5ĽC ăĆŃ5Ļ2ijĘĸ8ś2ƟĀ!ėBƐŐľŎŊ7ƓĶŨ3żȿŋ0ŀD1ʅ ă2ĖƵǵĝĈţżDZǫA6ŅŬNJ3ĹĸţƷCķ3Aŧŭƣ0099Aā ĉƩl2ƧĔ80ĈEĶĔĐŧǮNJŎŋ4Ĝ8ƊƵ7ȒBǕ98ƣ9ǵl6 ĄAȒE7ŚȓĴ5CƯŬĢŊ4œĥAšĥĝīĘĖŻ92Ğȧ3ȓļŸŶ¢Î	477

	,C,u,m, ,s,o,c,i,i,s,}	,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,112,105,116,32,97,114,99,117,46,32,67,117,109,32,115,111,99,105,105,115,	9,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281,500,284,87,13,75,85,68,86,55,306,33	EF6117928A46399B94ADBAD3AE4F6D8E7F1C41E842E543FBF46088B628C59553A0374432D1EEA49A001DB68690F928805B3CBFC45487FBF836E000DE2B58D6B12B140677D864C831E32CDC765289484F57FD5C1E01EE145553CEFC279D36533AE62220A077C0C3C1593BD9C2A12E5D9834185168760CA25FAB0C8C2AE032A400FB		Ƥ!ķȺFȝ℧ŤţEƒƛlJáɛńÕęÅĶĨ0QŁuřɋɻŖŘô1Ỹ6B9NJy8ţ5]ēÐ4ƒ87Ļĝ℀ĄôţīūĆ'žī1Ġć7ķ64NāEuỈCbfŝ4κĠţ[DġėŒoɋ?ĶċCĘCΘ9LjɬŋīÕÛ0ŴĈCłêDZŊj9¡A1ŞşÔâ1Ûl৳6łDŽŮıɬęƱĄuŗŖôĻ.	

EK-3. BLOWFISH ŞİFRELEME – LZW SIKIŞTIRMA

	S	*A*	*C*	*BLOWFISH*	*x*	*LZW*	*x'*
10	{L,o,r,e,m, ,i,p, s,u,}	76,111,114,101,109,32,105, 112,115,117,	110,105,18,342,138,1358,19,91,3 91,246	18a0d238c0f940049c1326e0f0e7be2f03751b9f7 983e88e127fac64dff21a02f86d62fab33d217257 a1c4b1a06975ba	96	18a0d238c0f940049c1326eĉ0e7be2f03751b9f7983 e88e127fac64dff21Ăĝ86d6ĝab33Ą17257a1c4bĻ06 9ġba	89
20	{L,o,r,e,m, ,i,p, s,u,m, ,d,o,l,o, r, ,s,i,}	76,111,114,101,109,32,105, 112,115,117,109,32,100,111 ,108,111,114,32,115,105,	110,105,18,342,138,1358,19,91,3 91,246,383,398,89,60,23,9,533,2 858,181,61	1228bca0aa0319864d6547880d47671b440d2c6b 31de60895ef0496c3868376b4994723fa77e0584f 8abd8f426f6a9c83308ca4f6c6918bd1bcd6eb5f1 d436a39fea6f26ffbfa1bd76e0607723027f7d	160	1228bca0aĆ319864d6547880dē671b44ė2c6bĈde6 0895ef0496c3č837ġĺ9ē23fa77e0584f8abd8f426f6a 9cĶ3ĩąŇIJ691ăđĜcÞeb5f1Ę3œ39feaöÕffbŁĜdĸŃĬ ŁĽ027f7d	133
30	{L,o,r,e,m, ,i,p, s,u,m, ,d,o,l,o, r, ,s,i,t, ,a,m,e, t,,, ,c,o,}	76,111,114,101,109,32,105, 112,115,117,109,32,100,111 ,108,111,114,32,115,105,11 6,32,97,109,101,116,44,32,9 9,111,	110,105,18,342,138,1358,19,91,3 91,246,383,398,89,60,23,9,533,2 858,181,61,243,13043,449,155,2 73,40,130,1011,7,22	02631ec8751ddf20c6086a743f3cf263c6728d07f b7b973fded73b4e41c7a5ac4597756e87c0a098fa d3c9569503684be40b58f2324167f2f20aec2f969 7e2eef1220e18b4c7e2d6bf8ec7655c257b1a0af6 458b7bda45d3618b9663b9aadda4390394f7c09e 7a154f10062ee419b042104970b4ed13a24b	240	02631ec8751ddf20c6086a743f3cčĂĐ728d07fb7b9 7ĘdedĪb4e41c7a5ac45ĩĈ6ećc0a098fadĚ956Õ0368 4bIJ0b5Ñ232ij6Ĥ2čÑąš9Ô7e2eef12Ďe18Íjūd6bf8ą 7655c25h1ńaf6ĻụhdaĻ.Đ6ỤĨ6ĂĪa'nǫė9Ô94f7l9eĶ1 5ŋ1006ŪIJ19b042ƨ4ĩɾıd13aŝb	191
40	{L,o,r,e,m, ,i,p, s,u,m, ,d,o,l,o, r, ,s,i,t, ,a,m,e, t,,, ,c,o,n,s,e,c, t,e,t,u,r,}	76,111,114,101,109,32,105, 112,115,117,109,32,100,111 ,108,111,114,32,115,105,11 6,32,97,109,101,116,44,32,9 9,111,110,115,101,99,116,1 01,116,117,114,32,	110,105,18,342,138,1358,19,91,3 91,246,383,398,89,60,23,9,533,2 858,181,61,243,13043,449,155,2 73,40,130,1011,7,22,40,4,284,74, 532,119,13,94,216,1150	656d6c2d065f1c109ebc71870762d0e96e1e24cb 02d0ed6b87e9599bc6b54e48a8a12cc4365d6eab bf2c13b73ff3b948fa33c1eabfa8c3ecae470f37b8 bf9a5f7234b7b290b69fd08ea54b639d45049f3a1 5a42f9440acf489631568f078213c79a9af5f77aba a07ccee1ccb451cc2a9cd8b4394f6a23e6d8b088e 65ebb6c63536d704cd9dcf08a332b23c982cdddc 2dc4c705e643f4cb8e902cee7b37fee5cd	320	656d6c2d0Āf1c109ebc7187076Ć0e96e1e24cb0Ěe ăbčĜ599đ6b54e48aķ12cc43Āăeabbfl.13b73ffn9Ķf a33ČlŊкc3ecajÈÔ7ÎŊ9a5f7234Ɒb290b69fċ8lijU3 9d4504Ŵ3a15a42fŐ40acfĶĝ3b68fè82ñĒťťf77Na aėļeğlbžċąa9cd8ţŹ4f6aŪeĀY08ŷĀĐỤc635LdĖġdż Əsœ32Ůf9ķʊddąÛġĖ52ľƐĤŷŰĻƨŢ37fƨ5Ʊ	238
50	{L,o,r,e,m, ,i,p, s,u,m, ,d,o,l,o, r, ,s,i,t, ,a,m,e, t,,, ,c,o,n,s,e,c, t,e,t,u,r, ,a,d,i, p,i,s,c,i,n,g,}	76,111,114,101,109,32,105, 112,115,117,109,32,100,111 ,108,111,114,32,115,105,11 6,32,97,109,101,116,44,32,9 9,111,110,115,101,99,116,1 01,116,117,114,32,97,100,1 05,112,105,115,99,105,110, 103,	110,105,18,342,138,1358,19,91,3 91,246,383,398,89,60,23,9,533,2 858,181,61,243,13043,449,155,2 73,40,130,1011,7,22,40,4,284,74, 532,119,13,94,216,1150,17,72,1, 6,234,226,388,135,271,168	a8bb615f393693fdba30e5209f13b9c962b79acbf d4d8d4194314586414bd7a22e2072aabc85d823e e61e5405fdfd6f256244b21963c86c5aac2cba85f 39b39c6cc44051f66727836945998b55874adfce 4fdbbd3bf15470ea7a6f95fba32a3b83de060f2c7 5a83664a9866c56cfa931565b5e34a82f818f8b8f 1ab70d4ca006fdcf4602ff44d48770d703bdc91d5 36441f19652438b930fd7daa1a12bfbff753889fd 389051ef5d61f2dd33fd422b8d20e3efb71881c40 716dfa0bdfe9c2931	384	a8bb615f3936ĉfdba30e5209f13b9c962b79acbč4d8 d4194314586Ĭ4bd7a22eĕ72aabc85ĩ23eeĄē40Ćdč6 f25ğ4ķ2ĩ63Ņ6c5ĺc2ĥĀĆĈbĈcŠčŕŐ1f66ĺ78ĊĮ599ā 5ij74aŒce4čĂdĚĘ547ĖaÍŔ9Ćd'3ŁÈŷde060fc7Ţýų ƁžųšŠfaĉą65b5e3Ɓ82f818ʒbʒ1ŃƎīca0Űlčcf4ŅzLjĨ 487ǀƎĚdĝ1d5ĈřüŜĚjãĉηĹdŧs1GfĦfo388ėÐʒŰefŇ AŗfdĐ3hĮ5dĕUdzġɛʐŮ0716Œ!кfeĜ2T	274
60	{L,o,r,e,m, ,i,p, s,u,m, ,d,o,l,o, r, ,s,i,t, ,a,m,e, t,,, ,c,o,n,s,e,c, t,e,t,u,r, ,a,d,i, p,i,s,c,i,n,g, ,e, l,i,t,., ,C,u,r,}	76,111,114,101,109,32,105, 112,115,117,109,32,100,111 ,108,111,114,32,115,105,11 6,32,97,109,101,116,44,32,9 9,111,110,115,101,99,116,1 01,116,117,114,32,97,100,1 05,112,105,115,99,105,110, 103,32,101,108,105,116,46, 32,67,117,114,	110,105,18,342,138,1358,19,91,3 91,246,383,398,89,60,23,9,533,2 858,181,61,243,13043,449,155,2 73,40,130,1011,7,22,40,4,284,74, 532,119,13,94,216,1150,17,72,1, 6,234,226,388,135,271,168,6699, 101,40,230,289,241,645,32,129,2 39	dfd9c6f8a43308c96e642adc288e5120d791b0ff1 481318dc2b0e0aa02d3ef0f042b40d6f4ae3b0fb5 db211671108a9081557df1df92860c6c9a57e0f72 8fca37a148a0d1da743c719d47ac03ef8cf9778d2 cbd3bcd5ed9acd06e4d1cbf363df209dc8c215746 3ce657a7eb210f962df062d81b5a2162c183dc37 8db0a112d0c2ec3fb08d302b1c45a4ec4bf331a1d 5fb745996b4deec64dc0a703ba6ec687ea840a4dc e34ae988a4aca85894c4bcec29f38d7b35c0c9dac 5a2490b6e27ef3d2b177fb88a3761a9a99eed0c5d be5d62e7b1ccac1d37fb0c1baada332e3d97e92b1 984c279baec04a	464	dfd9c6f8a43308c96e642adc288e5120d791b0ff148 1318Ê2Ĥe0aa02d3efĥ0ēb4ğą4ae3Ĥfb5db21167ÕĈ a9Č1557Ā1Ā9Ę60ĄĎařIJf7Ęfca37aĭćğŜa7ĉcö9d4 Ůc0ĹĆcf977Į2cbкbcd5eĂađ0Đ4d1cf363ĀĞŹcčðř 4ƛcdřŴeÕ1ĥďķĻ6ķĨnaõƱcïŰlcūĮĤůĝđ82e2ñČĸĶb Ħ45ĈDŽ4bǩĬůǫńẃ59ďĿdeDŽĒĖij7žbaĐĄ87ea8łĈ ÉŅNe9ęĈĴǫ5894cïĸė9ķĮ7b35ŽĎdƑĂ24fbÐ2ǩkđÍ1 ъńĜŭ61ŔR9əlcƆbědƱeâɬūƼĸ7džƼGěŰ3!ŰlбDzüdz åř9GDŽĽa	322

70	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77	5c4cc7627a722b91272623894b8dec1c949ed27249496799f55a2bfcfe3df8d4629f1b96c347a886bab1e16013a815bfe8406ed809e6236915b52f539bb6ca205185d1b9e28df2f31c73e2be9535b762668b7b12f71c512bff6d5f13bade77d7d935fab13d9b0b33ae3948518c0543bb9f451fb01857905051b98ce1aea3eeeb64c676e1173fb2111d2948050dfb80183a6fab0d59ef26448ef416582448e93bef820ab811699b04a0733ba7778bfe1d532c22dd86591331addee64ad63c18db3529ee8b613692c76f60012e0459782ba8d3a7ed12b68795ba076b15e6f01c7e32ff64b41a8f22e2f2ce0a60ba4bae43a88d52ab0650719b2c52101aae8da1610f2e9b823f386391	528	5c4cc7627a722b91ć2Ć3894b8dec1cě9edĐ4Ğ6799f55aČfcfe3dfĘ4ĆĬ1č6c34Ĉ886bab1e16013a815bĬ8406Ġ80ğĒ6ĎÖ52ĩ39bbĮĬ05185dĹġ2ĘfŠ3Ġ73eČe953Ŏąđ68b7ŅŠ7Ĝūĭf6d5fŋŅę77dɎ9Žfńŋdţ0b3Ŏıěūc̈Ũ43ŤĪ4ūfbĐūĦŪŬŮ8cÑaeaŷeeťĂĥŖ11ŶҐ2!1ġơɣYijėuŐ6ǩUơğỤ6448ef4ň582ůŭĦbŬĂ0ńŎřĥUʹ4a0ŸΣĉĜḂĺlj22cċddĿ5ĎƠ1aÆ₅Ûâ63ėĘΘşğeB6ŋŜṅąđ0Ɒ2e0T97ĂŃĘÕ7Ġđ'ť8ĦŐʒąŅ5eǑĐĄıŠƉĖəōỤşŠṅŢạ'nŃÉɛəōĘ̧ṣńr5j1ţṅş1ƉąɛĘañŸỤźė23fềĬĎ	351
80	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250	ae65d2061dbef5d348f24ba5001b00420dc045925696ca9aĄf39dĢffe857415bc2a4c2ba3d9e6f948cc46f446895649a4da341c9fb5ac99cce2e900a71baca82cf9c9416d877c572d20ab9a766b008a59f2f4c2346f46797ddc0e0eb456d30e6801a47a5d020c0ced291c487ec64b4f73fece5b4c3083008db599e3badd8825b16b1f27766fddf4ffb51a014d60f3e67ac5ede4d2fbfaad3d153c10e419975fb49ba6116007ac4b05f6275582b168e78cab7c5b60c8f29178f64ae0b77402726c6aadc8864da8b0f0b071e73f4e355df7f3e18875397563e442e1367e85f80abc85d10a68b1ed21d85c228f9170ef9aed57c35b4f9aae4b4057a27d04e6014e43f97a49bea27a2aa790d5be7bad9bd25f9c19c8e3982cd2f86d355de8e302fb9bbdcb5c4af5c5b	592	ae65d2061dbefã348f24ba5001bė4ądc045925696ca9aĄf39dĢffe857415bc2a4ĽĚ3d9āf9ďcc46f4Õ89ĥ4ī4daĎ1c9fb5aś9Ɒe2e9ėa7ęŠa82cŃśĹ6d877cķ2Ą0abī766Ĕ08ēŜ2ÕĽĎōÕ797dĞ0eɎbġŴ3Ɏ68ĘĿ7ěd0ąğceA9Śď7ec6ē4f73ĴŖĻH8ŀbĉĢŅ3Ĕƒ8ŮĻ1ƀ1đŷБfƴÓijŞ1aĘŖ60ĮāOŹƨeŖĐbfaĬŃĺ3c1ɎĹ9Ɛ5ŝŘĔćdžėÛē0Ǫ62755Ůbdž8e78ĩbŸĻÛcÐ2Σäf̈UʹÃ0Ëкɋ72Ĭ6ăĞ!Uŧ8ĖfŘ0ŪÄy4e3ÅdŬÛ1!ṅịṅ63ùĜe136TĶfЈŽĭăgaШǽƨ2ĈĶĽ2ĐīɎÑĀıŸâłrĀuǵƠŃƆ|zŮ1ɀ4yˇƐĿ9ĈaÑɤăƟ0ıĈ7lɕĄǪŢ1Ţóị̊Ůcī8ĶĘjÁơ|$ɕbĞŞŋąČŹb	393
90	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,p,i,t, ,a,r,c,u,., ,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,112,105,116,46,32,97,114,99,117,46,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281	d0c72582de6607c41c0d1807bbc2054ad4e8cb9dfb7ba87103cec569d89bfef7850cab78cad002752246fe9de96a958bb7684c0bffd5006b490beafc772dce9ac0b9388d335d4021df730a758f60e001e8beb656c3a6177214d27e4b7eeb9cf5726b480d37bc510f9ba77c044d95d9f64916e10cca0c74f7e4883eaed65af1b9b525b4b7246a02c11c114a6ea116ce1aa3fd399af8b851dbf89177fc73d1930cb68cd0e9472a7abb24bb2f7a774420ccfe34ed0cb61511e620f4ceeb564c4cae7056d3d8ba770a65ad52321ea87a55881f36c7ee5571b66f676fb39d1db81c55f2fa19f5526960639372c51993aeb6c3310481d7450206d44fc823a068bad2950eddd1c70283dd72b91320f11505767abc0f87892e3f696ceae06afbec104ecc28f47ad244fede8528fc0bea15cf9ea3f9e5e38143b0d5f147ea5552821951a5270226ec2a8e449b	656	d0c72582de6607c41c0d18Čbbc2054ad4e8cb9dfb7ba87103cec56ĥ89bfef785āaĪĢĞ0027522461Íĥe96a9ąė7684dĹfdĿ06b490beafĂăđı9aś9388d335ğŋ1Ħ730añ8fčeĻ1ģŦb6Ĵc3a617ă14dŃe4ĪeeĤcf5ăš4ĕŵĩıjIfкbč04hʋŘd9ϐţ16eĮccaā74ļĸu3tedđŬ1ĤbʾnŠŢĪŋŒɳc1ĐÚậzaÚ6ı1aa3ŝ39Ůf8bl'ZĹķF7ŭŽĕüāDQĂÓ4ăbŁbŋė2ļţŢĚɛÍ3ĠĀậ£5ŬčĖfŚʎNJ6ŚŠae7ĕ6ŵdgţđžŞ232CĨ7a5q81f3ŭ7ʎāĭDõ6ŗhǎēdĞĐâfǽŭUà2ĵċó6ã3ăƨÆƎШŭŶĮɑIZsŁĚŞŢûĆƎŠţŔ0DŽdēĂŋ83ģãĤ1q0fŰŁÓĮŁdGĽ92eãBıÙŠŰŦðIJęƁHʊĜŋsDŽg'nƄśt15ŋ9tāz5ɷę43bĒ5ŀHʊtȧʊ8Ŕǔ'nÛƉzęĩĶţb	430

100	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,p,i,t, ,a,r,c,u,., ,C,u,m, ,s,o,c,i,i,s,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,112,105,116,32,97,114,99,117,46,32,67,117,109,32,115,111,99,105,105,115,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281,500,284,87,13,75,85,68,86,55,306,33	fb3729ca3afb371143df5e084984851f55b5e7ed848335f51ca690bda323d53543c4d9d65e607380c8c0611a081d4c184923d4b78ae6c2d68a4a31da49cfc6da596d8077677593bd90accf307513ef0a9445f1bdd56b6d09229c55c5933b2d4d32ca1d17a2c4c422737fbf4442004e412d1ca348ba64e4e37e4d92ce7fe6dcc385c4a635a729d8bc7366eea39036ea99425ece35cd9a67832496f6f4ca2b5a17d4670791b721dea62fbd934a224b46f1755247cba1cacc74045c4a1cd9e2c13a1fa45622ef21794a1a8d82a8d0eb95b21c73ccc9c9db2dd3475cf29131b49ce9bc9c5463062e93c2aa93567747038b76ba07b9ad902cf45a9e377343e01ff52fd689e94f7301689bd6596f614452cd58365ce963e7c71ff459e6968cc5c1281226ffb8c55332e9d85614870a051ea8b4803430b9ca22f83603f27276d5f444f64b354cc5acc0da3e41d29fe4134ca219f19303e8fc77f837138541e620fbbf14c204bff31d6ea3757dae7de9b2ed177f	736	fb3729ca3aĀĂ1143df5e0849ĕ851d5bĒ7edĘ335d1Ć690bdć2d'5ĥĎc4d9d6Ē607380c8c06ČaĖ1d4c1ĕ9İŋb78ae6c2Ĺf4ćŊaĖcfc6Ɩ59ťŀ776ūt3ĩacƮ3ĽĚ3ef0a944Ħ1ĭd56bť0ÕĄc55đ9ĤbŚĶ32ĆĎ17alÕ422l'7Āfĩŋ004e41ŚĬć48ba6ΣΣĂļķleƠĮŗdŴŁ∃ŝ6ĥaăĸƱcl'66eećĩ3njŽž2Ēceĥcķ YŘħŲĖ6fĂÕƛĞaķŋūĽ9ƁăĽDă62Āķ3ŝΘ4b4Ăķǫ247cUƮŴ74ƨlČŭ9el1ČĕŠbΘź2ķžĞa8Ġ2Ṣd0eb9ĝÛlj3Ŵcąĸƒdd Ǵ7∃fĄȍƄše9bâđṅÝĵŋẓţŽĥŭĖ70Ł œƆñ7Ⅱaķ0ŀP5ŽűūǴŹ0ĕđDZśÎžſ'd68ŧĹtĂŋſ5lƀ8Ǒ∃ŋ!2ljɔbĪĨgŴ∃T8T2ĂĀŃǫĤ2ŋĢƀč8Ʊ ňĚăƱuʌĎĬąƛDZŀ3Ôă ŨЬPſâƤijɸцńĮŹţdĄfļöæŨ9f1ƐΛe8ţūfl7öęţɾ20ĀbↄǫÕ ƥff3ƉðĂ57Ɩ2dŋƒġķƇ	470

EK-4. RSA ŞİFRELEME – LZW SIKIŞTIRMA

	S	*A*	*C*	*RSA*	*x*	*LZW*	*x'*
10	{L,o,r,e,m, ,i,p,s,u,}	76,111,114,101,109,32,105,112,115,117,	110,105,18,342,138,1358,19,91,391,246	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9	73	4-Ā-c-eā-ă2ĄĆbČ6-d-5ČĂĎČąĚ-ĉĚĆ3Ċĝ̃ĀąĎğĊĚĐ9	46
20	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4	143	4-Ā-c-eā-ă2ĄĆbČ6-d-5ČĂĎČąĚ-ĉĚĆ3Ċĝ̃ĀąĎğĊĚĐ9č-ČĎġ-ĝĕČĝąĥăąĚīīĊĉĎķĚČĚēīĆlJā	69
30	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, , a,m,e,t,,, ,c,o,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5	221	4-Ā-c-eā-ă2ĄĆbČ6-d-5ČĂĎČąĚ-ĉĚĆ3Ċĝ̃ĀąĎğĊĚĐ9č-ČĎġ-ĝĕČĝąĥăąĚīīĊĉĎķĚČĚēīĆlJłđď-ķĘăĐķĐĐıĆĉĉĵ-aŇĆĐĴĆĎřĂĆĠŔĤđ	99
40	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, , a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c	291	4-Ā-c-eā-ă2ĄĆbČ6-d-5ČĂĎČąĚ-ĉĚĆ3Ċĝ̃ĀąĎğĊĚĐ9č-ČĎġ-ĝĕČĝąĥăąĚīīĊĉĎķĚČĚēīĆlJłđď-ķĘăĐķĐĐıĆĉĉĵ-aŇĆĐĴĆĎřĂĆĠŔĤŅąŘŗšČĐąŕŪęŖĚėĜłķĝūĒĀĥůÂĉc	124
50	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, , a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b	359	4-Ā-c-eā-ă2ĄĆbČ6-d-5ČĂĎČąĚ-ĉĚĆ3Ċĝ̃ĀąĎğĊĚĐ9č-ČĎġ-ĝĕČĝąĥăąĚīīĊĉĎķĚČĚēīĆlJłđď-ķĘăĐķĐĐıĆĉĉĵ-aŇĆĐĴĆĎřĂĆĠŔĤŅąŘŗšČĐąŕŪęŖĚėĜłķĝūĒĀĥůÂĉŜŠūŢńýŅĎŴŅběŁůĎŒŅŕşŶĩ	144
60	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, , a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C ,u,r,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-3	437	4-Ā-c-eā-ă2ĄĆbČ6-d-5ČĂĎČąĚ-ĉĚĆ3Ċĝ̃ĀąĎğĊĚĐ9č-ČĎġ-ĝĕČĝąĥăąĚīīĊĉĎķĚČĚēīĆlJłđď-ķĘăĐķĐĐıĆĉĉĵ-aŇĆĐĴĆĎřĂĆĠŔĤŅąŘŗšČĐąŕŪęŖĚėĜłķĝūĒĀĥůÂĉŜŠūŢńýŅĎŴŅběŁůĎŒŅŕşŶĆ-ĥĥĝõŝşŤœśŦĸœĐşĥĐĎĎŮĆĒõĶī	169
70	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, , a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C ,u,r,a,b,i,t,u,r, ,e,g,e,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,32,115,105,116,44,32,108,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,10	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,1	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-	503	4-Ā-c-eā-ă2ĄĆbČ6-d-5ČĂĎČąĚ-ĉĚĆ3Ċĝ̃ĀąĎğĊĚĐ9č-ČĎġ-ĝĕČĝąĥăąĚīīĊĉĎķĚČĚēīĆlJłđď-ķĘăĐķĐĐıĆĉĉĵ-aŇĆĐĴĆĎřĂĆĠŔĤŅąŘŗšČĐąŕŪęŖĚėĜłķĝūĒĀĥůÂĉŜŠūŢńýŅĎŴŅběŁůĎŒŅŕşŶĆ-ĥĥĝõŝşŤœśŦĸœĐşĥĐĎĎŮĆĒõĶλŅĕĿţć	189

		1,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,	8,77	3-e-5-b-e-2-b-e-d-c-5-e-3-5-e-3-c-e-a-9-e-d-c-a-e-5-9-d-e-4-b-e-a-a		žĝoŋĝřřbŤŕœΘłěŕa	
80	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-3-e-5-b-e-2-b-e-d-c-5-e-3-5-e-3-c-e-a-9-e-d-c-a-e-5-9-d-e-4-b-e-a-a-e-5-9-5-e-2-c-e-5-4-9-e-5-4-e-5-4-5-e-5-b-e-6-b-e-6-d-e-5-5-3-e-5-2-c	573	4-Āc-eā-ă2ĄĆbČ6-d-5ČĂĎČąĚ-ĉĚĆ3ĊĝĀąĎğĊĚĐ9č-ČĎġ-ĝĕČĝąĥăąĒīīĊĉĎķĚČĚĕīĆIJłďď-ķĘăĐķĐĐıĆĉĉĵ-aŇĊĐĴĆĎřĂĆĠŔĤŅąŘŗŝČĐąfŪęĘĖèĜłķĝūĒĀĥůÂĉŜŠūŢńýŅĎŴŅběŁůĎŒŅŕşŶĆ-ĥĥĝŏŝşŤœśŦĸœĐşĥĐĎĎŮĆĒŏĶƛŊĚŁţć žĝoŋĝřřbŤŕœΘłěŕzьžźšƎœşŵžĽĬěĖšβšź	206
90	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,p,i,t, ,a,r,c,u,., ,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,112,105,116,32,97,114,99,117,46,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-3-e-5-b-e-2-b-e-d-c-5-e-3-5-e-3-c-e-a-9-e-d-c-a-e-5-9-d-e-4-b-e-a-a-e-5-9-5-e-2-c-e-5-4-9-e-5-4-e-5-4-5-e-5-b-e-6-b-e-6-d-e-5-5-3-e-5-2-c-e-4-6-4-e-6-a-e-6-e-4-a-4-e-d-a-6-e-5-5-a-e-9-a-e-2-a-6-e-5-b-4	637	4-Āc-eā-ă2ĄĆbČ6-d-5ČĂĎČąĚ-ĉĚĆ3ĊĝĀąĎğĊĚĐ9č-ČĎġ-ĝĕČĝąĥăąĒīīĊĉĎķĚČĚĕīĆIJłďď-ķĘăĐķĐĐıĆĉĉĵ-aŇĊĐĴĆĎřĂĆĠŔĤŅąŘŗŝČĐąfŪęĘĖèĜłķĝūĒĀĥůÂĉŜŠūŢńýŅĎŴŅběŁůĎŒŅŕşŶĆ-ĥĥĝŏŝşŤœśŦĸœĐşĥĐĎĎŮĆĒŏĶƛŊĚŁţć žĝoŋĝřřbŤŕœΘłěŕzьžźšƎœşŵžĽĬěĖšβšpcń Ď5ňŠlŠ]ɗĈh5ćŕűČ4	224
100	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,p,i,t, ,a,r,c,u,., ,C,u,m, ,s,o,c,i,i,s,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,112,105,116,32,97,114,99,117,46,32,67,117,109,32,115,111,99,105,105,115,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281,500,284,87,13,75,85,68,86,55,306,33	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-3-e-5-b-e-2-b-e-d-c-5-e-3-5-e-3-c-e-a-9-e-d-c-a-e-5-9-d-e-4-b-e-a-a-e-5-9-5-e-2-c-e-5-4-9-e-5-4-e-5-4-5-e-5-b-e-6-b-e-6-d-e-5-5-3-e-5-2-c-e-4-6-4-e-6-a-e-6-e-4-a-4-e-d-a-6-e-5-5-a-e-9-a-e-2-a-6-e-5-b-4-e-2-c-c-e-5-b-d-e-b-a-e-4-6-e-a-2-e-b-2-e-9-b-e-b-9-e-2-2-e-6-c-9-e-6-6	709	4-Āc-eā-ă2ĄĆbČ6-d-5ČĂĎČąĚ-ĉĚĆ3ĊĝĀąĎğĊĚĐ9č-ČĎġ-ĝĕČĝąĥăąĒīīĊĉĎķĚČĚĕīĆIJłďď-ķĘăĐķĐĐıĆĉĉĵ-aŇĊĐĴĆĎřĂĆĠŔĤŅąŘŗŝČĐąfŪęĘĖèĜłķĝūĒĀĥůÂĉŜŠūŢńýŅĎŴŅběŁůĎŒŅŕşŶĆ-ĥĥĝŏŝşŤœśŦĸœĐşĥĐĎĎŮĆĒŏĶƛŊĚŁţć žĝoŋĝřřbŤŕœΘłěŕzьžźšƎœşŵžĽĬěĖšβšpcń Ď5ňŠlŠ]ɗĈh5ćŕűČşpřdžūČ5ĘŽĎĽůƏČbőĬăbŇ	243

EK-5. DES ŞİFRELEME – ARİTMETİK KODLAMA

	S	*A*	*C*	*DES*	*x*	*ARİTMETİK KODLAMA*	*x'*
10	{L,o,r,e,m, ,i,p,s,u,}	76,111,114,101,109,32,105,112,115,117,	110,105,18,342,138,1358,19,91,391,246	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuwnSaF8z+hrLw==	56	0((x)F1*)*8*E;)](7)lF.)])wv)?(5*-V*1))(=8(;]/R)iL)i*1r)Rev(*.5)z(J*4[(4*0e*8m(=_2*HH(*>	87
20	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDXhLUgGrPTFs=	108	0((x)F1*)*8*E;)](7)lF.)])wv)?(5*-V*1))(=8(;]/R)iL)i*1r)Rev(*.5)z(K(J*H)ug)+(G)6A74]*<*A)+aT***G(H,*5e*Fn*H0P)t(:(F)|?(=p*8*7)>(3R)@)7(=*K))(@>~v~*@(:*E(N)m(.R7+).}) .()i	168
30	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m, e,t,,, ,c,o,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUemYmC60mhj4nw==	152	0((x)F1*)*8*E;)](7)lF.)])wv)?(5*-V*1))(=8(;]/R)iL)i*1r)Rev(*.5)z(K(J*H)ug)+(G)6A74]*<*A)+aT***G(H,*5e*Fn*H0P)t(:(F)|?(=p*8*7)>(3R)@)7(=*K))(@>~v~*@@3)dR)MP)X*)*:D0)U)F*Q)Q*Q)-)r((*9*+|(O*I*.){(Ac)Q*hK)25*3)_8))g*6>)=(C+-)Lf)Il*)o0	230
40	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m, e,t,,, ,c,o,n,s,e,c,t,e,t, u,r,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnDqxtz6it6icsTC/v20ML+mB7lGBPU+5fLMV5WmAz+1u5QKRmgPUVKCb4=	204	0((x)F1*)*8*E;)](7)lF.)])wv)?(5*-V*1))(=8(;]/R)iL)i*1r)Rev(*.5)z(K(J*H)ug)+(G)6A74]*<*A)+aT***G(H,*5e*Fn*H0P)t(:(F)|?(=p*8*7)>(3R)@)7(=*K))(@>~v~*@@3)dR)MP)X*)*:D0)U)F*Q)Q*Q)-)r((*9*+|(O*I*.){(Ac)Q*hK)25*3)_8))C(@)h(rZ){(G(8(C(F*>*M*5)?)<@(L6?*K*8)8P)b(9(K*>)5+Z)E(I*+)))N){(Am)W)8)n)k(Cp*F(G:)^l)C*5)a)@((2(B)g))	314
50	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m, e,t,,, ,c,o,n,s,e,c,t,e,t, u,r, ,a,d,i,p,i,s,c,i,n,g,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,115,112,105,115,99,105,110,103,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnDqxtz6it6icsTC/v20ML+mB7lGBPU+5fLMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTloxYe/YqS3NlBJuMmLwjN48c1s2IZvyjKKg==	248	0((x)F1*)*8*E;)](7)lF.)])wv)?(5*-V*1))(=8(;]/R)iL)i*1r)Rev(*.5)z(K(J*H)ug)+(G)6A74]*<*A)+aT***G(H,*5e*Fn*H0P)t(:(F)|?(=p*8*7)>(3R)@)7(=*K))(@>~v~*@@3)dR)MP)X*)*:D0)U)F*Q)Q*Q)-)r((*9*+|(O*I*.){(Ac)Q*hK)25*3)_8))C(@)h(rZ){(G(8(C(F*>*M*5)?)<@(L6?*K*8)8P)b(9(K*>)5+Z)E(I*+)))N){(Am)W)8)n)k(Cp*F(EN)_)7r)~)|)](O)M8(7)3O@)y*-)R0)Qa*?(9)eD)7(C*@@e*+~*P(4)X(;***4)i)CF(ED,,)m(@)?*@*Q)K	377
60	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m, e,t,,, ,c,o,n,s,e,c,t,e,t, u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,115,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnDqxtz6it6icsTC/v20ML+mB7lGBPU+5fLMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbxYURSH/0bc9RSM/6J49qmnIkMcM934jCTFa+DW58CcjmG9/4V/ERZny4=	300	0((x)F1*)*8*E;)](7)lF.)])wv)?(5*-V*1))(=8(;]/R)iL)i*1r)Rev(*.5)z(K(J*H)ug)+(G)6A74]*<*A)+aT***G(H,*5e*Fn*H0P)t(:(F)|?(=p*8*7)>(3R)@)7(=*K))(@>~v~*@@3)dR)MP)X*)*:D0)U)F*Q)Q*Q)-)r((*9*+|(O*I*.){(Ac)Q*hK)25*3)_8))C(@)h(rZ){(G(8(C(F*>*M*5)?)<@(L6?*K*8)8P)b(9(K*>)5+Z)E(I*+)))N){(Am)W)8)n)k(Cp*F(EN)_)7r)~)|)](O)M8(7)3O@)y*-)R0)Qa*?(9)eD)7(C*@@e*+~*P(4)X(;***6))=(=*:)YJ(;uo*6=BR)Z*Dqg).*I)])`*8*JO(@)?D)))e(/C(	447

						-))YM)t)}=a(:N(6H)`){}j*7C9(5D*-)h*C(.)^z	
70	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m, e,t,,, ,c,o,n,s,e,c,t,e,t, u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t, u,r, ,e,g,e,}	76,111,114,101,109,32,105, 112,115,117,109,32,100,111 ,108,111,114,32,115,105,11 6,32,97,109,101,116,44,32,9 9,111,110,115,101,99,116,1 01,116,117,114,32,97,100,1 05,112,105,115,99,105,110, 103,32,101,108,105,116,46, 32,67,117,114,97,98,105,11 6,117,114,32,101,103,101,	110,105,18,342,138,1358,19,9 1,391,246,383,398,89,60,23,9, 533,2858,181,61,243,13043,44 9,155,273,40,130,1011,7,22,40 ,4,284,74,532,119,13,94,216,1 150,17,72,1,6,234,226,388,135 ,271,168,6699,101,40,230,289, 241,645,32,129,239,28,58,402, 92,90,76,407,264,18,77	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWt H+eMMuxuxVIbBGTpLQ+iymEHb4kFfC fvPGWrDq5L/2lov45dkzm+rYqhyvvNRD pLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlR meiOCIUelyfLYukbnDqxtz6it6icsTC/v20 ML+mB7lGBPU+5fLMV5WmAz+1u5Q8 7DJTP/W+tXvATZjoNTloxYe/YqS3NlBJu MmLwjN48dgVQZehfbxYURSH/0bc9RS M/6J49qmnIkMcM934jCTFa+DW58CcjmG wPCN6Llljxjo53k6aePsgVMnnol6iaX8ZoAQU7lJ8f4tGxmzn7uocdQ==	344	0((x)F1*)*8*E;)](7)lF.)])wv)?(5*-V*1))(=8(;]/R)iL)i*1r)Rev(*.5)z(K(J*H)ug)+(G)6A74]*<*A)+aT***G(H,*5e*Fn*H0P)t(:(F)|?(=p*8*7)>(3R)@)7(=*K))(@>~v~*@@3)dR)MP)X*)*:D0)U)F*Q)Q*Q)-)r((*9*+|(O*I*.){(Ac)Q*hK)25*3)_8))C(@)h(rZ){(G(8(C(F*>*M*5)?)<@(L6?*K*8)8P)b(9(K*>)5+Z)E(I*+)))N){(Am)W)8)n)k(Cp*F(EN)_)7r)~)|)](O)M8(7)3O@)y*-)R0)Qa*?(9)eD)7(C*@@e*+~*P(4)X(;***6))=(=*:)YJ(;uo*6=BR)Z*Dqg).*I)])`*8*JO(@)?D)))e(/C(-))YM)t)}=a(:N(6H)`){*F)Y)6(N*@*H(4)E)?)9)O*))tI.5*8trj)C1(1W)>)GE*,xU)T3(N)Q*8FKZ`^{}2*D*I(E\(O))())	507
80	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m, e,t,,, ,c,o,n,s,e,c,t,e,t, u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t, u,r, ,e,g,e,s,t,a,s, ,s,u ,s,c,i,}	76,111,114,101,109,32,105, 112,115,117,109,32,100,111 ,108,111,114,32,115,105,11 6,32,97,109,101,116,44,32,9 9,111,110,115,101,99,116,1 01,116,117,114,32,97,100,1 05,112,105,115,99,105,110, 103,32,101,108,105,116,46, 32,67,117,114,97,98,105,11 6,117,114,32,101,103,101,1 15,116,97,115,32,115,117,1 15,99,105,	110,105,18,342,138,1358,19,9 1,391,246,383,398,89,60,23,9, 533,2858,181,61,243,13043,44 9,155,273,40,130,1011,7,22,40 ,4,284,74,532,119,13,94,216,1 150,17,72,1,6,234,226,388,135 ,271,168,6699,101,40,230,289, 241,645,32,129,239,28,58,402, 92,90,76,407,264,18,77,262,50 ,216,21,212,28,38,34,229,250	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWt H+eMMuxuxVIbBGTpLQ+iymEHb4kFfC fvPGWrDq5L/2lov45dkzm+rYqhyvvNRD pLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlR meiOCIUelyfLYukbnDqxtz6it6icsTC/v20 ML+mB7lGBPU+5fLMV5WmAz+1u5Q8 7DJTP/W+tXvATZjoNTloxYe/YqS3NlBJu MmLwjN48dgVQZehfbxYURSH/0bc9RS M/6J49qmnIkMcM934jCTFa+DW58CcjmG wPCN6Llljxjo53k6aePsgVMnnol6iaX8ZoAQU7lJ8f4u3TMwDRO6HHikNGiOj+YB +fpwR2igDWkuRvm8OyI+ufSF7rPhkWg5U	384	0((x)F1*)*8*E;)](7)lF.)])wv)?(5*-V*1))(=8(;]/R)iL)i*1r)Rev(*.5)z(K(J*H)ug)+(G)6A74]*<*A)+aT***G(H,*5e*Fn*H0P)t(:(F)|?(=p*8*7)>(3R)@)7(=*K))(@>~v~*@@3)dR)MP)X*)*:D0)U)F*Q)Q*Q)-)r((*9*+|(O*I*.){(Ac)Q*hK)25*3)_8))C(@)h(rZ){(G(8(C(F*>*M*5)?)<@(L6?*K*8)8P)b(9(K*>)5+Z)E(I*+)))N){(Am)W)8)n)k(Cp*F(EN)_)7r)~)|)](O)M8(7)3O@)y*-)R0)Qa*?(9)eD)7(C*@@e*+~*P(4)X(;***6))=(=*:)YJ(;uo*6=BR)Z*Dqg).*I)])`*8*JO(@)?D)))e(/C(-))YM)t)}=a(:N(6H)`){*F)Y)6(N*@*H(4)E)?)9)O*))tI.5*8trj)C1(1W)>)GE*,xU)T3(N)Q*8FS(?n*5)6(<)R)2)s*)Va)q(+(D)R*1(:8*GX(0)`*K*5)8(?(3)/):*()=v**1*I5a(K)d*12)D)t*E(D	567
90	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m, e,t,,, ,c,o,n,s,e,c,t,e,t, u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t, u,r, ,e,g,e,s,t,a,s, ,s,u ,s,c,i,p,i,t, ,a,r,c,u,., ,}	76,111,114,101,109,32,105, 112,115,117,109,32,100,111 ,108,111,114,32,115,105,11 6,32,97,109,101,116,44,32,9 9,111,110,115,101,99,116,1 01,116,117,114,32,97,100,1 05,112,105,115,99,105,110, 103,32,101,108,105,116,46, 32,67,117,114,97,98,105,11 6,117,114,32,101,103,101,1 15,116,97,115,32,115,117,1 15,99,105,112,105,116,32,9 7,114,99,117,46,	110,105,18,342,138,1358,19,9 1,391,246,383,398,89,60,23,9, 533,2858,181,61,243,13043,44 9,155,273,40,130,1011,7,22,40 ,4,284,74,532,119,13,94,216,1 150,17,72,1,6,234,226,388,135 ,271,168,6699,101,40,230,289, 241,645,32,129,239,28,58,402, 92,90,76,407,264,18,77,262,50 ,216,21,212,28,38,34,229,250, 131,37,3,171,473,227,67,573,2 81	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWt H+eMMuxuxVIbBGTpLQ+iymEHb4kFfC fvPGWrDq5L/2lov45dkzm+rYqhyvvNRD pLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlR meiOCIUelyfLYukbnDqxtz6it6icsTC/v20 ML+mB7lGBPU+5fLMV5WmAz+1u5Q8 7DJTP/W+tXvATZjoNTloxYe/YqS3NlBJu MmLwjN48dgVQZehfbxYURSH/0bc9RS M/6J49qmnIkMcM934jCTFa+DW58CcjmG wPCN6Llljxjo53k6aePsgVMnnol6iaX8ZoAQU7lJ8f4u3TMwDRO6HHikNGiOj+YB +fpwR2igDWkuRvm8OyI+ufSsJEcJ2cQQ UI7laiFq6L+LtUkCeu4mEAB6izajU7VUo	428	0((x)F1*)*8*E;)](7)lF.)])wv)?(5*-V*1))(=8(;]/R)iL)i*1r)Rev(*.5)z(K(J*H)ug)+(G)6A74]*<*A)+aT***G(H,*5e*Fn*H0P)t(:(F)|?(=p*8*7)>(3R)@)7(=*K))(@>~v~*@@3)dR)MP)X*)*:D0)U)F*Q)Q*Q)-)r((*9*+|(O*I*.){(Ac)Q*hK)25*3)_8))C(@)h(rZ){(G(8(C(F*>*M*5)?)<@(L6?*K*8)8P)b(9(K*>)5+Z)E(I*+)))N){(Am)W)8)n)k(Cp*F(EN)_)7r)~)|)](O)M8(7)3O@)y*-)R0)Qa*?(9)eD)7(C*@@e*+~*P(4)X(;***6))=(=*:)YJ(;uo*6=BR)Z*Dqg).*I)])`*8*JO(@)?D)))e(/C(-))YM)t)}=a(:N(6H)`){*F)Y)6(N*@*H(4)E)?)9)O*	625

				JhEKWEgqT/k=		))tI.5*8trj)C1(1W)>)GE*,xU)T3(N)Q*8FS(?n*5)6(<)R)2)s*)Va)q(+(D)R*1(:8*GX(0)`*K*5)8(?(3)/):*()=v**1*I)jF)V()}aMR)`*MB)pB3)z(H)Z`)>Y)V*@)o)w|)y)i)J6)E)|HV*3(B*A)UHA)gp1)Q)ud)K)i	
100	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m, e,t,,, ,c,o,n,s,e,c,t,e,t, u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t, u,r, ,e,g,e,s,t,a,s, ,s,u ,s,c,i,p,i,t, ,a,r,c,u,., , C,u,m, ,s,o,c,i,i,s,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,32,115,117,115,99,105,112,105,116,32,97,114,99,117,46,32,67,117,109,32,115,111,99,105,105,115,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281,500,284,87,13,75,85,68,86,55,306,33	00lJ4z3Tikrc5GCYNG98bGgeICohe07pWtH+eMMuxuxVIbBGTpLQ+iymEHb4kFfCfvPGWrDq5L/2lov45dkzm+rYqhyvvNRDpLbczI1yqIJyZwIsBeKyDldSIN9RKUbvlRmeiOCIUelyfLYukbnDqxtz6it6icsTC/v20ML+mB7lGBPU+5fLMV5WmAz+1u5Q87DJTP/W+tXvATZjoNTloxYe/YqS3NlBJuMmLwjN48dgVQZehfbxYURSH/0bc9RSM/6J49qmnIkMcM934jCTFa+DW58CcjmGwPCN6Lllxjo53k6aePsgVMnnol6iaX8ZoAQU7lJ8f4u3TMwDRO6HHikNGiOj+YB+fpwR2igDWkuRvm8OyI+ufSsJEcJ2cQQUI7laiFq6L+LtUkCeu4mEAB6izajU7VUoBOzvYuLLYo5InN82c3XoPAAXcc2KzxlkROKxMm1aAHbH821TVbudaz5kal2wqSkh	480	0((x)F1*)*8*E;)](7)lF.)])wv)?(5*-V*1))(=8(;]/R)iL)i*1r)Rev(*.5)z(K(J*H)ug)+(G)6A74]*<*A)+aT***G(H,*5e*Fn*H0P)t(:(F)|?(=p*8*7)>(3R)@)7(=*K))(@>~v~*@@3)dR)MP)X*)*:D0)U)F*Q)Q*Q)-)r((*9*+|(O*I*.){(Ac)Q*hK)25*3)_8))C(@)h(rZ){(G(8(C(F*>*M*5)?)<@(L6?*K*8)8P)b(9(K*>)5+Z)E(I*+)))N){(Am)W)8)n)k(Cp*F(EN)_)7r)~)|)](O)M8(7)3O@)y*-)R0)Qa*?(9)eD)7(C*@@e*+~*P(4)X(;***6))=(=*:)YJ(;uo*6=BR)Z*Dqg).*I)])`*8*JO(@)?D)))e(/C(-))YM)t)}=a(:N(6H)`){*F)Y)6(N*@*H(4)E)?)9)O*))tI.5*8trj)C1(1W)>)GE*,xU)T3(N)Q*8FS(?n*5)6(<)R)2)s*)Va)q(+(D)R*1(:8*GX(0)`*K*5)8(?(3)/):*()=v**1*I)jF)V()}aMR)`*MB)pB3)z(H)Z`)>Y)V*@)o)w|)y)i)J6)E)|HV*3(B(O(Eb(=(I(J)M(N)}*E2()95K*.)0*/Q)A)`TP)<Uv@)6*E(.q(H(K)U3*9f*+]):)quV)W(6F*2)^)S)SDB)^@	693

EK-6. AES ŞİFRELEME – ARİTMETİK KODLAMA

	S	*A*	*C*	*AES*	*x*	*ARİTMETİK KODLAMA*	*x'*
10	{L,o,r,e,m, ,i,p,s,u,}	76,111,114,101,109,32,105,112,115,117,	110,105,18,342,138,1358,19,91,391,246	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD847377713479A9E4E67EB7009274D784E0	96	5*N)wD*D*3b*G*O(6*@;(M)A*0)(@)c)O)<)A0y*+*9(@(7)Z*Ol)K2)**+Ub*CK)q*4*9*2K/(C*J)[^(:)2)9*P)9)3)U*>**)F)kK*(A)M`)Z)1*I)8)>(K}\(2)w)z(()a)p4	138
20	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAA67CF4718EC002F080F051FDC656E0DD	160	5*N)wD*D*3b*G*O(6*@;(M)A*0)(@)c)O)<)A0y*+*9(@(7)Z*Ol)K2)**+Ub*CK)q*4*9*2K/(C*J)[^(:)2)9*P)9)3)W)c*7(?*1):39q(:)}*+(M-)?*3)D)Ib.M(*M)SfL)d)m*5),TU*I(7*7(.{Hzf)WA(A{*.)k)M);kw(A)*/(L)kk)^*GB)E-=)[*Q@(g*+A(8	205
30	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D540611607DA67B4A3573A22CCDA2D2868AF6F11	224	5*N)wD*D*3b*G*O(6*@;(M)A*0)(@)c)O)<)A0y*+*9(@(7)Z*Ol)K2)**+Ub*CK)q*4*9*2K/(C*J)[^(:)2)9*P)9)3)W)c*7(?*1):39q(:)}*+(M-)?*3)D)Ib.M(*M)SfL)d)m*5),TU*I(7*7(.{Hzf)WA(A{*.)k)M)h(BY)Z)b(D(5*)k*M(.OgJ)/1*O`=w*Ie)A()j2)K*2OMAT)e)00)\1())b)X*A),)SL(*S*5)(*E))(N*F(((>);)9(H*8`*)*J(9)yD	276
40	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D54061165523FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81CCF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FFE1BFD6152DB3B0657E3BFC644CF6211D	320	5*N)wD*D*3b*G*O(6*@;(M)A*0)(@)c)O)<)A0y*+*9(@(7)Z*Ol)K2)**+Ub*CK)q*4*9*2K/(C*J)[^(:)2)9*P)9)3)W)c*7(?*1):39q(:)}*+(M-)?*3)D)Ib.M(*M)SfL)d)m*5),TU*I(7*7(.{Hzf)WA(A{*.)k)M)h(BY)Z)b(D(5*)k*M(.OgJ)/1*O`=w*Ie)A()j2)K*2OMAT)e)00)\1())b)X*A),)c):])mQ4)H*+))(G)i(/*@1*<t(,)|)co*O()*,(D)4O*J.R(1(B)BF*()v*L*3@](?on(/>)DtEIq*G)nO)/)(G*:)0C*8)2)C*:V)P(<):)`o(>l(K4*An)j(4(O>U/0	368
50	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D54061165523FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81CCF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF227637A1EB08514946293740DD1A370D93304FD1509E26136FDF877DE7015D2AE428D7C54830C8875AFCF716F7A5507A	384	5*N)wD*D*3b*G*O(6*@;(M)A*0)(@)c)O)<)A0y*+*9(@(7)Z*Ol)K2)**+Ub*CK)q*4*9*2K/(C*J)[^(:)2)9*P)9)3)W)c*7(?*1):39q(:)}*+(M-)?*3)D)Ib.M(*M)SfL)d)m*5),TU*I(7*7(.{Hzf)WA(A{*.)k)M)h(BY)Z)b(D(5*)k*M(.OgJ)/1*O`=w*Ie)A()j2)K*2OMAT)e)00)\1())b)X*A),)c):])mQ4)H*+))(G)i(/*@1*<t(,)|)co*O()*,(D)4O*J.R(1(B)BF*()v*L*3@](?on(/>)DtEIq*G)nO)/)(G*:)0C*8)2(C)c**)4)y=)A)e*I~*JV))(A*>):*6)l?`)h*4(D)@-)L))I)*e)W*I;(1(C*I((H(5)6(*)T*>)vA()r.)i5p{)w)y)81(EO3	435
60	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D54061165523FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81CCF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF227637A1EB08514946293740DD1A370D93304FD1509E26136FDF877DE7015D2A6B68F0F34E84DE3CD83AE158EE0099A317EA8162EB498077EDDBCFCE1040F623CD4EA804C762F39A1E58CA3D61519285	448	5*N)wD*D*3b*G*O(6*@;(M)A*0)(@)c)O)<)A0y*+*9(@(7)Z*Ol)K2)**+Ub*CK)q*4*9*2K/(C*J)[^(:)2)9*P)9)3)W)c*7(?*1):39q(:)}*+(M-)?*3)D)Ib.M(*M)SfL)d)m*5),TU*I(7*7(.{Hzf)WA(A{*.)k)M)h(BY)Z)b(D(5*)k*M(.OgJ)/1*O`=w*Ie)A()j2)K*2OMAT)e)00)\1())b)X*A),)c):])mQ4)H*+))(G)i(/*@1*<t(,)|)co*O()*,(D)4O*J.R(1(B)BF*()v*L*3@](?on(/>)DtEIq*G)nO)/)(G*:)0C*8)2(C)c**)4)y=)A)e*I~*JV))(A*>):*6)l?`)h*4(D)@-)L))I)*e)W*I;(1(C*I((H(4(KJ),*,*?(7(1)0*/)wr)mq*,*-)D(8)y0*N)=~q)n)Q7-4()*G*@)}(O(7O[*7*=]e*:Zh)~*1k)A)m)s])M(C(,)r*)k()	500
70	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D54061165523FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81CCF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF227637A1EB08514946293740DD1A370D93304FD1509E26136FDF877DE7015D2A6B68F0F34E84DE3CD83AE158EE0099A317EA8162EB498077EDDBCFCE1040F623346F8CA6CC79AB2F98EE9FD166E0A9AE71FA33B5D685CF9341BF7AD4F7F829EF9455387865D631C2	512	5*N)wD*D*3b*G*O(6*@;(M)A*0)(@)c)O)<)A0y*+*9(@(7)Z*Ol)K2)**+Ub*CK)q*4*9*2K/(C*J)[^(:)2)9*P)9)3)W)c*7(?*1):39q(:)}*+(M-)?*3)D)Ib.M(*M)SfL)d)m*5),TU*I(7*7(.{Hzf)WA(A{*.)k)M)h(BY)Z)b(D(5*)k*M(.OgJ)/1*O`=w*Ie)A()j2)K*2OMAT)e)00)\1())b)X*A),)c):])mQ4)H*+))(G)i(/*@1*<t(,)|)co*O()*,(D)4O*J.R(1(B)BF*()v*L*3@](?on(/>)DtEIq*G)nO)/)(G*:)0C*8)2(C)c**)4)y=)A)e*I~*JV))(A*>):*6)l?`)h*4(D)@-)L))I)*e)W*I;(1(C*I((H(4(KJ),*,*?(7(1)0*/)wr)mq*,*-)D(8)y0*N)=~q)n)Q7-	557

		103,101,		12F96789B190711E		4()*G*@)}(O(7O[*7)a*M*7X)z)R0d(F)/)U)BH)p(HHT)~(HT*F*E*8*+*9~<`)-(?)-8;(OY2)A(7)w*)+)a(H(CK1wnX)7*PQ3*>g*4)Y	
80	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D54061165523FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81CCF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF227637A1EB08514946293740DD1A370D93304FD1509E26136FDF877DE7015D2A6B68F0F34E84DE3CD83AE158EE0099A317EA8162EB498077EDDBCFCE1040F623346F8CA6CC79AB2F98EE9FD166E0A9AE71FA33B5D685CF9341BF7AD4F7F829EF6117928A46399B94ADBAD3AE4F6D8E7F1C41E842E543FBF46088B628C59553A06EC6E4355E486E1799F26BB4807C0403	576	5*N)wD*D*3b*G*O(6*@;(M)A*0)(@)c)O)<)A0y*+*9(@(7)Z*Ol)K2)**+Ub*CK)q*4*9*2K/(C*J)[^(:)2)9*P)9)3)W)c*7(?*1):39q(:)}*+(M-)?*3)D)Ib.M(*M)SfL)d)m*5),TU*I(7*7(.{Hzf)WA(A{*.)k)M)h(BY)Z)b(D(5*)k*M(.OgJ)/1*O`=w*Ie)A()j2)K*2OMAT)e)00)\1())b)X*A),)c):])mQ4)H*+))(G)i(/*@1*<t(,)|)co*O()*,(D)4O*J.R(1(B)BF*()v*L*3@](?on(/)>)DtEIq*G)nO)/)(G*:)0C*8)2(C)c**)4)y=)A)e*I~*JV))(A*>):*6)l?`)h*4(D)@-)L))l)*e)W*I;(1(C*I((H(4(KJ),*,*?(7(1)0*/)wr)mq*,*-)D(8)y0*N)=~q)n)Q7-4()*G*@)}(O(7O[*7)a*M*7X)z)R0d(F)/)U)BH)p(HHT)~(HT*F*E*8*+*9~<`)-(?)-8;(OY2)@)v<*M)**Fw(K*)0(-)f)D*?)y)h)jytR^*F)G]9-])V)p(I*Ku)1)a*G=)r)W)y)/*2*9*=R)Q2)/(M*8)T*+)q):])\J))	621
90	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,p,i,t, ,a,r,c,u,., ,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,112,105,116,32,97,114,99,117,46,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D54061165523FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81CCF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF227637A1EB08514946293740DD1A370D93304FD1509E26136FDF877DE7015D2A6B68F0F34E84DE3CD83AE158EE0099A317EA8162EB498077EDDBCFCE1040F623346F8CA6CC79AB2F98EE9FD166E0A9AE71FA33B5D685CF9341BF7AD4F7F829EF6117928A46399B94ADBAD3AE4F6D8E7F1C41E842E543FBF46088B628C59553A0374432D1EEA49A001DB68690F928805B3CBFC45487FBF836E000DE2B58D6B12BAB354C0A81E634A08BB43A55EF633BB0	640	5*N)wD*D*3b*G*O(6*@;(M)A*0)(@)c)O)<)A0y*+*9(@(7)Z*Ol)K2)**+Ub*CK)q*4*9*2K/(C*J)[^(:)2)9*P)9)3)W)c*7(?*1):39q(:)}*+(M-)?*3)D)Ib.M(*M)SfL)d)m*5),TU*I(7*7(.{Hzf)WA(A{*.)k)M)h(BY)Z)b(D(5*)k*M(.OgJ)/1*O`=w*Ie)A()j2)K*2OMAT)e)00)\1())b)X*A),)c):])mQ4)H*+))(G)i(/*@1*<t(,)|)co*O()*,(D)4O*J.R(1(B)BF*()v*L*3@](?on(/)>)DtEIq*G)nO)/)(G*:)0C*8)2(C)c**)4)y=)A)e*I~*JV))(A*>):*6)l?`)h*4(D)@-)L))l)*e)W*I;(1(C*I((H(4(KJ),*,*?(7(1)0*/)wr)mq*,*-)D(8)y0*N)=~q)n)Q7-4()*G*@)}(O(7O[*7)a*M*7X)z)R0d(F)/)U)BH)p(HHT)~(HT*F*E*8*+*9~<`)-(?)-8;(OY2)@)v<*M)**Fw(K*)0(-)f)D*?)y)h)jytR^*F)G]9-])V)p(I*Ku)1)a*G=)r)M)r+(/,(.)\)v7M)Y0*@I5)Gl)IC)m_{*Ni)f)u(/3)o)v>C-*7*A)w(C*-9)2{):iD)d)G(+*L(?)X)f)c)7)N*F))	677
100	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,p,i,t, ,a,r,c,u,., ,C,u,m, ,s,o,c,i,i,s,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,112,105,116,32,97,114,99,117,46,32,67,117,109,32,115,111,99,105,105,115,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281,500,284,87,13,75,85,68,86,55,306,33	6319E06677E05353975E6CBC61EF496F8AF5CFDF75B5E202962B38AABF3BDD84EFFB94944F40C8B6C3B06C89335F4625101B9BA429C1FF789473D44DE83E1ACAB858D7B421BA01DB34A0B17028933C68678989763D6FFCA6C4D2D808D54061165523FE88275F0EE4BCEB8250D89642432A0DF3D361EFA81CCF7ED9EF61B7A21635D30FEAF594C9E66C85FB2F3EF848FF227637A1EB08514946293740DD1A370D93304FD1509E26136FDF877DE7015D2A6B68F0F34E84DE3CD83AE158EE0099A317EA8162EB498077EDDBCFCE1040F623346F8CA6CC79AB2F98EE9FD166E0A9AE71FA33B5D685CF9341BF7AD4F7F829EF6117928A46399B94ADBAD3AE4F6D8E7F1C41E842E543FBF46088B628C59553A0374432D1EEA49A001DB68690F928805B3CBFC45487FBF836E000DE2B58D6B12B140677D864C831E32CDC765289484F57FD5C1E01EE145553CEFC279D36533AE62220A077C0C3C1593BD9C2A12E5D9834185168760CA25FAB0C8C2AE032A400FB	736	5*N)wD*D*3b*G*O(6*@;(M)A*0)(@)c)O)<)A0y*+*9(@(7)Z*Ol)K2)**+Ub*CK)q*4*9*2K/(C*J)[^(:)2)9*P)9)3)W)c*7(?*1):39q(:)}*+(M-)?*3)D)Ib.M(*M)SfL)d)m*5),TU*I(7*7(.{Hzf)WA(A{*.)k)M)h(BY)Z)b(D(5*)k*M(.OgJ)/1*O`=w*Ie)A()j2)K*2OMAT)e)00)\1())b)X*A),)c):])mQ4)H*+))(G)i(/*@1*<t(,)|)co*O()*,(D)4O*J.R(1(B)BF*()v*L*3@](?on(/)>)DtEIq*G)nO)/)(G*:)0C*8)2(C)c**)4)y=)A)e*I~*JV))(A*>):*6)l?`)h*4(D)@-)L))l)*e)W*I;(1(C*I((H(4(KJ),*,*?(7(1)0*/)wr)mq*,*-)D(8)y0*N)=~q)n)Q7-4()*G*@)}(O(7O[*7)a*M*7X)z)R0d(F)/)U)BH)p(HHT)~(HT*F*E*8*+*9~<`)-(?)-8;(OY2)@)v<*M)**Fw(K*)0(-)f)D*?)y)h)jytR^*F)G]9-])V)p(I*Ku)1)a*G=)r)M)r+(/,(.)\)v7M)Y0*@I5)Gl)IC)m_{*Ni)f)u(/3)o)v>C-*7*A)v*L)MY.*0(8)c{)A)7)8(1-(5{tX)Bs{(D)v7)6***+)4)j(J)H(B*,)E)9)_[(>a)}1(8(1*:(L?((s*Gu(<W*MS)^)D)F**(G)E)F{v*A)3(5h(-P(G)R)_oc	763

EK-7. BLOWFISH ŞİFRELEME – ARİTMETİK KODLAMA

	S	*A*	*C*	*BLOWFISH*	*x*	*ARİTMETİK KODLAMA*	*x'*
10	{L,o,r,e,m , ,i,p,s,u,}	76,111,114,101,109, 32,105,112,115,117,	110,105,18,342,138,13 58,19,91,391,246	18a0d238c0f940049c1326e0f0e7be2f03751b9f7983e88e1 27fac64dff21a02f86d62fab33d217257a1c4b1a06975ba	96	1(/*:@*I)55sA)W(.T(/k(M)v)oQ(E?())2)nkA**);*A)F)C)m(8y)`f _)lT7):),)L}(L/H*(;J*))4);)r(N(;I)k)g(:(7)-(C(3}(2`)|*G(F)g*L)JM)8*,])4)<k))	134
20	{L,o,r,e,m , ,i,p,s,u, m, ,d,o,l,o ,r, ,s,i,}	76,111,114,101,109, 32,105,112,115,117, 109,32,100,111,108, 111,114,32,115,105,	110,105,18,342,138,13 58,19,91,391,246,383,3 98,89,60,23,9,533,2858 ,181,61	1228bca0aa0319864d6547880d47671b440d2c6b31de6089 5ef0496c3868376b4994723fa77e0584f8abd8f426f6a9c83 308ca4f6c6918bd1bcd6eb5f1d436a39fea6f26ffbfa1bd76e 0607723027f7d	160	1())tve)vy).)u*D)M(K(,V)V:(@*I+*K(6Q])TA*N*7(,:`);(M)9)B(?W*.)f(O9*:)t*?(.Ph(7])0)Ki)A)M(0*B(5)iS_5*I*,)?M)Y)r(+8)H*9)e)^)Se)-)fp)@M)Z):2)o*@)3)]y)/6P(8)q*8)H*6*-(5(=)])M(A(A)M~B)g(N)}(9)l)A*:)G)d(HC(+)D)im*()rp){	213
30	{L,o,r,e,m , ,i,p,s,u, m, ,d,o,l,o ,r, ,s,i,t, ,a ,m,e,t,,, ,c ,o,}	76,111,114,101,109, 32,105,112,115,117, 109,32,100,111,108, 111,114,32,115,105, 116,32,97,109,101,1 16,44,32,99,111,	110,105,18,342,138,13 58,19,91,391,246,383,3 98,89,60,23,9,533,2858 ,181,61,243,13043,449, 155,273,40,130,1011,7, 22	02631ec8751ddf20c6086a743f3cf263c6728d07fb7b973fd ed73b4e41c7a5ac4597756e87c0a098fad3c9569503684be4 0b58f2324167f2f20aec2f9697e2eef1220e18b4c7e2d6bf8e c7655c257b1a0af6458b7bda45d3618b9663b9aadda43903 94f7c09e7a154f10062ee419b042104970b4ed13a24b	240	0(*)x(B)q>L5s(F).e*/Q1)c*<Z)F-I*E(3(7z-)7ktk):)2*M(>(D)]*<|(J*(+[()[R|)@0nE^*1*On*8=)E/E)o)tm*IU *(9(@VUM3l)w):(:)Q(D)\Ve*8))(60)2)z(O).)5*E*M(O*D)A(5*O *3vA(G)=(9Z<:)W)33*P)D*,)Nzy)u)jW)Q*C?)_RiSU)r)p)@*+(B)h)R+*D)L)@p(@(;E=*@n})rZ5(4})X))(),oihv98*E9;~(+P{*;)5)H(:pq)})F)+).))	275
40	{L,o,r,e,m , ,i,p,s,u, m, ,d,o,l,o ,r, ,s,i,t, ,a ,m,e,t,,, ,c ,o,n,s,e,c, t,e,t,u,r,}	76,111,114,101,109, 32,105,112,115,117, 109,32,100,111,108, 111,114,32,115,105, 116,32,97,109,101,1 16,44,32,99,111,110, 115,101,99,116,101, 116,117,114,32,	110,105,18,342,138,13 58,19,91,391,246,383,3 98,89,60,23,9,533,2858 ,181,61,243,13043,449, 155,273,40,130,1011,7, 22,40,4,284,74,532,119 ,13,94,216,1150	656d6c2d065f1c109ebc71870762d0e96e1e24cb02d0ed6b 87e9599bc6b54e48a8a12cc4365d6eabbf2c13b73ff3b948fa 33c1eabfa8c3ecae470f37b8bf9a5f7234b7b290b69fd08ea5 4b639d45049f3a15a42f9440acf489631568f078213c79a9a f5f77abaa07ccee1ccb451cc2a9cd8b4394f6a23e6d8b088e65 ebb6c63536d704cd9dcf08a332b23c982cdddc2dc4c705e6 43f4cb8e902cee7b37fee5cd	320	5*P)w)d*PN*H);Ewg);)~*C)?3(2w)L)D)M*Lt)dN*.*()Z)}*8*3)_K)C)h)j)-(E(H(@)r)u6(KcD()(*(5)X(?)/)kp)H*OV*.(Nlr)h)Z(BP)/(2*(l)yex*?(@o)Q*C(*L)4)v)9)]y*?)=(I*E),)gGw)b(<(I)/f){(:f4(+E*6(G)>Odoe@*P(Ln*<AaD)1)V(6L*2o)t(2E)<)qvTw*NK)):);fq*HG+y)U]*L<U*J)W)`l)S)v;*+73(67(M(;N*>)A)c)O*,)G*GFT,)3)n)i*7)\)|))c(1)zg<)O)x)H*Ge)G).)-)i)=)3(B*=(46,(C(**)0?/d)y(>(G)0)`)QW(E*(**B())^)1)y	370
50	{L,o,r,e,m , ,i,p,s,u, m, ,d,o,l,o ,r, ,s,i,t, ,a ,m,e,t,,, ,c ,o,n,s,e,c, t,e,t,u,r, ,a ,d,i,p,i,s,c, i,n,g,}	76,111,114,101,109, 32,105,112,115,117, 109,32,100,111,108, 111,114,32,115,105, 116,32,97,109,101,1 16,44,32,99,111,110, 115,101,99,116,101, 116,117,114,32,97,1 00,105,112,105,115, 99,105,110,103,	110,105,18,342,138,13 58,19,91,391,246,383,3 98,89,60,23,9,533,2858 ,181,61,243,13043,449, 155,273,40,130,1011,7, 22,40,4,284,74,532,119 ,13,94,216,1150,17,72, 1,6,234,226,388,135,27 1,168	a8bb615f393693fdba30e5209f13b9c962b79acbfd4d8d419 4314586414bd7a22e2072aabc85d823ee61e5405fdfd6f256 244b21963c86c5aac2cba85f39b39c6cc44051f6672783694 5998b55874adfce4fdbbd3bf15470ea7a6f95fba32a3b83de0 60f2c75a83664a9866c56cfa931565b5e34a82f818f8b8f1ab 70d4ca006fdcf4602ff44d48770d703bdc91d536441f19652 438b930fd7daa1a12bfbff753889fd389051ef5d61f2dd33fd 422b8d20e3efb71881c40716dfa0bdfe9c2931	384	`*)(D6W)l)i*A]p)F*N)<)\\(F)g)mQl(C)d)8)4)XX)G)9Sp(/)4)s(,9*7G(/(Do1P2*Nm)w|)^*/**U**)X)W*/(>*K`){):**/R)V)cM)i*H(>)8/*/*={)C(O*+)i(.)l}(;[*.)H)[)u)_m*y=)Q)n)1*2*5C)q)+)5r*8*-)O)t)xr)2)j(AqU(4)TG*L(M(?:)s*O)+q)p*6)e)n(H)1)I)+f)ns(O(NKQ)g)Mf=3eM*C)]*7)\W.*0)=(L8Y*0ko)Q)Z*O6*0)k*)bSH)g*Q).)a(Cxv)B*Mx)2)o)L)x),((V)^*0)R)J)y)Y)1(*1{(G>vj|H)AG)Ji-V)1)S(-*()d)u)TvC(G)u)w*K*G(+A)G(-)g)~)r:)V)T(O)4*)k)l|.S(L-B)j)kCC*L1(@9(5*0)+?=*?*Qd)[])	433
60	{L,o,r,e,m , ,i,p,s,u, m, ,d,o,l,o ,r, ,s,i,t, ,a ,m,e,t,,, ,c ,o,n,s,e,c, t,e,t,u,r, ,a ,d,i,p,i,s,c, i,n,g, ,e,l,i ,t,., ,C,u,r, }	76,111,114,101,109, 32,105,112,115,117, 109,32,100,111,108, 111,114,32,115,105, 116,32,97,109,101,1 16,44,32,99,111,110, 115,101,99,116,101, 116,117,114,32,97,1 00,105,112,105,115, 99,105,110,103,32,1 01,108,105,116,46,3 2,67,117,114,	110,105,18,342,138,13 58,19,91,391,246,383,3 98,89,60,23,9,533,2858 ,181,61,243,13043,449, 155,273,40,130,1011,7, 22,40,4,284,74,532,119 ,13,94,216,1150,17,72, 1,6,234,226,388,135,27 1,168,6699,101,40,230, 289,241,645,32,129,23 9	dfd9c6f8a43308c96e642adc288e5120d791b0ff1481318dc 2b0e0aa02d3ef0f042b40d6f4ae3b0fb5db211671108a9081 557df1df92860c6c9a57e0f728fca37a148a0d1da743c719d 47ac03ef8cf9778d2cbd3bcd5ed9acd06e4d1cbf363df209dc 8c2157463ce657a7eb210f962df062d81b5a2162c183dc378 db0a112d0c2ec3fb08d302b1c45a4ec4bf331a1d5fb745996 b4deec64dc0a703ba6ec687ea840a4dce34ae988a4aca8589 4c4bcec29f38d7b35c0c9dac5a2490b6e27ef3d2b177fb88a 3761a9a99eed0c5dbe5d62e7b1ccac1d37fb0c1baada332e3 d97e92b1984c279baec04a	464	d(*)<)0m)`*J)j(G(N*.);)@M*@1i6(+0*LH)2)r(7)X)9(3a)iE*=)u*?*E<(I*M*I)^K)pf(?(@*K)C(()TCa(1BP)>*(7)JSd(0I)()I{)t7}})Z)-((Q)}(2{(5)_7m(6(:),)e8t):(A=)M*7)y)q*<k)y(5)z(L)m)9g(()y)M),W{)Z(2)2~*)(C-w)B);(+n(GG)])NHE*9)@c)?)q*D0).)0)(a);b:(K(,)*Gc)D(.){)H)g(H);(:)Q)7)f)_(B*>)I)_):*0)P?((2(=1)V)w)P)Yq)b)F*)*?)lI)p)W()*>d(K(.)v<)h)`O)/*+)`)8[)Lma)}x)r(C)6c(K(E[b)G(6`(,R)i(E)2*Mv)Q(@P)g*IV0p)e(((1)n)a(@)ZW9)9V)x*D*I*8*;sT)_=)h)>wc)f:)E*L*9))*1){(@Jk)0F)R^)z)J)E,*-)d)y)|:)q[)27(I),)0}*4*B*?*;)~*G(G*E*BA)x)l(D)?(N*1)i)@Wi*Q6*:	521

70	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77	5c4cc7627a722b91272623894b8dec1c949ed27249496799f55a2bfcfe3df8d4629f1b96c347a886bab1e16013a815bfe8406ed809e6236915b52f539bb6ca205185d1b9e28df2f31c73e2be9535b762668b7b12f71c512bff6d5f13bade77d7d935fab13d9b0b33ae3948518c0543bb9f451fb0185790505 1b98ce1aea3eeeb64c676e1173fb2111d2948050dfb80183a6fab0d59ef26448ef416582448e93bef820ab811699b04a0733ba7778bfe1d532c22dd86591331addee64ad63c18db3529ee8b613692c76f60012e0459782ba8d3a7ed12b68795ba076b15e6f01c7e32ff64b41a8f22e2f2ce0a60ba4bae43a88d52ab0650719b2c52101aae8da1610f2e9b823f386391	528	5.>)YM)[o)xJe*=)|)m)Q)~(4)ZJH*J*B`)L(:)9>z)9)@d*EK_=?8*Cl*()<)I*Qh(F)u*4(C)y)v(?R*F)k,)0\)Q*4)R(*);)ec(I)3*>(@6(7*1)H*(*J)f(*)f)Mi*C`*<)U(Bic*/n*Q)l)k)p*9Qr(()B)P(,)b**m)d)M,*:(7*L|8)p*7*M)T)jr)sn[_e)<[:Hb((*6(3yF)(N*-)AP0?*F>*C+)^\)s)C)N)L(()D)M0C9*.)0)D(,)+u)k)s)q(-S*+)g)Q2(2)O)B(K).(2D(D)S(E*>(2(9c6)5Rk(Lp9*1-{T,(9F(5(4)Wj2>).)jim*(YG)No*9)k?*M)f*Pa*/)w~z](@*H(LE;*98(@)H),`8x~(4)k)3)C(,*D*?G)8B)?)]3)O)J)E(;(5w)I)p*6)Q)-)V*)(@*0aOx^LN*7*9)U(I(F(O(1)=W(Etw)_)t)0)]Oa)?Q*C*GKV(J)(=)5)a(((;@)FUj()*8)gC(-k)*(N*C`:H)s*B)-f(2)6L)VUT)O)F)K*.(,(1(/*,)S-)M)W){)1z	561
80	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t, a,s, ,s,u,s ,c,i,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,46,32,115,117,115,112,101,110,100,105,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250	ae65d2061dbef5d348f24ba5001b00420dc045925696ca9ad2f39d59ffe857415bc2a4c2ba3d9e6f948cc46f446895649a4da341c9fb5ac99cce2e900a71baca82cf9c9416d877c572d20ab9a766b008a59f2f4c2346f46797ddc0e0eb456d30e6801a47a5d020c0ced291c487ec64b4f73fece5b4c3083008db599e3badd8825b16b1f27766fddf4ffb51a014d60f3e67ac5ede4d2fbfaad3d153c10e419975fb49ba6116007ac4b05f6275582b168e78cab7c5b60c8f29178f64ae0b77402726c6aadc8864da8b0f0b071e73f4e355df7f3e18875397563e442e1367e85f80abc85d10a68b1ed21d85c228f9170ef9aed57c35b4f9aae4b4057a27d04e6014e43f97a49bea27a2aa790d5be7bad9bd25f9c19c8e3982cd2f86d355de8e302fb9bbdcb5c4af5c5b	592	a(,fX(*)V(3*)*))u)d(A)+(,_N*Jg(4(;)y7(.)\)).>(AD)d)4)))t)F=*A)Pm(M)d(:*Eb(K)X_()))m*P*M)n)9s)I*.)Lh)G(,)w)c)~(A)8*@XM)*JTZ)+iE)1*?)N*L(/.)qJ(Mu*;(>,)CbV(<)p*-7)G[(I)}**/(2*-\)U(J)q*JuF)~*<)7AX8*>)V*,)k*6(C(J()L(L)i>)qy(<(N)\4(?(1))^*>=)X*/*4)Z5*?h)Rd_X)[(7o*Ed)S*02)v{)x)S9)a~):(I*;){*0(;*K)u*<6(=*I)}_*0)h(O)1)U)x7)8s)4)m(-(J)T)G(G)_)B(4)Z)v*/)1e*6)M@*4q4y(4-)1u)h)^Tl(/GHL5w)Vf)`0)2)mU){u){)u(2)rn)-H)h(H(((,(:)y)u+**D|*.*5*7)D)D){pgA)<)I)c/,z1h)Q(**H*5*I\(N)E(G*H*Ds)r36*4)E)J*B*8*;*.Q*3Jx.m(;X.)7d)4*-)@)v(MC)x*I(A*H*:);])>):(D)v)P*7(?(()I),)9O((_)>)V)w*?*L(G*3(2g`)e()x*4*O):i)7G)k@)|(FI<g(@*3)U(G)B)A)j*H9[)W(=H(:)F(M*4()i)c(=(2),)m}*O)i	644
90	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t, a,s, ,s,u,s ,c,i,p,i,t, , a,r,c,u,., ,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,46,32,115,117,115,112,101,110,100,105,112,105,116,32,97,114,99,117,46,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281	d0c72582de6607c41c0d1807bbc2054ad4e8cb9dfb7ba87103cec569d89bfef7850cab78cad002752246fe9de96a958bb7684c0bffd5006b490beafc772dce9ac0b9388d335d4021df730a758f60e001e8beb656c3a6177214d27e4b7eeb9cf5726b480d37bc510f9ba77c044d95d9f64916e10cca0c74f7e4883eaed65af1b9b525b4b7246a02c11c114a6ea116ce1aa3fd399af8b851dbf89177fc73d1930cb68cd0e9472a7abb24bb2f7a774420ccfe34ed0cb61511e620f4ceeb564c4cae7056d3d8ba770a65ad52321ea87a55881f36c7ee5571b66f676fb39d1db81c55f2fa19f5526960639372c51993aeb6c3310481d7450206d44fc823a068bad2950eddd1c70283dd72b91320f11505767abc0f87892e3f696ceae06afbec104ecc28f47ad244fede8528fc0bea15cf9ea3f9e5e38143b0d5f147ea5552821951a5270226ec2a8e449b	656	c)u6)zH*<)Q*IC*F)U*O)DIu)s)|K*?6*F*/*H)Ce)c(H(0u)e(6*4B*3T)V)()>)L*N*O_(A35`0*7*H*))X(5\F*,*@5(J<(*1)N*(M(.AV*H)U[),)+n(-)c*=)5(IV)Y),)3)B)nY(()S)o)Q/|(C2[)A)f)3*A)OS*GL*)O_)I)MT*9m*((;)2HlXe)e.(>)FG*J)C(Ez)`()(.6*Q)]*P*Q)Cw);~7)y)3)o(<(=w56*,f?i)bL(2x)i)X*/n)*J){*?*4a(,)o)J)r*D)1)4)+)rOs()[G(,)O)T)M)Ou<P*K)z)^\)3B)]N)D*D*L-*>G*7(7)E)nnG)K}(B)-(G(8<)Y(I)RiI)3(M(1W*B(=)j(6)re)EM1*Qh)w(4(=(2)4(9)G*G*B*-D*Q)n])2)^[OP-(4)*)+)9){})KDU()lF(6m)o^o)p)9)I):6(**D(E*:)o<(7)@(=T*K(DQ(L*M){*4u)z)N*JY)((,*Ga*?(AO<])Q)@(LpWi)h*D)_)F/p)+O(58*GJ*B)46(6);q(Ou)/(;7*6)c(()1)f)1(6(D)q)**6)>)jb(B(G*GuX):yE*E)p)r)>(K)Q)B*-)s](G)g(I(3;)})[p(B5:k(/Vl)U;)aG)d)]Y)Oe(9(9?)a*H)p(I)+*1(I)g*0*G**)~)c)-)?yI((v22@	690

100	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,p,i,t, ,a,r,c,u,., ,C,u,m, ,s,o,c,i,i,s,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,112,105,115,32,97,114,99,117,46,32,67,117,109,32,115,111,99,105,105,115,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250,131,37,3,171,473,227,67,573,281,500,284,87,13,75,85,68,86,55,306,33	fb3729ca3afb371143df5e084984851f55b5e7ed848335f51ca690bda323d53543c4d9d65e607380c8c0611a081d4c184923d4b78ae6c2d68a4a31da49cfc6da596d8077677593bd90accf307513ef0a9445f1bdd56b6d09229c55c5933b2d4d32ca1d17a2c4c422737fbf4442004e412d1ca348ba64e4e37e4d92ce7fe6dcc385c4a635a729d8bc7366eea39036ea99425ece35cd9a67832496f6f4ca2b5a17d4670791b721dea62fbd934a224b46f1755247cba1cacc74045c4a1cd9e2c13a1fa45622ef21794a1a8d82a8d0eb95b21c73ccc9c9db2dd3475cf29131b49ce9bc9c5463062e93c2aa93567747038b76ba07b9ad902cf45a9e377343e01ff52fd689e94f7301689bd6596f614452cd58365ce963e7c71ff459e6968cc5c1281226ffb8c55332e9d85614870a051ea8b4803430b9ca22f83603f27276d5f444f64b354cc5acc0da3e41d29fe4134ca219f19303e8fc77f837138541e620fbbf14c204bff31d6ea3757dae7de9b2ed177f	736	e*MtK)R*P)s*->(Mg@ZB)u((L*))u(1*:)9)f*<*8wC)f)E(L)W*I(E*2)\Bh*6.)r(J)Y2)a:9]*H(:)r(*H(-*3*P*3(3K)()o(.*/)p*(<I(=X)6)2q*7(7Y(8)B)>)H*Q(N1\Fe-(>)c])(B*O*P)rL)^(C)O*,t*NS)qK):K)R:)V**H*9.)*S(7;)*){D*O)N)V)w)3E+*5)U@tlv[])[](Rf*))Yd)B)rr)k0(Ab)@*O});(3(@*)`(/*I*8-)1)1jvA(@),)_x)Z)}(J)|)6*0E)A)8(5j)e(0*1(OaN)+(3*On(E(Z(:o(7*14{)`k)M)W*)(3C-):w)kNoTJ*IH(A\)O)usp)>j)F(..)OB)5)M)*I*4)b*-*))(*J*9])9(M((9q3uF)4(L(0C)=<)R(*:FeRu*DP(<)=4)|)o5)2)b)I*8)A*4)0)lW)BN)w*+(7)T)F*E(.*G*:*>)MS*<s*FoE(0)0)/`(>)v*3_*:(I)P(/(J(0)C*:(-n*2?J*N)^ln5*G)@(H0Hz{(L)5WW)1)Y(-@7(B*N)U(C)H)UuW(:)1)W*P*>))]R()(>)*+(/*<SKp)J)8(*-*?*3)~*N(1}*:*Dj])z(,)Y)]5F)xt*M)T\)r(D4Xk8(=){(=*@U*C(3)n(-_)[(MX(H*:J)E*J)1)D*-(9Z)x*A/*,)Q(H*E)n)9(+*4)16((HK*3)_*Q(3y)s-)G**)F)M|)B)g)sJ)c){-)0):*Q)f))	757

EK-8. RSA ŞİFRELEME – ARİTMETİK KODLAMA

	S	*A*	*C*	*RSA*	*x*	*ARİTMETİK KODLAMA*	*x'*
10	{L,o,r,e,m , ,i,p,s,u,}	76,111,114,101,109,32,105,112,115, 117,	110,105,18,342,138,1358, 19,91,391,246	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9	73	3*J*4(K)q);T*x(JWi*@)j)11d)0(:)u)[(3*E(K(;r)l(*()-j(>)[5*;J*9*K~)0x91VE*5Y)NG)yiD`	82
20	{L,o,r,e,m , ,i,p,s,u, m, ,d,o,l,o ,r, ,s,i,}	76,111,114,101,109,32,105,112,115, 117,109,32,100,111,108,111,114,32, 115,105,	110,105,18,342,138,1358, 19,91,391,246,383,398,89 ,60,23,9,533,2858,181,61	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4	143	3*J*4(K)q);T*x(JWi*@)j)11d)0(:)u)[(3*E(K(;r)l(*()-j(>)[5*;J*9*K~)0x91VE*5Y)NG)yh*9*I)-0*H*J)0RG@)C*FG(7)M())T(:X)?a)j)U)[d(7q((m(Jz(9)J*F)Z),)C[)i	146
30	{L,o,r,e,m , ,i,p,s,u, m, ,d,o,l,o ,r, ,s,i,t, ,a ,m,e,t,,, ,c ,o,}	76,111,114,101,109,32,105,112,115, 117,109,32,100,111,108,111,114,32, 115,105,116,32,97 ,109,101,116,44,32,99,111,	110,105,18,342,138,1358, 19,91,391,246,383,398,89 ,60,23,9,533,2858,181,61, 243,13043,449,155,273,40,130,1011,7,22	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5	221	3*J*4(K)q);T*x(JWi*@)j)11d)0(:)u)[(3*E(K(;r)l(*()-j(>)[5*;J*9*K~)0x91VE*5Y)NG)yh*9*I)-0*H*J)0RG@)C*FG(7)M())T(:X)?a)j)U)[d(7q((m(Jz(9)J*F)Z),)B)T)t)p)Y)+*()N)7)<(C)k*N(@b*J)1)H*<(>(:)}-(?@)d)MA(,)(F+(4])0*CJ(8*L))	213
40	{L,o,r,e,m , ,i,p,s,u, m, ,d,o,l,o ,r, ,s,i,t, ,a ,m,e,t,,, ,c ,o,n,s,e,c, t,e,t,u,r,}	76,111,114,101,109,32,105,112,115, 117,109,32,100,111,108,111,114,32, 115,105,116,32,97 ,109,101,116,44,32,99,111,110,115, 101,99,116,101,116,117,114,32,	110,105,18,342,138,1358, 19,91,391,246,383,398,89 ,60,23,9,533,2858,181,61, 243,13043,449,155,273,40,130,1011,7,22,40,4,284, 74,532,119,13,94,216,1150	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c	291	3*J*4(K)q);T*x(JWi*@)j)11d)0(:)u)[(3*E(K(;r)l(*()-j(>)[5*;J*9*K~)0x91VE*5Y)NG)yh*9*I)-0*H*J)0RG@)C*FG(7)M())T(:X)?a)j)U)[d(7q((m(Jz(9)J*F)Z),)B)T)t)p)Y)+*()N)7)<(C)k*N(@b*J)1)H*<(>(:)}-(?@)d)MA(,)(F+(4])0*CJ(6*P)D4(1S(C)[)w)J)v(G*A(Io*:ax(/*DL)R)`)9R)]h(/(/(*)R(3*I	265
50	{L,o,r,e,m , ,i,p,s,u, m, ,d,o,l,o ,r, ,s,i,t, ,a ,m,e,t,,, ,c ,o,n,s,e,c, t,e,t,u,r, ,a ,d,i,p,i,s,c, i,n,g,}	76,111,114,101,109,32,105,112,115, 117,109,32,100,111,108,111,114,32, 115,105,116,32,97 ,109,101,116,44,32,99,111,110,115, 101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,	110,105,18,342,138,1358, 19,91,391,246,383,398,89 ,60,23,9,533,2858,181,61, 243,13043,449,155,273,40,130,1011,7,22,40,4,284, 74,532,119,13,94,216,1150 ,17,72,1,6,234,226,388,135,271,168	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b	359	3*J*4(K)q);T*x(JWi*@)j)11d)0(:)u)[(3*E(K(;r)l(*()-j(>)[5*;J*9*K~)0x91VE*5Y)NG)yh*9*I)-0*H*J)0RG@)C*FG(7)M())T(:X)?a)j)U)[d(7q((m(Jz(9)J*F)Z),)B)T)t)p)Y)+*()N)7)<(C)k*N(@b*J)1)H*<(>(:)}-(?@)d)MA(,)(F+(4])0*CJ(6*P)D4(1S(C)[)w)J)v(G*A(Io*:ax(/*DL)R)`)9R)]h(/(/(*)R(2()(.*/V*9)x*-j)Y1)nk)RU0b*G)c)?fTm)Z)m*+Uy(M*Bx)i	312
60	{L,o,r,e,m , ,i,p,s,u, m, ,d,o,l,o ,r, ,s,i,t, ,a ,m,e,t,,, ,c ,o,n,s,e,c, t,e,t,u,r, ,a ,d,i,p,i,s,c, i,n,g, ,e,l,i ,t,., ,C,u,r, }	76,111,114,101,109,32,105,112,115, 117,109,32,100,111,108,111,114,32, 115,105,116,32,97 ,109,101,116,44,32,99,111,110,115, 101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,	110,105,18,342,138,1358, 19,91,391,246,383,398,89 ,60,23,9,533,2858,181,61, 243,13043,449,155,273,40,130,1011,7,22,40,4,284, 74,532,119,13,94,216,1150 ,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-3	437	3*J*4(K)q);T*x(JWi*@)j)11d)0(:)u)[(3*E(K(;r)l(*()-j(>)[5*;J*9*K~)0x91VE*5Y)NG)yh*9*I)-0*H*J)0RG@)C*FG(7)M())T(:X)?a)j)U)[d(7q((m(Jz(9)J*F)Z),)B)T)t)p)Y)+*()N)7)<(C)k*N(@b*J)1)H*<(>(:)}-(?@)d)MA(,)(F+(4])0*CJ(6*P)D4(1S(C)[)w)J)v(G*A(Io*:ax(/*DL)R)`)9R)]h(/(/(*)R(2()(.*/V*9)x*-j)Y1)nk)RU0b*G)c)?fTm)Z)m*+Uy(M*@*H)O)[)(GV)V(0*N2*>mmyJ)e*.)oF*;)o*>)<i(@)+*KA)6(BX)X(M(H	366

		32,101,108,105,116,46,32,67,117,114,					
70	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-3-e-5-b-e-2-b-e-d-c-5-e-3-5-e-3-c-e-a-9-e-d-c-a-e-5-9-d-e-4-b-e-a-a	503	3*J*4(K)q);T*x(JWi*@)j)11d)0(:)u)[(3*E(K(;r)l(*()-j(>)[5*;J*9*K~)0x91VE*5Y)NG)yh*9*I)-0*H*J)0RG@)C*FG(7)M())T(:X)?a)j)U)[d(7q((m(Jz(9)J*F)Z),)B)T)t)p)Y)+*()N)7)<(C)k*N(@b*J)1)H*<(>(:)}-(?@)d)MA(,)(F+(4])0*CJ(6*P)D4(1S(C)[)w)J)v(G*A(Io*:ax(/*DL)R)`)9R)]h(/(/(*)R(2()(.*/V*9)x*-j)Y1)nk)RU0b*G)c)?fTm)Z)m*+Uy(M*@*H)O)[)(GV)V(0*N2*>mmyJ)e*.)oF*;)o*>)<i(@)+*KA)6(BX)W*:((*1)>)U)t*K0)t)MS)3)T*H`(CH(B)/)fIm)ek)2t0)D)u	411
80	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,a,b,i,t,u,r, ,e,g,e,s,t,a,s, ,s,u,s,c,i,}	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,264,18,77,262,50,216,21,212,28,38,34,229,250	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-3-e-5-b-e-2-b-e-d-c-5-e-3-5-e-3-c-e-a-9-e-d-c-a-e-5-9-d-e-4-b-e-a-a-e-5-9-5-e-2-c-e-5-4-9-e-5-4-e-5-4-5-e-5-b-e-6-b-e-6-d-e-5-5-3-e-5-2-c	573	3*J*4(K)q);T*x(JWi*@)j)11d)0(:)u)[(3*E(K(;r)l(*()-j(>)[5*;J*9*K~)0x91VE*5Y)NG)yh*9*I)-0*H*J)0RG@)C*FG(7)M())T(:X)?a)j)U)[d(7q((m(Jz(9)J*F)Z),)B)T)t)p)Y)+*()N)7)<(C)k*N(@b*J)1)H*<(>(:)}-(?@)d)MA(,)(F+(4])0*CJ(6*P)D4(1S(C)[)w)J)v(G*A(Io*:ax(/*DL)R)`)9R)]h(/(/(*)R(2()(.*/V*9)x*-j)Y1)nk)RU0b*G)c)?fTm)Z)m*+Uy(M*@*H)O)[)(GV)V(0*N2*>mmyJ)e*.)oF*;)o*>)<i(@)+*KA)6(BX)W*:((*1)>)U)t*K0)t)MS)3)T*H`(CH(B)/)fIm)ek)2t0))m3*C)us(0)5N)7)k*Q)W(+)H)R)A)8B?S(.*(4)}VT)B*J	455
90	{L,o,r,e,m, ,i,p,s,u,m, ,d,o,l,o,r, ,s,i,t, ,a,m,e,t,,, ,c,o,n,s,e,c,t,e,t,u,r, ,a,d,i,p,i,s,c,i,n,g, ,e,l,i,t,., ,C,u,r,	76,111,114,101,109,32,105,112,115,117,109,32,100,111,108,111,114,32,115,105,116,32,97,109,101,116,44,32,99,111,110,115,101,99,116,101,116,117,114,32,97,100,105,112,105,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,115,116,97,115,32,115,117,115,99,105,110,103,32,101,108,105,116,46,32,67,117,114,97,98,105,116,117,114,32,101,103,101,	110,105,18,342,138,1358,19,91,391,246,383,398,89,60,23,9,533,2858,181,61,243,13043,449,155,273,40,130,1011,7,22,40,4,284,74,532,119,13,94,216,1150,17,72,1,6,234,226,388,135,271,168,6699,101,40,230,289,241,645,32,129,239,28,58,402,92,90,76,407,	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-3-e-5-b-e-2-b-e-d-c-5-e-3-5-e-3-c-e-a-9-e-d-c-a-e-5-9-d-e-4-b-e-a-a-e-5-9-5-e-2-c-e-5-4-9-e-5-4-e-5-4-5-e-5-b-e-6-b-e-6-d-e-5-5-3-e-5-2-c-e-4-6-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-3-e-5-b-e-2-b-e-d-c-5-e-3-5-e-3-c-e-a-9-e-d-c-a-e-5-9-d-e-4-b-e-a-a-e-5-9-5-e-2-c-e-5-4-9-e-5-4-e-5-4-5-e-5-b-e-6-b-e-6-d-e-5-5-3-e-5-2-c-e-4-6-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-3-e-5-b-e-2-b-e-d-c-5-e-3-5-e-3-c-e-a-9-e-d-c-a-e-5-9-d-e-4-b-e-a-a-e-5-9-5-e-2-c-e-5-4-9-e-5-4-e-5-4-5-e-5-b-e-6-b-e-6-d-e-5-5-3-e-5-2-c-e-4-6-	637	3*J*4(K)q);T*x(JWi*@)j)11d)0(:)u)[(3*E(K(;r)l(*()-j(>)[5*;J*9*K~)0x91VE*5Y)NG)yh*9*I)-0*H*J)0RG@)C*FG(7)M())T(:X)?a)j)U)[d(7q((m(Jz(9)J*F)Z),)B)T)t)p)Y)+*()N)7)<(C)k*N(@b*J)1)H*<(>(:)}-(?@)d)MA(,)(F+(4])0*CJ(6*P)D4(1S(C)[)w)J)v(G*A(Io*:ax(/*DL)R)`)9R)]h(/(/(*)R(2()(.*/V*9)x*-j)Y1)nk)RU0b*G)c)?fTm)Z)m*+Uy(M*@*H)O)[)(GV)V(0*N2*>mmyJ)e*.)oF*;)o*>)<i(@)+*KA)6(BX)W*:((*1)>)U)t*K0)t)MS)3)T*H`(CH(B)/)fIm)ek)2t0))m3*C)us(0)5N)7)k*Q)W(+)H)R)A)8B?S(.*(4)}VTt)g(2*4)v**7)Uyol)T0(7)f*@(Je*3*L)U	496

	a,b,i,t,u,r, ,e,g,e,s,t, a,s, ,s,u,s ,c,i,p,i,t, , a,r,c,u,., ,}	5,99,105,110,103, 32,101,108,105,11 6,46,32,67,117,11 4,97,98,105,116,1 17,114,32,101,103 ,101,115,116,97,1 15,32,115,117,115 ,99,105,112,105,1 16,32,97,114,99,1 17,46,	264,18,77,262,50,216,21, 212,28,38,34,229,250,131 ,37,3,171,473,227,67,573, 281	4-e-6-a-e-6-e-4-a-4-e-d-a-6-e-5-5-a-e-9-a-e-2-a-6-e-5-b-4		0()c)P(-))	
100	{L,o,r,e,m , ,i,p,s,u, m, ,d,o,l,o ,r, ,s,i,t, ,a ,m,e,t,,, ,c ,o,n,s,e,c, t,e,t,u,r, ,a ,d,i,p,i,s,c, i,n,g, ,e,l,i ,t,., ,C,u,r, a,b,i,t,u,r, ,e,g,e,s,t, a,s, ,s,u,s ,c,i,p,i,t, , a,r,c,u,., , C,u,m, ,s, o,c,i,i,s,}	76,111,114,101,10 9,32,105,112,115, 117,109,32,100,11 1,108,111,114,32, 115,105,116,32,97 ,109,101,116,44,3 2,99,111,110,115, 101,99,116,101,11 6,117,114,32,97,1 00,105,112,105,11 5,99,105,110,103, 32,101,108,105,11 6,46,32,67,117,11 4,97,98,105,116,1 17,114,32,101,103 ,101,115,116,97,1 15,32,115,117,115 ,99,105,112,105,1 16,32,97,114,99,1 17,46,32,67,117,1 09,32,115,111,99, 105,105,115,	110,105,18,342,138,1358, 19,91,391,246,383,398,89 ,60,23,9,533,2858,181,61, 243,13043,449,155,273,4 0,130,1011,7,22,40,4,284, 74,532,119,13,94,216,115 0,17,72,1,6,234,226,388,1 35,271,168,6699,101,40,2 30,289,241,645,32,129,23 9,28,58,402,92,90,76,407, 264,18,77,262,50,216,21, 212,28,38,34,229,250,131 ,37,3,171,473,227,67,573, 281,500,284,87,13,75,85, 68,86,55,306,33	4-4-c-e-4-c-2-e-4-b-e-6-d-5-e-4-6-b-e-4-6-2-b-e-4-3-e-3-4-e-6-3-4-e-5-d-9-e-6-b-6-e-6-3-b-e-b-3-e-9-c-e-5-6-e-3-e-2-6-6-e-5-b-2-b-e-4-b-4-e-9-4-e-5-d-6-e-4-6-c-d-6-e-d-d-3-e-4-2-2-e-5-a-6-e-d-c-e-4-6-c-e-4-c-4-4-e-a-e-5-5-e-d-c-e-d-e-5-b-d-e-a-d-e-2-6-5-e-4-4-3-e-4-6-e-3-d-e-5-4-9-e-4-4-2-c-e-4-a-e-a-5-e-4-e-9-e-5-6-d-e-5-5-9-e-6-b-b-e-4-6-2-e-5-a-4-e-4-9-b-e-9-9-3-3-e-4-c-4-e-d-c-e-5-6-c-e-5-b-3-e-5-d-4-e-9-d-2-e-6-5-e-4-5-3-e-5-6-3-e-5-b-e-2-b-e-d-c-5-e-3-5-e-3-c-e-a-9-e-d-c-a-e-5-9-d-e-4-b-e-a-a-e-5-9-5-e-2-c-e-5-4-9-e-5-4-e-5-4-5-e-5-b-e-6-b-e-6-d-e-5-5-3-e-5-2-c-e-4-6-4-e-6-a-e-6-e-4-a-4-e-d-a-6-e-5-5-a-e-9-a-e-2-a-6-e-5-b-4-e-2-c-c-e-5-b-d-e-b-a-e-4-6-e-a-2-e-b-2-e-9-b-e-b-9-e-2-2-e-6-c-9-e-6-6	709	3*J*4(K)q);T*x(JWi*@)j)11d)0(:)u)[(3*E(K(;r)l(*()-j(>)[5*;J*9*K~)0x91VE*5Y)NG)yh*9*I)-0*H*J)0RG@)C*FG(7)M())T(:X)?a)j)U)[d(7q((m(Jz(9)J*F)Z),)B)T)t)p)Y)+*()N)7)<(C)k*N(@b*J)1)H*<(>(:)}-(?@)d)MA(,)(F+(4])0*CJ(6*P)D4(1S(C)[)w)J)v(G*A(Io*:ax(/*DL)R)`)9R)]h(/(/(*)R(2()(.*/V*9)x*-j)Y1)nk)RU0b*G)c)?fTm)Z)m*+Uy(M*@*H)O)[)(GV)V(0*N2*>mmyJ)e*.)oF*;)o*>)<i(@)+*KA)6(BX)W*:((*1)>)U)t*K0)t)MS)3)T*H`(CH(B)/)fIm)ek)2t0))m3*C)us(0)5N)7)k*Q)W(+)H)R)A)8B?S(.*(4)}VTt)g(2*4)v**7)Uyol)T0(7)f*@(Je*3*L)U0()c)O<)\)Q(Nj(7)QI(I1*<)5(,d)2)}(:)c))*7)k)((;K(**9)]*L).@	545

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı	: KENDİRLİ, Oğuzhan
Uyruğu	: T.C.
Doğum tarihi ve yeri	: 06.06.1984 Niğde
Telefon	: 05053148952
E-posta	: oguzhankendirli@duzce.edu.tr

Eğitim

Derece	**Eğitim Birimi**	**Mezuniyet tarihi**
Yüksek Lisans	Düzce Üniversitesi	
Lisans	Selçuk Üniversitesi	2008
Lise	F.Ş. Anadolu Teknik Lisesi	2002

İş Deneyimi

Yıl	**Yer**	**Görev**
2013-	Düzce Üniversitesi	Öğretim Görevlisi
2012-2013	Nevşehir Üniversitesi	Öğretim Görevlisi
2011-2012	MEB	Öğretmen
2010-2010	Selçuk Üniversitesi	Öğretim Görevlisi
2008-2009	MEB	Öğretmen
2006-2006	ODTÜ	Teknisyen

Yabancı Dil

İngilizce (ÜDS:60)

Yayınlar

1. Şatır E., Kendirli O., A Hybrid Steganographic Approach Via Web Addresses, *İleri Teknoloji Bilimleri Dergisi,* 2 (3) (**2013**) 53-60.