VERİ DEPOLAMA VE SIKIŞTIRMA ALGORİTMALARI

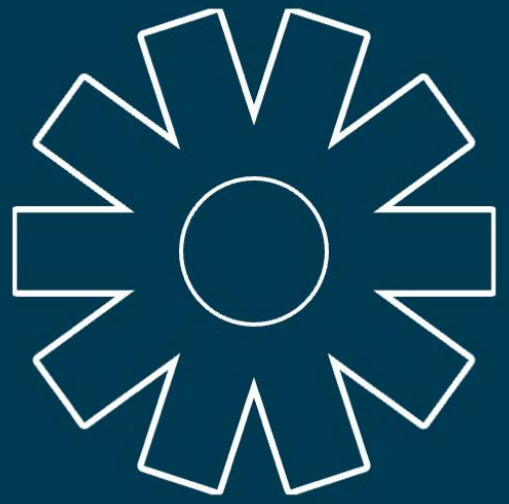
SUNUM İÇERİĞİ

1- Metin, resim ve ses verilerinin bit düzeyinde temsili.

2- Veri sıkıştırma neden gereklidir?

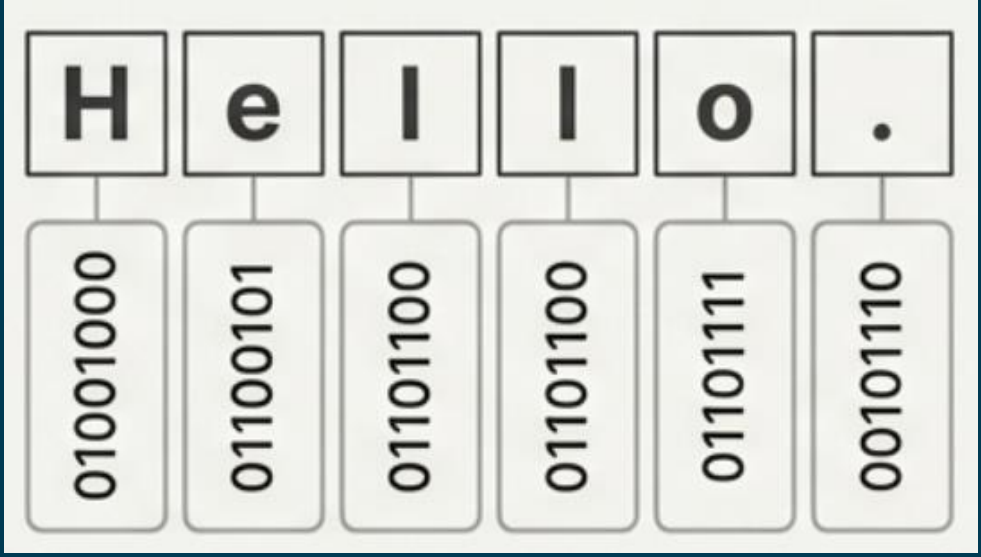
3- Run-Length Encoding (RLE) gibi temel sıkıştırma mantıkları.

METİN, RESİM VE SES VERİLERİNİN BİT DÜZEYİNDE TEMSİLİ



Bilgisayarlar tüm verileri **bit (0 ve 1)** seviyesinde işler. Metin, resim ve ses gibi farklı veri türleri; bilgisayar tarafından anlaşılabilmesi için **ikili (binary) biçimde** temsil edilir. Bu sunumda, bu üç veri türünün **bit düzeyinde** nasıl temsil edildiği açıklayacağım.

METİN VERİLERİNİN BİT DÜZEYİNDE TEMSİLİ

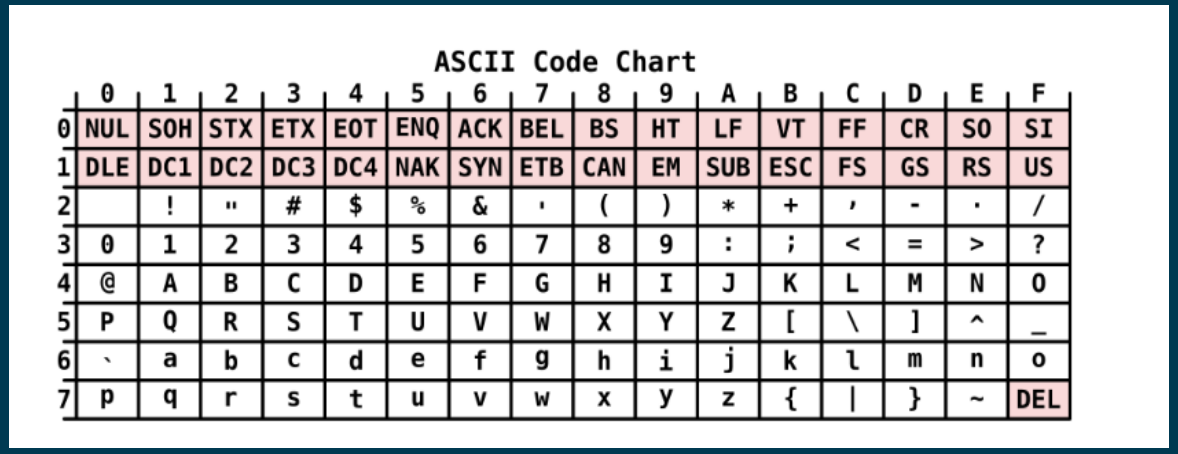


Her karaktere (harf, rakam, sembol) benzersiz bir bit deseni atanır.Dünya çapında kulllanılan iki karakter bit temsili vardır. Bunlar;

ASCII İngilizce için temel standartken

Unicode (UTF-8) tüm dünya dillerini kapsar.

ASCII



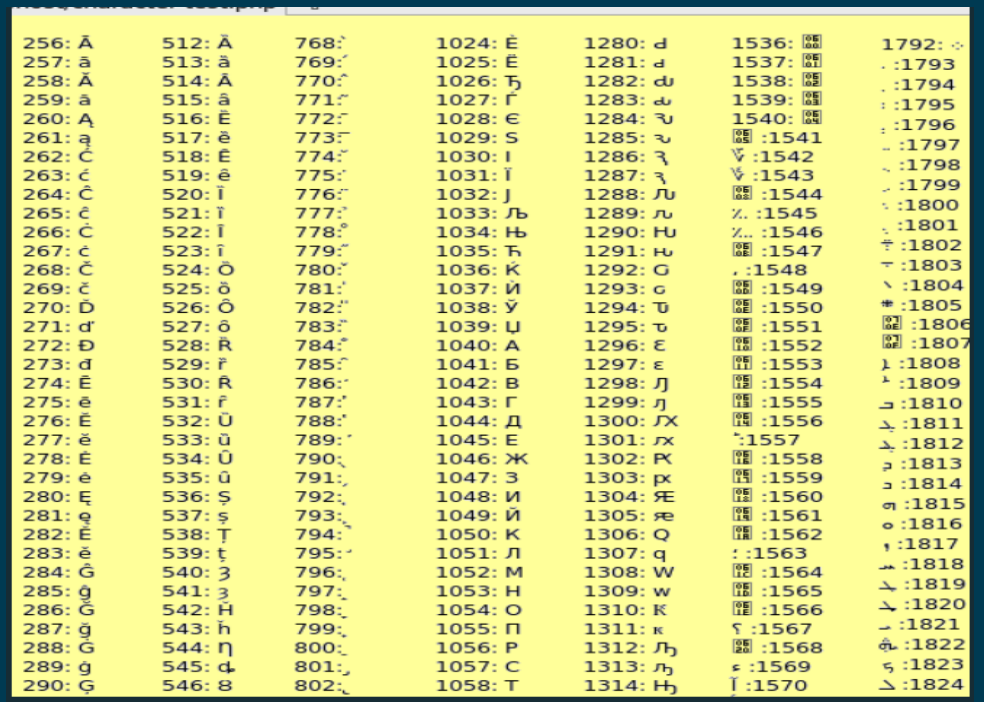
ASCII Code Chart

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

İngiliz alfabesindeki sembol ve ifadelerin gösterimi için 7-bit deseni kullanır

Bütün dünya dillerine açık değildir.

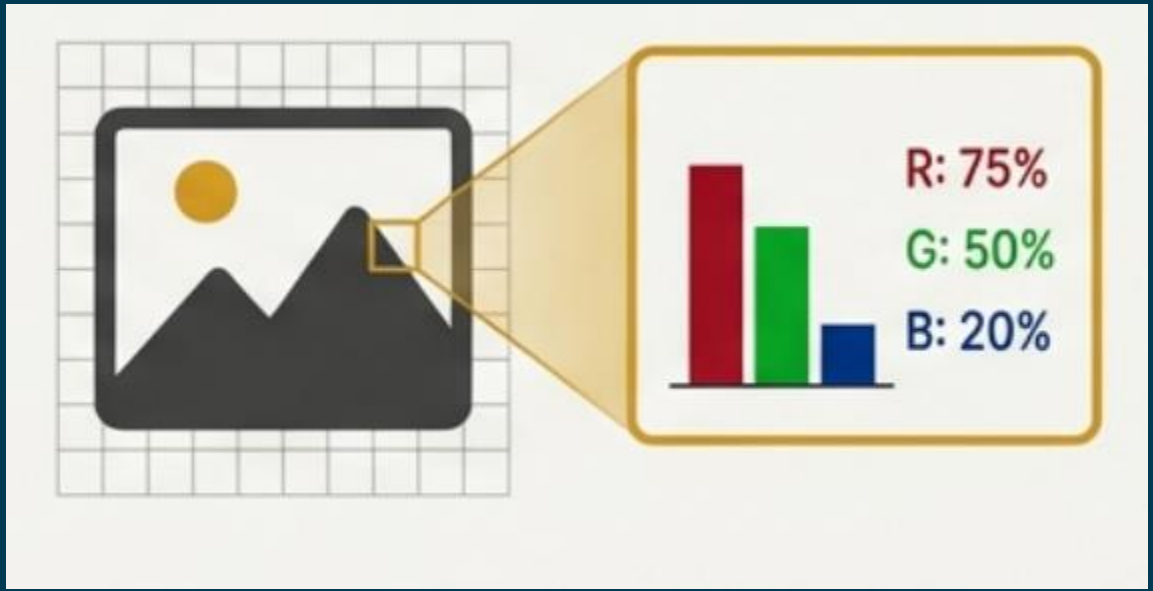
UNİCODE (UTF-8)



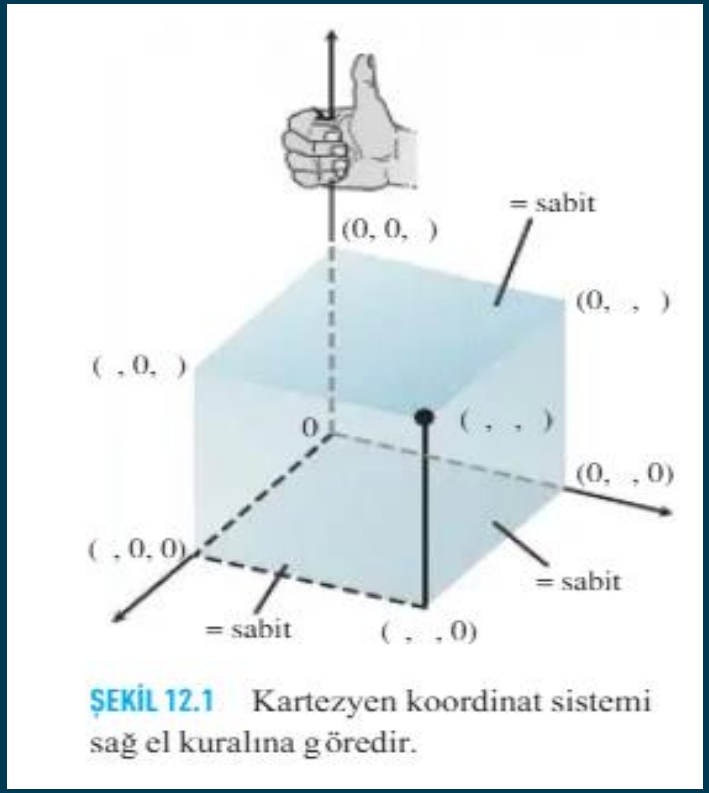
A selection of Unicode code points viewed in Firefox

- Dünya genelinde kullanılan dillerdeki sembol ve ifadelerin gösterimi için 16-bit 21-bit arası desenleri kullanır.
- Kullanıcılar açısından daha kullanışlıdır

RESİM VERİLERİNİN BİT DÜZEYİNDE TEMSİLİ



- Bit Eşlem (Bitmap) görüntüyü piksel matrisi olarak kodlar. Her pikselin rengi RGB (Kırmızı, Yeşil, Mavi) değerleriyle belirtilir.
- Renklerin elde edilmesi ise kırmızı, yeşil,mavi renklerin belli oranlarda karıştırılması ile elde edilir.

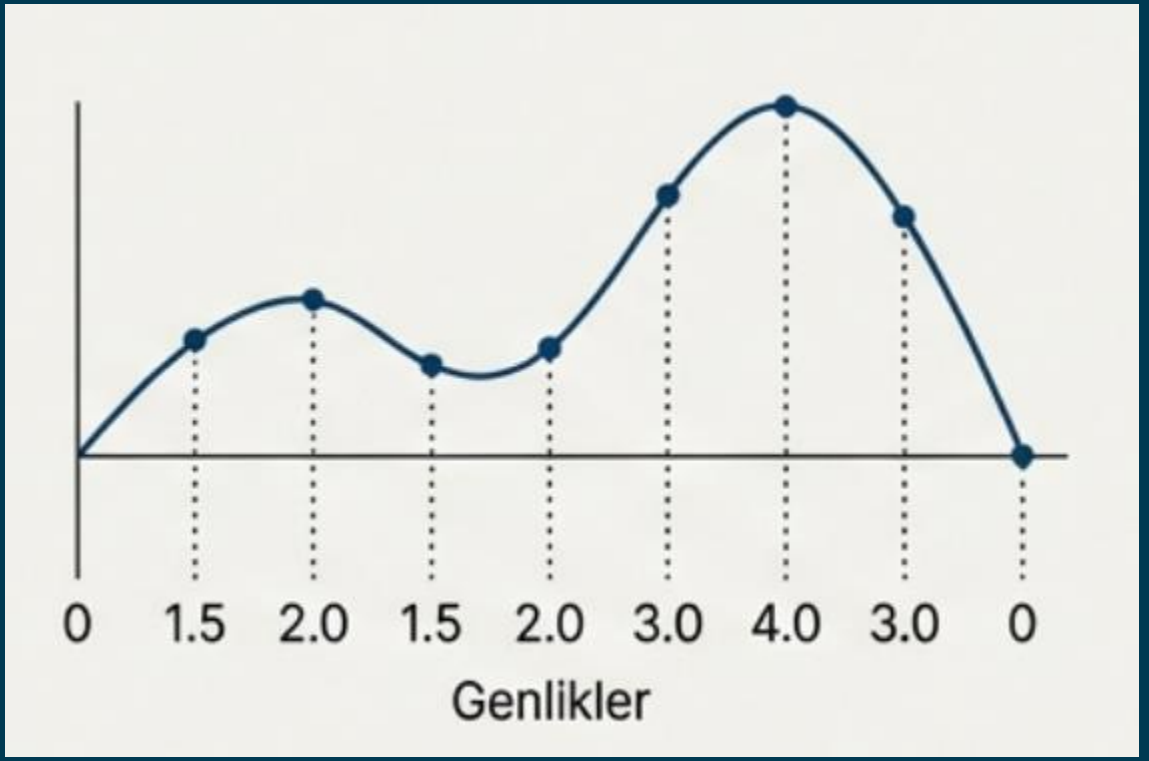


ŞEKİL 12.1 Kartezyen koordinat sistemi sağ el kuralına göredir.

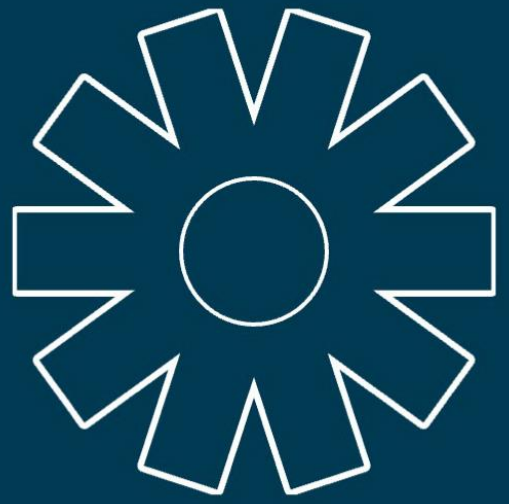
- Vektör ise görüntüyü geometrik formüllerle tanımlar ve kalite kaybı olmadan ölçeklenir.
- Resmin ölçeklenmesini kolaylaştırır
- görüntü büyütüldüğünde piksellenme (kalite kaybı) yaşanmaz

SES VERİLERİNİN BİT DÜZEYİNDE TEMSİLİ

- Ses, doğası gereği sürekli bir dalgadır ancak bilgisayarlar bu dalgayı dijitalleştirmek için **örnekleme (sampling)** tekniklerini kullanır
- Ses dalgasının genliği belirli zaman aralıklarında ölçülür ve bu ölçümler sayılara (bit desenlerine) dönüştürülür
- **Örnekleme hızı**, sesin kalitesini belirler. Örneğin, uzun mesafe telefon görüşmelerinde saniyede 8,000 örnek alınırken, yüksek kaliteli bir CD sesinde saniyede 44,100 örnek alınır



VERİ SIKIŞTIRMA NEDEN GEREKLİDİR?



Veri sıkıştırma, dijital dünyada verinin yönetilebilir, depolanabilir ve iletilebilir olması için kritik bir gerekliliktir. Kaynaklara göre bu gerekliliğin temel nedenleri şunlardır:

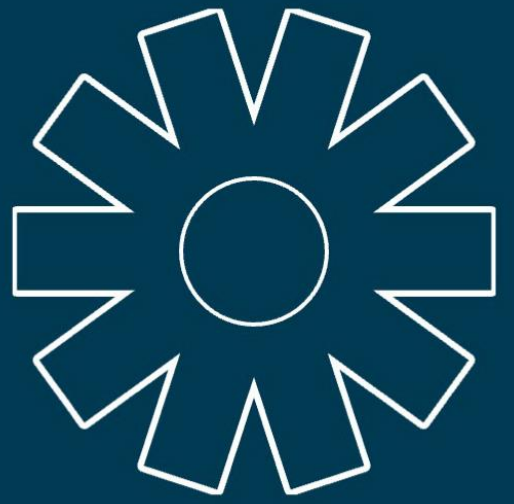
1-**Depolama Alanı Tasarrufu**

2-**Bant Genişliği Verimliliği**

3-**Hız ve Düşük Gecikme**

4-**Yazılım ve Donanım Uyumluluğu**

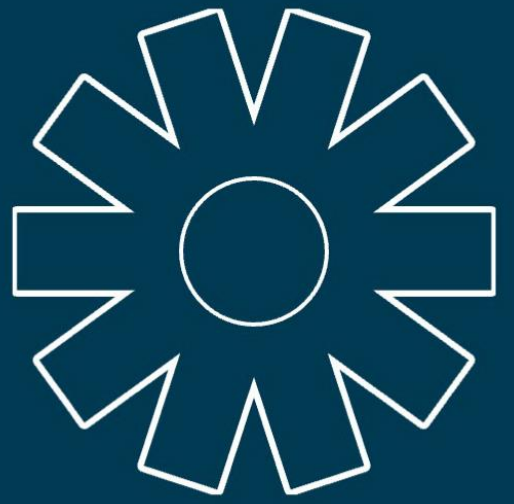
DEPOLAMA ALANI TASARRUFU



Yüksek çözünürlüklü görüntüler (pikseller) ve kaliteli ses kayıtları (saniyede binlerce örnekleme) devasa boyutlarda bit desenleri oluşturur. Sıkıştırma, bu verilerin manyetik diskler, flash sürücüler veya SD kartlar gibi **yığın depolama aygıtlarında** çok daha az yer kaplamasını sağlar.

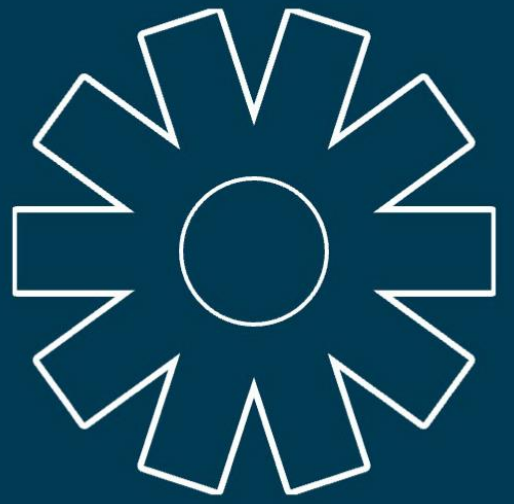
BANT GENİŞLİĞİ VERİMLİLİĞİ

Bant genişliği, birim zamanda transfer edilebilen toplam bit miktarını ifade eder. Veri sıkıştırıldığında, aynı ağ kapasitesi üzerinden daha fazla bilgi gönderilebilir; bu da özellikle internet üzerinden **video konferans** veya **yüksek kaliteli televizyon yayınları (MPEG)** için hayati önem taşır.



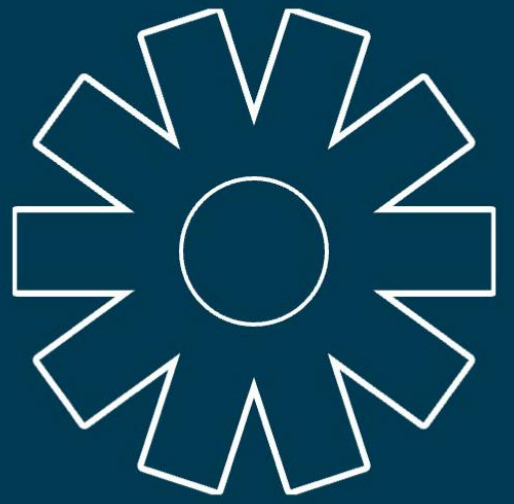
HIZ VE DÜŞÜK GECİKME

Veri transferi isteği ile verinin ulaşması arasında geçen süreye **gecikme (latency)** denir. Sıkıştırılmış veri daha küçük bir boyuta sahip olduğu için iletim hattı üzerinden daha hızlı taşınır ve gecikme süresini minimize eder.



YAZILIM VE DONANIM UYUMLULUĞU

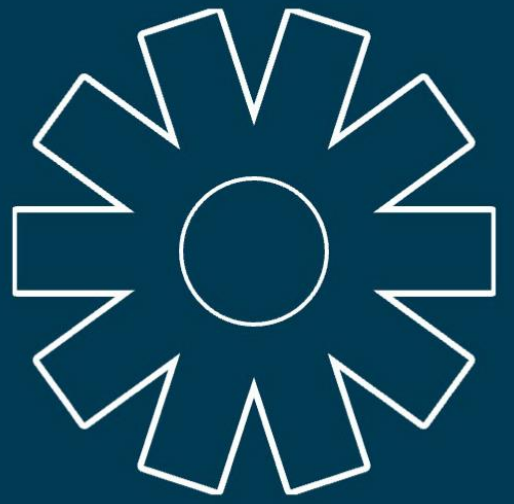
Günümüzde kullandığımız MP3 (ses/müzik) ve JPEG (fotoğraf) gibi popüler formatlar, veriyi hem kaliteli hem de taşınabilir boyutlarda tutmak için geliştirilmiş özel sıkıştırma standartlarıdır.



VERİ SIKIŞTIRMA OLMASAYDI, BUGÜN KULLANDIĞIMIZ AKILLI TELEFONLAR VE İNTERNET SERVİSLERİ BU KADAR BÜYÜK MİKTARDA GÖRSEL VE İŞİTSEL VERİYİ BU KADAR HIZLI İŞLEYEMEZDİ.

TEMEL SIKIŞTIRMA MANTIKLARI

veri sıkıştırma yöntemleri temel olarak **kayıplı (lossy)** ve **kayıpsız (lossless)** olmak üzere ikiye ayrılır.



KAYIPSIZ SIKIŞTIRMA

Kullanım Alanları:

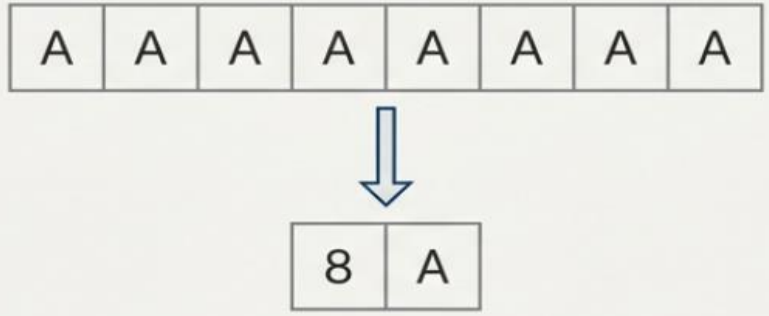
Metin dosyaları (ZIP), program kodları, veri tabanları. Bilginin %100 korunması gereken her yer.

Örnek Formatlar:

PNG, GIF, ZIP.

Kayıpsız (Lossless)

Orijinal verinin tamamı sıkıştırılmış halden geri geri elde edilebilir. Tekrarlanan desenleri daha verimli bir şekilde kodlar.



GIF (Kayıpsız)

KAYIPLI SIKIŞTIRMA

1.2 MB

150 KB

JPEG (Kayıplı), MP3 (Kayıplı)

Kayıplı Sıkıştırma (Lossy)

Dosya boyutunu önemli ölçüde azaltmak için insan algısının fark edemeyeceği bazı bilgiler kalıcı olarak silinir.

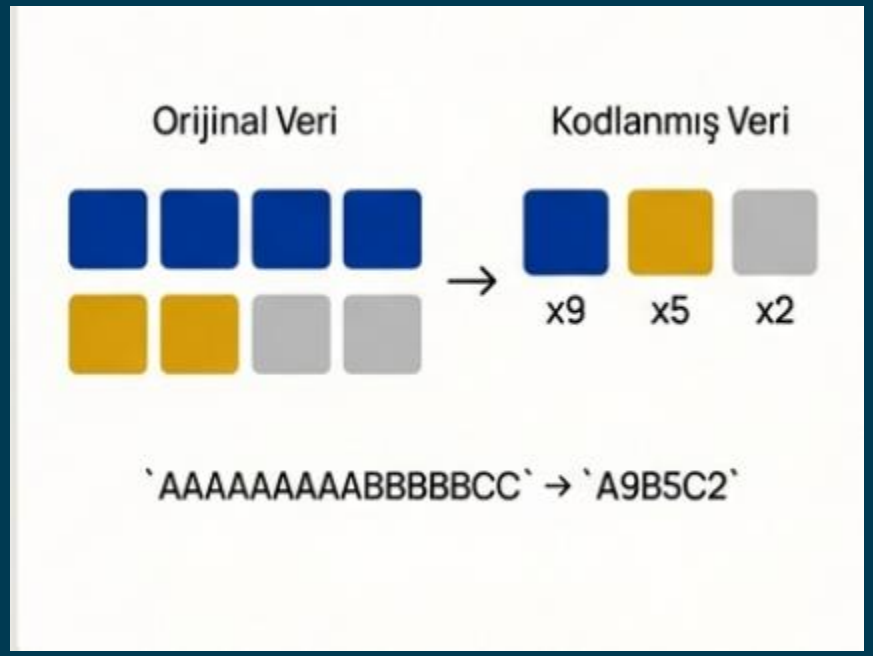
Kullanım Alanları:

Fotoğraflar, videolar, müzik. Kalitede küçük bir kaybın kabul edilebilir olduğu durumlar.

Örnek Formatlar:

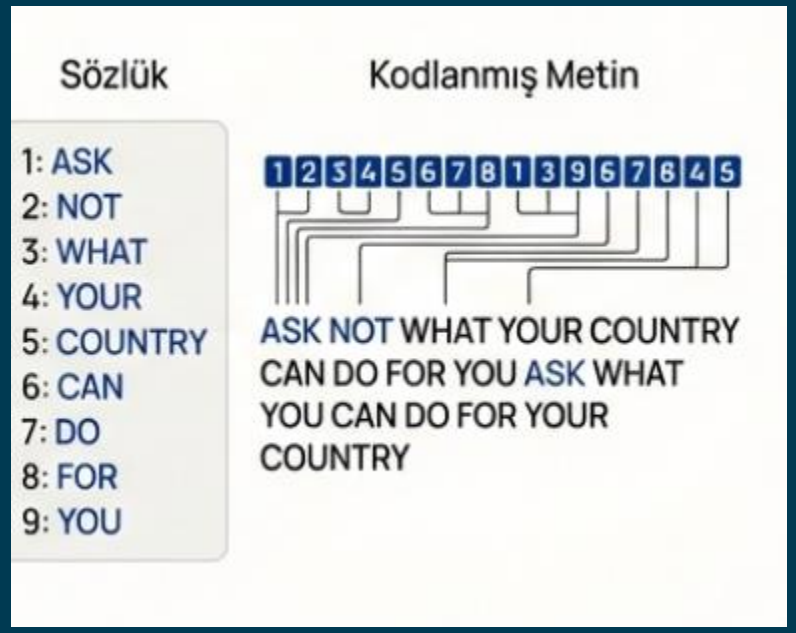
JPEG, MP3, MPEG.

1. İŞLEM UZUNLUĞU KODLAMA (RUN-LENGTH ENCODİNG - RLE)



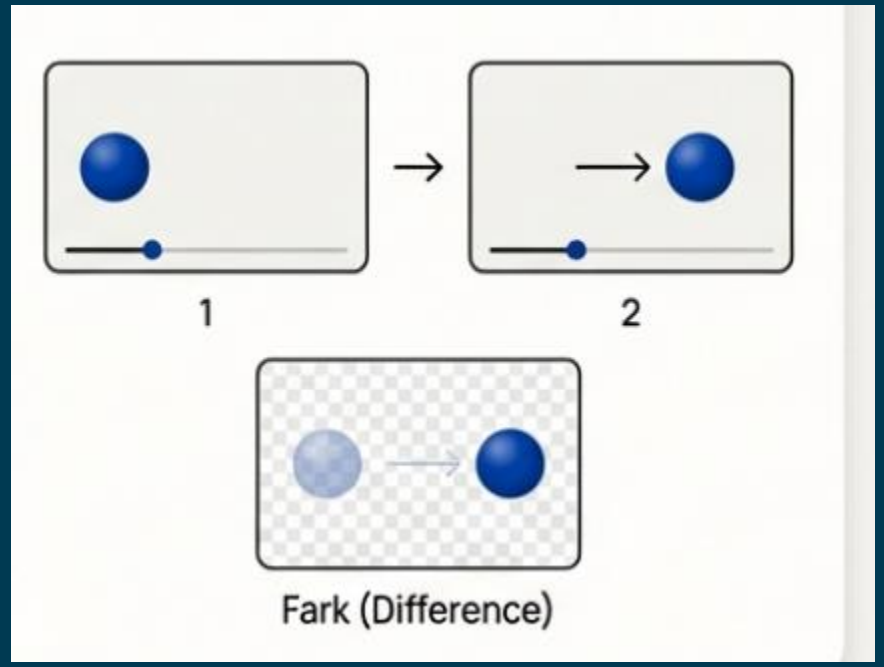
- Aynı karakterin veya değerin uzun bir dizi boyunca tekrar etmesi durumunda, bu karakteri tek tek yazmak yerine karakterin kendisini ve kaç kez tekrar ettiğini kaydederek sıkıştırma yapar
- Metin dosyaları için genellikle uygun değildir; çünkü metinlerde aynı harflerin yan yana ikiden fazla tekrar etmesi nadir bir durumdur

SÖZLÜK TABANLI KODLAMA (DİCTİONARY-BASED)



- "Sözlük" adı verilen bir listede kelimelere numaralar atanır. Orijinal metin içinde bu kelimeler geçtiğinde, kelimenin kendisi yerine sözlükteki indeks numarası yazılır. Örneğin, uzun bir metindeki "COUNTRY" kelimesi yerine sadece "5" rakamı kullanılarak veri boyutu küçültülür.

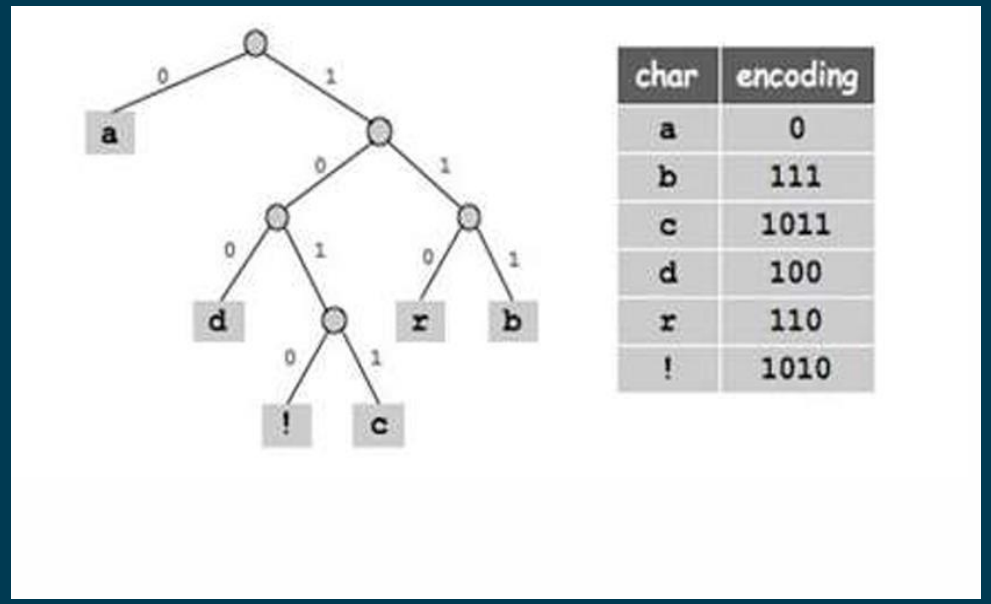
GÖRECELİ KODLAMA (RELATİVE ENCODİNG)



- Verinin tamamını saklamak yerine, bir önceki veri bloğundan sadece farkları kaydeder.Video sıkıştırmada yaygındır.
- Fotoğraflar için **JPEG** ve **PNG**, çizgi filmler için **GIF** formatları farklı algoritmalar kullanır.

FREKANS BAĞIMLI KODLAMA (HUFFMAN KODLARI)

- Bir veri setinde en sık kullanılan karakterlere en kısa bit desenlerini, daha nadir kullanılanlara ise daha uzun bit desenlerini atar. Bu sayede toplamda kullanılan bit sayısı minimize edilir



char	encoding
a	0
b	111
c	1011
d	100
r	110
!	1010

KAYNAKÇA

1- wikipedia

2 - chapter-1