



**BURSA TEKNİK
ÜNİVERSİTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ
BLM101 - BİLGİSAYAR MÜHENDİSLİĞİNE GİRİŞ
ÇOK FONKSİYONLU TABAN DÖNÜŞTÜRÜCÜ
DÖNEM PROJESİ**

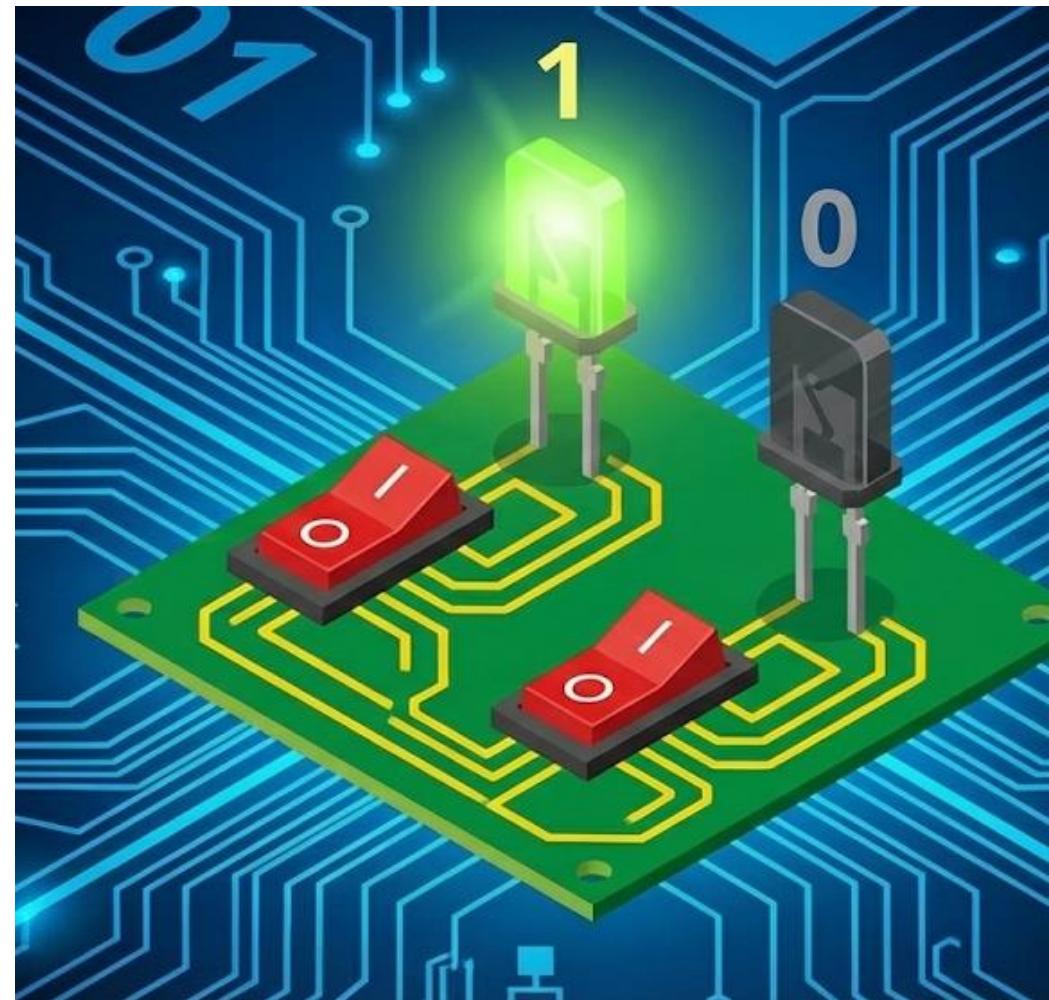
İŞİM-SOYİSİM: NİSA DERİN ERDOĞAN

ÖĞRENCİ NO: 25360859335

KONU: VERİ DEPOLAMA VE SAYISAL SİSTEMLER

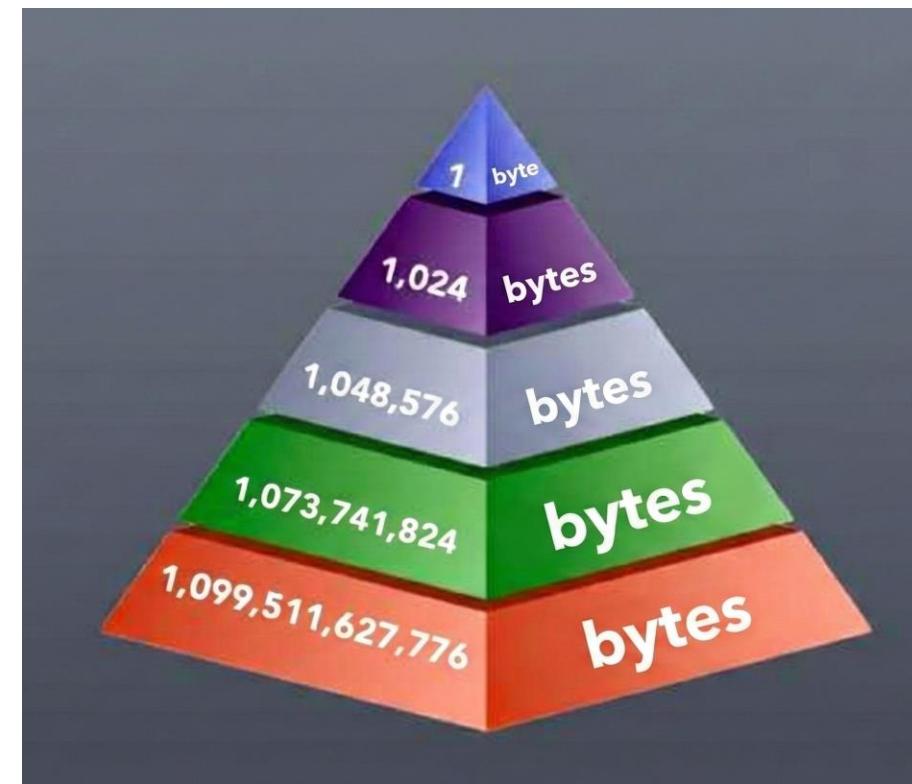
BIT KAVRAMI

- Bit (Binary Digit): Bilgisayardaki en küçük veri depolama birimidir.
- Değerleri: 0 veya 1 şeklindedir.
- Elektronik devrelerde açık/kapalı durumları temsil eder.



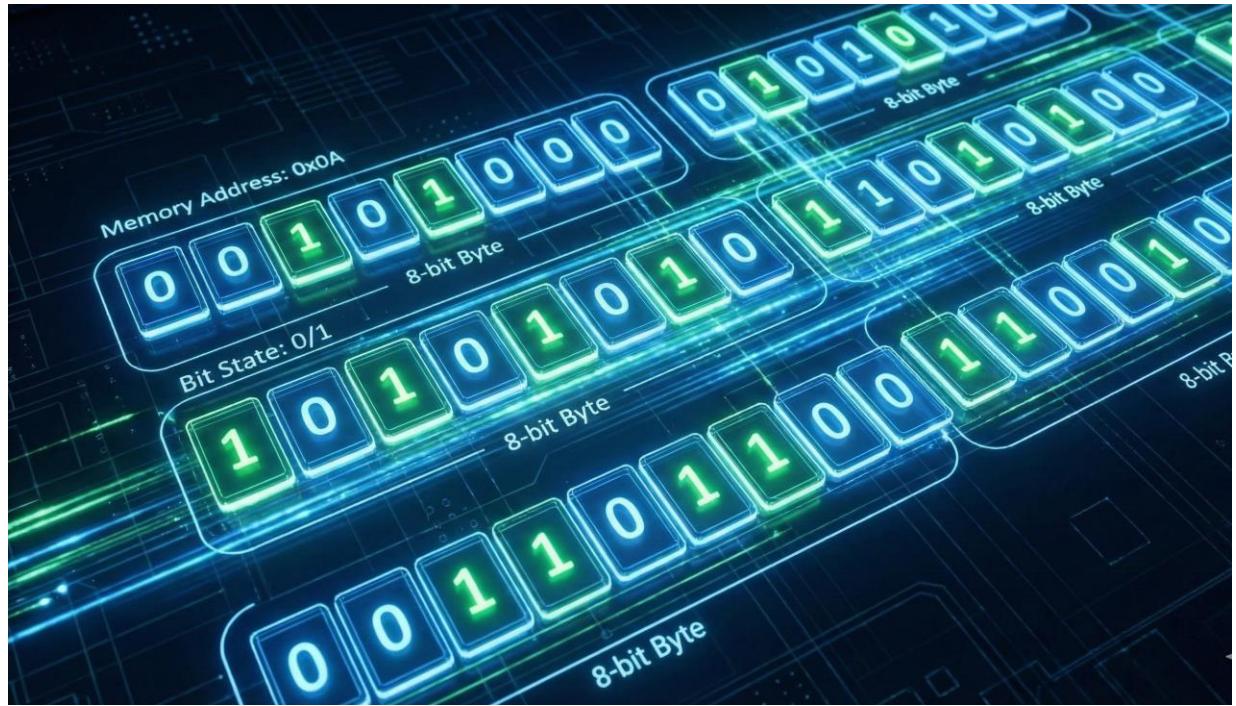
BYTE KAVRAMI

- 1 Byte = 8 Bit
- Bilgisayarlarda bellek adresleme byte üzerinden yapılır.
- Bir karakter genellikle 1 byte ile temsil edilir.



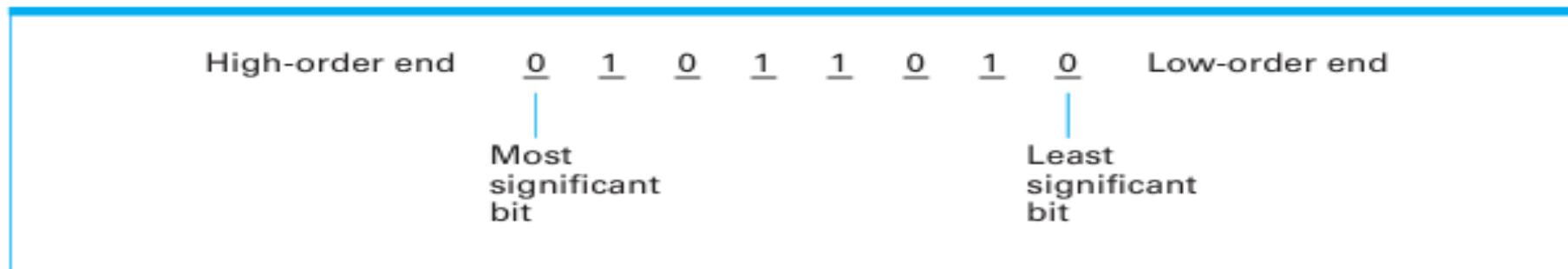
BELLEKTE BIT GÖSTERİMİ

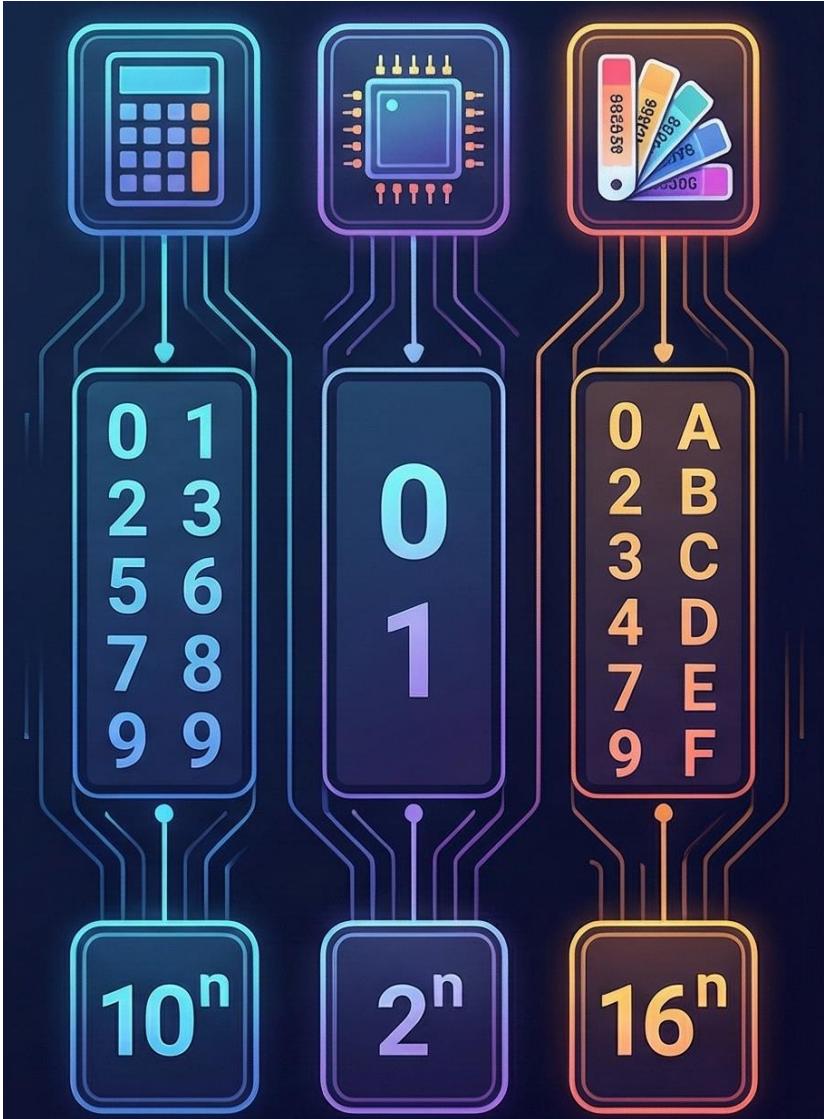
Donanım seviyesinde veri temsili, her bir kutucuğun tek bir biti simgelediği 8 bitlik gruplar (bytelar) halinde organize edilir.



EN ÇOK ANLAMLI VE EN AZ ANLAMLI BITLER MSB-LSB

Figure 1.7 The organization of a byte-size memory cell





BASAMAK DEĞERİ SİSTEMLERİ

- Sayının değeri basamağın yerine bağlıdır.
- Her basamak bir kuvveti temsil eder.
- Tabana göre (base) hesaplama yapılır.

ONLUK SİSTEM VE İKİLİK SİSTEM

1. Onluk Sayı Sistemi (Decimal)

(İnsanların Kullandığı Sistem)

- **Taban:** 10 (Logaritma ve günlük hesaplamaların temeli).
- **Kullanılan Rakamlar:** 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
- **Basamak Mantığı:** Her basamak, **10'un kuvvetleri** olarak artar (1'ler, 10'lar, 100'ler basamağı).
- **Örnek:** 25 sayısı = $(2 \times 10) + (5 \times 1)$.

ONLUK SİSTEM VE İKİLİK SİSTEM

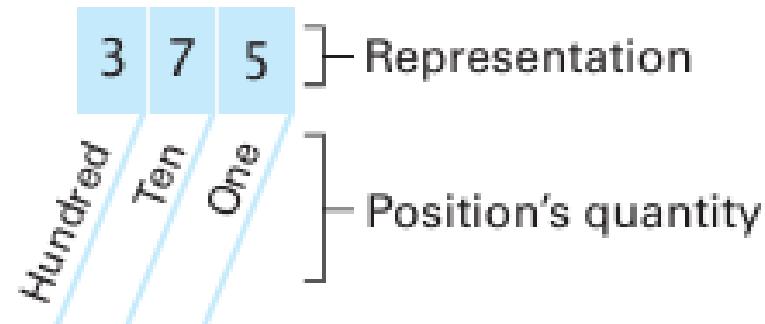
2. İkilik Sayı Sistemi (Binary)

(Bilgisayarların Kullandığı Sistem)

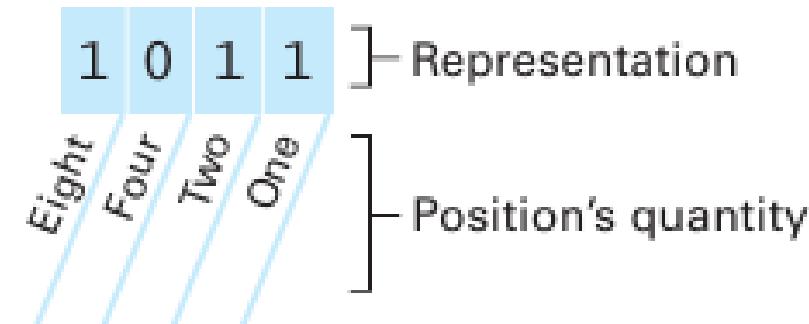
- **Taban:** 2 (Dijital elektronliğin temeli).
- **Kullanılan Rakamlar:** Sadece **0** ve **1**.
- **Basamak Mantığı:** Her basamak, **2'nin kuvvetleri** olarak artar (1, 2, 4, 8, 16...).
- **Örnek:** 11001 (Bu ifade Onluk sistemde **25** sayısına eşittir).

Figure 1.13 The base 10 and binary systems

a. Base 10 system



b. Base two system

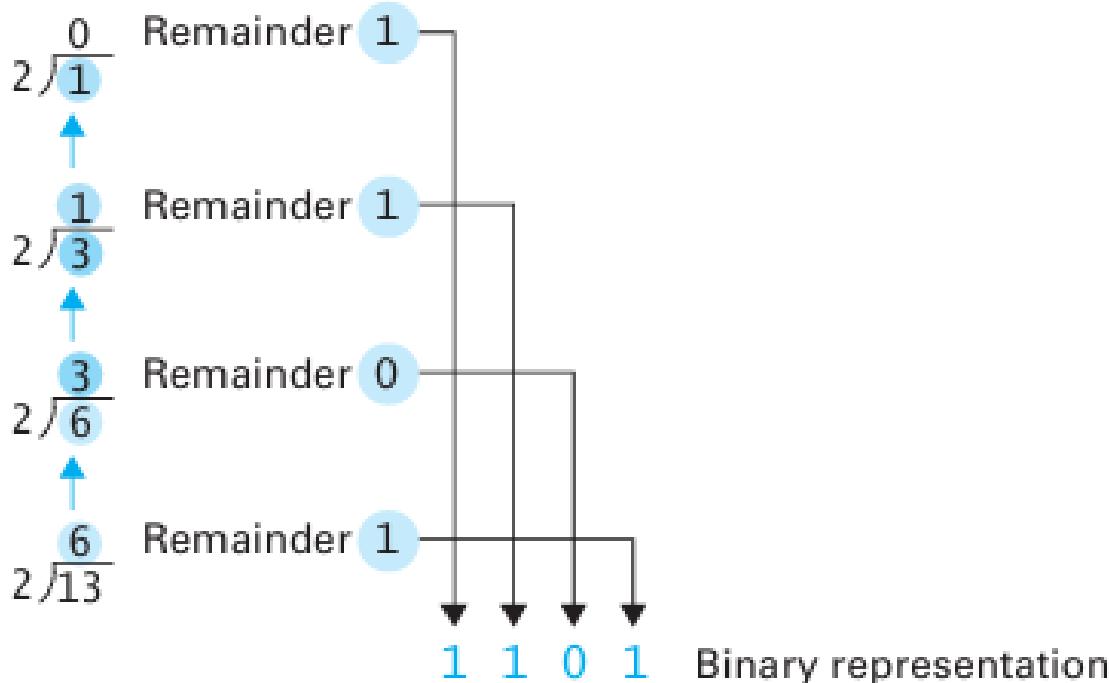


ONLUK → BINARY DÖNÜŞÜM ALGORİTMASI

Figure 1.15 An algorithm for finding the binary representation of a positive integer

- Step 1. Divide the value by two and record the remainder.
- Step 2. As long as the quotient obtained is not zero, continue to divide the newest quotient by two and record the remainder.
- Step 3. Now that a quotient of zero has been obtained, the binary representation of the original value consists of the remainders listed from right to left in the order they were recorded.

Figure 1.16 Applying the algorithm in Figure 1.15 to obtain the binary representation of thirteen



ONALTILIK (HEXADECIMAL) GÖSTERİM

Tabanı 16 olan ve 0-9 ile A-F sembollerini kullanan Onaltılık (Hexadecimal) sistem, özellikle büyük binary sayıların daha kolay okunup ifade edilmesini sağlar.

Figure 1.6 The hexadecimal encoding system

Bit pattern	Hexadecimal representation
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

ONLUK → HEXADECIMAL DÖNÜŞÜM

- Sayı 16'ya bölünür
- Kalanlar hex tablosuna göre yazılır.
- Kalanlar tersten okunur.

		Remainders
16	2547	3 (LSB)
16	159	15 [15=F]
16	9	9 (MSB)
	0	

$$\text{Decimal } (2547)_{10} = \text{Hexadecimal } (9F3)_{16}$$

NEGATİF SAYI PROBLEMI

- Binary sistemde negatif işaret yoktur.
- Ayrı bir temsil yöntemi gereklidir.
- Çözüm: Two's Complement

Figure 1.19 Two's complement notation systems

a. Using patterns of length three

Bit pattern	Value represented
011	3
010	2
001	1
000	0
111	-1
110	-2
101	-3
100	-4

b. Using patterns of length four

Bit pattern	Value represented
0111	7
0110	6
0101	5
0100	4
0011	3
0010	2
0001	1
0000	0
1111	-1
1110	-2
1101	-3
1100	-4
1011	-5
1010	-6
1001	-7
1000	-8

TWO'S COMPLEMENT

- **Tanım ve Önem:** Modern bilgisayarlarda negatif sayıları temsil etmek ve aritmetik işlemleri basitleştirmek (çıkarma yerine toplama yapmak) için kullanılan standart yöntemdir.
- **Hesaplama:** Sayının bitleri ters çevrilip ($1 \leftrightarrow 0$) elde edilen değere 1 eklenerek oluşturulur.

Figure 1.21 Addition problems converted to two's complement notation

Problem in base 10	Problem in two's complement	Answer in base 10
$\begin{array}{r} 3 \\ + 2 \\ \hline \end{array}$	$\begin{array}{r} 0011 \\ + 0010 \\ \hline 0101 \end{array}$	5
$\begin{array}{r} -3 \\ + -2 \\ \hline \end{array}$	$\begin{array}{r} 1101 \\ + 1110 \\ \hline 1011 \end{array}$	-5
$\begin{array}{r} 7 \\ + -5 \\ \hline \end{array}$	$\begin{array}{r} 0111 \\ + 1011 \\ \hline 0010 \end{array}$	2

ÇOK FONKSİYONLU TABAN DÖNÜŞTÜRÜCÜ

Hazır fonksiyonlar (bin, hex) yerine saf matematiksel algoritmalar kullanan bu proje, onluk sayıları istege göre İkilik veya Onaltılık tabana dönüştürür. Dönüşüm sonuçlarını sadece sayısal olarak değil, aynı zamanda bellekteki 8-bitlik fiziksel karşılığını simüle eden kutucuklar halinde görselleştirerek sunar.

--- Cok Fonksiyonlu Taban Donusturucu ---

- 1 - Binary cevir
- 2 - Hexadecimal cevir
- 3 - Cikis

Seciminizi giriniz:

PROGRAMIN ALGORİTMASI

- Adım 1. Başla
- Adım 2. Menüyü göster
- Adım 3. Kullanıcıdan seçim al
- Adım 4. Eğer seçim geçersizse
Uyarı ver ve menüye dön
- Adım 5. Kullanıcıdan onluk sayı al
- Adım 6. Eğer Binary seçilmişse
Eğer sayı ≥ 0 ise
Binary dönüşümünü yap
Aksi halde
Two's Complement uygula
- Adım 7. Eğer Hexadecimal seçilmişse
Eğer sayı < 0 ise
Uyarı ver ve menüye dön
Aksi halde
Onluk \rightarrow Hexadecimal dönüşümü yap
- Adım 8. Sonucu ve bellek görünümünü ekranaya yazdır
- Adım 9. Menüye geri dön
- Adım 10. Bitir

BINARY DÖNÜŞÜM ALGORİTMASI

Fonksiyon onluk_binary_cevir(sayı)

Adım 1. Eğer sayı = 0 ise

"0" döndür

Adım 2. Boş bir liste oluştur (bitler)

Adım 3. Sayı > 0 olduğu sürece tekrarla

a. Kalan = sayı mod 2

b. Kalanı listeye ekle

c. sayı = sayı / 2 (tam sayı bölme)

Adım 4. Bit listesini ters çevir (MSB → LSB)

Adım 5. Bitleri birleştir ve sonucu döndür

Fonksiyon Sonu

HEXADECIMAL DÖNÜŞÜM ALGORİTMASI

Fonksiyon onluk_hex_cevir(sayı)

Adım 1. Eğer sayı = 0 ise
 "0" döndür

Adım 2. Hex karakter tablosu tanımla
 (0-9, A-F)

Adım 3. Boş bir liste oluştur (basamaklar)

Adım 4. Sayı > 0 olduğu sürece tekrarla

- a. Kalan = sayı mod 16
- b. Kalanın hex karşılığını listeye ekle
- c. sayı = sayı / 16 (tam sayı bölme)

Adım 5. Listeyi ters çevir

Adım 6. Basamakları birleştir ve sonucu döndür

Fonksiyon Sonu

TWO'S COMPLEMENT ALGORİTMASI

Fonksiyon negatif_binary_cevir(sayı)

Adım 1. Sayının mutlak değerini al

Adım 2. Sayıyı binary'e çevir

Adım 3. Binary sayıyı 8 bit olacak şekilde düzenle

Adım 4. Tüm bitleri tersle
 $(0 \rightarrow 1, 1 \rightarrow 0)$

Adım 5. Terslenmiş binary sayıya 1 ekle

Adım 6. Elde edilen sonucu döndür

Fonksiyon Sonu

PROGRAMIN ÇALIŞMASI

Binary dönüşüm için kullanıcı 1 numaralı seçimi yapar. Daha sonra kullanıcadan dönüştüreceği sayı istenir ve sekiz bitler halinde sonuç gösterilir.

--- Çok Fonksiyonlu Taban Donusturucu ---

1 - Binary cevir

2 - Hexadecimal cevir

3 - Cikis

Seciminizi giriniz: 1

Onluk sayı giriniz: 8

Binary Sonuc: 00001000

Bellek Gorunumu: [0][0][0][0][1][0][0][0]

--- Çok Fonksiyonlu Taban Donusturucu ---

1 - Binary cevir

2 - Hexadecimal cevir

3 - Cikis

Seciminizi giriniz:

PROGRAMIN ÇALIŞMASI

Hexadecimal dönüşüm için kullanıcı 2 numaralı seçimi yapar. Daha sonra kullanıcadan dönüştüreceği sayı istenir sonuç ekrana verilir.

--- Çok Fonksiyonlu Taban Donusturucu ---

- 1 - Binary cevir
- 2 - Hexadecimal cevir
- 3 - Cikis

Seciminizi giriniz: 2

Onluk sayı giriniz: 15

Hexadecimal Sonuc: F

Binary Karsiligi: 00001111

Bellek Gorunumu: [0][0][0][0][1][1][1][1]

--- Çok Fonksiyonlu Taban Donusturucu ---

- 1 - Binary cevir
- 2 - Hexadecimal cevir
- 3 - Cikis

Seciminizi giriniz:

PROGRAMIN ÇALIŞMASI

Çıkış yapması için kullanıcının 3 numaralı seçimi yapması beklenir. Daha sonra kullanıcıya programın sonlandığına dair bir uyarı yazılır.

```
--- Cok Fonksiyonlu Taban Donusturucu ---
1 - Binary cevir
2 - Hexadecimal cevir
3 - Cikis
Seciminizi giriniz: 3
Program kapatiliyor...
```

PROGRAMIN ÇALIŞMASI

Kullanıcı eğer hatalı bir seçim girerse bunun için ekrana bir hata mesajı verilir ve yeniden seçim yapması istenir.

--- Çok Fonksiyonlu Taban Donusturucu ---

- 1 - Binary cevir
- 2 - Hexadecimal cevir
- 3 - Cikis

Seciminizi giriniz: 5
Hatali secim yaptiniz.

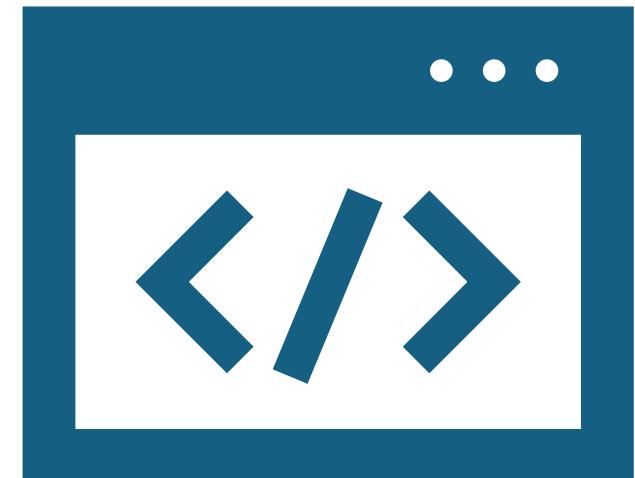
--- Çok Fonksiyonlu Taban Donusturucu ---

- 1 - Binary cevir
- 2 - Hexadecimal cevir
- 3 - Cikis

Seciminizi giriniz:

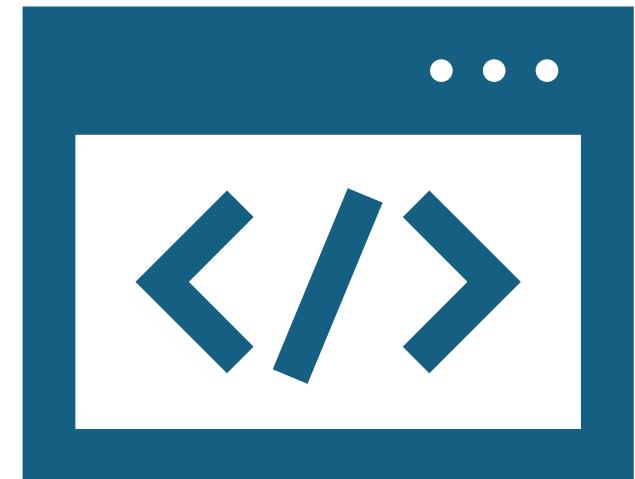
SONUÇ

- Bu çalışmada, bilgisayarlarda verinin **bit ve byte düzeyinde nasıl temsil edildiği** incelenmiştir.
- Onluk (Decimal), ikilik (Binary) ve onaltılık (Hexadecimal) sayı sistemleri arasındaki dönüşümler açıklanmıştır.
- Negatif sayıların bilgisayar belleğinde tutulabilmesi için kullanılan **Two's Complement yöntemi** detaylı olarak ele alınmıştır.
- Elde edilen sonuçların **8-bitlik bellek görünümü** ile gösterilmesi, donanım seviyesindeki veri temsilinin daha iyi anlaşılmasını sağlamıştır.



GELİŞTİRME ÖNERİLERİ

- Program, farklı bit uzunluklarını (16-bit, 32-bit gibi) destekleyecek şekilde geliştirilebilir.
- İkilikten onluya ve onaltılıktan onluya dönüşüm özellikleri eklenebilir.
- Yalnızca binary seçim için değil hexadecimal seçim yapıldıktan sonra da negatif sayı için dönüşüm yapılabilir.



KAYNAKÇA

- Brookshear, J. G., & Brylow, D. (2015). *Computer science: An overview* (12th Global ed.). Pearson Education.
- Text-Toolz. (n.d.). *Decimal to hexadecimal converter*.
<https://text-toolz.com/number-tools/decimal-to-hexadecimal>
- GeeksforGeeks. (n.d.). *Program for decimal to hexadecimal conversion*.
<https://www.geeksforgeeks.org/dsa/program-decimal-hexadecimal-conversion/>
- GeeksforGeeks. (n.d.). *How to write a pseudo code*.
<https://www.geeksforgeeks.org/dsa/how-to-write-a-pseudo-code/>
- Wikipedia. (n.d.). *Bit (bilisim)*.
[https://tr.wikipedia.org/wiki/Bit_\(bilisim\)](https://tr.wikipedia.org/wiki/Bit_(bilisim))
- Google. (2024). *Gemini Nano Banana – Yapay zekâ destekli görsel oluşturma aracı*.
<https://deepmind.google/technologies/gemini/>



**VAKİT AYIRDIĞINIZ
İÇİN TEŞEKKÜRLER :)**