

BİLGİSAYAR MÜHENDİSLİĞİNE GİRİŞ

Bölüm Yazarları:

Prof. Dr. Şaban EREN
Prof. Dr. Mesut RAZBONYALI
Prof. Dr. Halil ŞENGONCA
Prof. Dr. Ali OKATAN
Prof. Dr. Ali YAZICI
Doç. Dr. Murat ERTEM
Dr. Taner ARSAN
Doç. Dr. Nevcihan DURU
Dr. Rifat ÇÖLKESEN
Dr. A. Turan ÖZCERİT
Dr. Cengiz UĞURKAYA
Dr. M. Erhan SARIDOĞAN

Editör:

Dr. Rifat ÇÖLKESEN

PAPATYA YAYINCILIK EĞİTİM

İstanbul, Ankara, İzmir, Adana

© PAPATYA YAYINCILIK EĞİTİM - **Kasım 2011**

BİLGİSAYAR SİS. SAN. VE TİC. A.Ş.

Ankara Cad. Prof. F. Kerim Gökkay Vakfı İşhanı Giriş
No:11/3 Cağaloğlu (Fatih) / İstanbul

Tel : (212) 527 52 96 GSM: (532) 311 31 10

Faks : (212) 527 52 97

e-Posta : bilgi@papatya.gen.tr

Web : <http://www.papatya.gen.tr>

<http://www.papatya.info.tr>

Bilgisayar Mühendisliğine Giriş – Editör: Rifat ÇÖLKESEN
Bölüm Yazarlı

3. Basım Kasım 2011

Editör : Dr. Rifat ÇÖLKESEN

Türk Dili : Necdet AVCI

Üretim : Olcay KAYA

Pazarlama : Batuhan AVCI ve Ziya ÇÖLKESEN

Satış : Mustafa DEMİR

Sayfa Düzenleme : Papatya & Kelebek Tasarım

Kapak Tasarım : Papatya & Kelebek Tasarım

Basım ve Ciltleme : Pasifik Ofset Ltd. Şti. (Sertifika No:12027)- İstanbul

© Bu kitabın her türlü yayın hakkı Papatya Yayıncılık Eğitim A.Ş.'ye aittir. Yayınevinden yazılı izin alınmaksızın alıntı yapılamaz, kısmen veya tamamen hiçbir şekil ve teknikle ÇOĞALТИLAMAZ, BASILAMAZ, YAYIMLANAMAZ. Kitabin, tamamı veya bir kısmının fotokopi makinesi, ofset gibi teknikle çoğaltıması, hem çoğaltan hem de bulunduranlar için yasası bir davranıştır.

Lütfen kitabımızın fotokopi yöntemiyle çoğaltımasına engel olunuz.

Fotokopi hızsızlıktır.

Çölkesen, Rifat (Editör).

Bilgisayar Mühendisliğine Giriş / Bölüm Yazarlı – İstanbul: Papatya Yayıncılık Eğitim, 2011

xiv, 272 s. ; 24 cm.

Kaynakça ve dizin var.

Sertifika No: 11218

ISBN 978-975-6797-77-8.

1. Yazılım. 2. Donanım. 3. Algoritma. 4. Programlama Dili 5. Veri Modeli I. Title

*Bu eseri,
bilişime gönül vermiş
ve Türkiye Bilişim Vakfı çatısında
Bilgisayar Mühendisliği Bölüm Başkanları
toplantıları düzenleyerek, Bilgisayar Mühendisliği'ne
katkıları olmuş rahmetli Tuncer Üney'e
ve Türk Gençliğine
armağan ediyoruz.*

Bu kitapta referans olarak kullanılan ve kaynakçada verilen birçok eser için yazılı izin alınmıştır.

İÇİNDEKİLER

ÖNSÖZ	11
BİLGİSAYAR ÖLÇÜ BİRİMLERİ	13
Bölüm 1. BAŞLARKEN; TEMEL KAVRAMLAR	15
<i>Doç. Dr. Murat ERTEN</i>	
1.1. Bilgisayar Mühendisliği Eğitimi ve Görevleri	15
1.2. Bilgisayar Mühendisliği Konuları	16
1.3. Bilgisayar Donanımı	18
1.3.1. Merkezi İşlem Birimi	18
1.3.2. Bellek	19
1.3.3. Giriş Çıkış Birimleri	21
1.3.4. Bilgisayar Ağları	21
1.4. Bilgisayar Yazılımı	23
1.4.1. Yazılım	23
1.4.2. İşletim Sistemi	25
1.4.3. Sayı Sistemleri	26
1.4.4. Algoritma	27
1.4.5. Veri Yapıları	28
1.4.6. Programlama Dilleri	29
1.4.7. Veritabanları	30
1.4.8. Veri Güvenliği	30
1.5. Özet	31
1.6. Çalışma Soruları	32
Bölüm 2. BOOLE CEBİRİ	33
<i>Prof. Dr. Şaban EREN</i>	
2.1. Boole Cebrinin Esasları	34
2.1.1. Boole Cebri Teoremleri ve Örnekler	34
2.1.2. Doğruluk Tabloları ve Boole Cebrinde İşlemi Basitleştirme	35
2.2. Lojik Kapılar ve Doğruluk Tabloları	37
2.2.1. VE Kapısı (AND)	37
2.2.2. VEYA Kapısı (OR)	38
2.2.3. DEĞİL Kapısı (NOT)	38
2.2.4. VE DEĞİL Kapısı (NAND)	39
2.2.5. VEYA DEĞİL Kapısı (NOR)	39
2.2.6. YA DA Kapısı (XOR)	39

2.2.7. YA DA DEĞİL Kapısı (XNOR)	40
2.2.8. TAMPON Kapısı (BUFFER)	40
2.2.9. Lojik Kapı Örnekleri	40
2.2.9.1. Lojik Diyagramdan Matematiksel İfadedenin Elde Edilmesi	40
2.2.9.2. Matematiksel Eşitlikten Lojik Diyagramın Elde Edilmesi	43
2.3. Lojik İfadelerin Sadeleştirilmesi/İndirgenmesi	44
2.3.1. Tanımlar	44
2.3.1.1. Minumum Terimler (Minterm)	45
2.3.1.2. Maksimum Terimler (Maksterm)	46
2.3.2. Karnaugh Diyagramları (Haritalar)	47
2.3.3. Üç Değişkenli Karnaugh Diyagramları	51
2.3.4. Karnaugh Diyagram Örnekleri	52
2.4. Özet	55
2.5. Çalışma Soruları	55
Bölüm 3. ALGORİTMA VE AKIŞ ŞEMALARI	57
<i>Dr. Taner ARSAN ve Rıfat ÇÖLKESEN</i>	
3.1. Algoritmada Olması Gereken Özellikler	58
3.2. Algoritma Tasarımı	59
3.3. Akış Şemaları	63
3.3.1. Akış Şeması Simgeleri	63
3.3.2. Tipik Akış Şeması Parçaları	66
3.4. Akış Şeması Örnekleri	72
3.5. Akış Şeması Dışında Şekilsel Tasarım	75
3.6. Özet	76
3.7. Çalışma Soruları	76
Bölüm 4. PROGRAMLAMA DİLLERİ	77
<i>Doç. Dr. Nevcihan DURU</i>	
4.1. Yazılım Geliştirme Süreci	78
4.2. Yazılım Geliştirme Sürecinde Programlama Dilinin Önemi	78
4.3. Programlama Dillerinin Tarihçesi ve Sınıflandırılması	80
4.3.1. Programlama Dillerinin Tarihçesi	80
4.3.2. Programlama Dillerinin Sınıflandırılması	83
4.3.3. Nesneye Yönelik Programlama Kavramı	85
4.4. Programlama Ortamı	86
4.5. Programlama Dillerinin Elemanları	87
4.6. Özet	91
4.7. Çalışma Soruları	91

Bölüm 5. İŞLETİM SİSTEMLERİ	93
<i>Prof. Dr. Mesut RAZBONYALI</i>	
5.1. Bilinen İşletim Sistemleri	94
5.1.1 Windows Ailesi	94
5.1.2. Linux Ailesi	94
5.1.3. UNIX İşletim Sistemi	95
5.1.4. VM İşletim Sistemi	96
5.2. İşletim Sisteminin Görevleri	96
5.3. İşletim Sistemlerinin Gelişim Evresi	97
5.4. İşletim Sistemi Türleri	99
5.5. İşletim Sistemi Yapısı/Mimarisi	102
5.5.1. Bilgisayar Sistemi Kaynakları	103
5.5.2. Çekirdek Sistem	104
5.6. Prosesler ve Proses Yönetimi	106
5.7. Bellek Yönetimi	108
5.8. Özet	109
5.9. Çalışma Soruları	110
Bölüm 6. MİKROİŞLEMÇİLER VE ASSEMBLY PROGRAMLAMA	111
<i>Dr. A. Turan ÖZCERİT</i>	
6.1. Bilgisayarların Evreleri	112
6.2. Mikroişlemcilere Genel Bakış	113
6.2.1. Temel Mikroişlemci Mimari Kavramları	116
6.2.2. Mikroişlemcilere Donanımsal Bakış	119
6.2.3. Assembly Dili ve Özellikleri	123
6.2.4. Assembly Dilinde Adresleme Yöntemleri	126
6.2.5. Assembly Diliyle Programlama Örnekleri	129
6.3. Özet	132
6.4. Çalışma Soruları	133
Bölüm 7. BİLGİSAYAR AĞLARI VE İNTERNET	135
<i>Dr. Cengiz UĞURKAYA</i>	
7.1. Komple Ağın Parametreleri	136
7.2. Band Genişliği ve İletişim Hızı	138
7.3. OSI Başvuru Modeli	139
7.4. Ağ Cihazları	140
7.5. Kablolama ve Kablosuz Bağlantı	144
7.6. Ağ Üzerine Çeşitli Kavramlar	146
7.7. Ağ Teknolojileri	148
7.7.1. Ethernet ve Türevleri	148
7.7.2. Jetonlu Halka ve Türevleri	149

7.7.3. Bazi WAN ve Uzak Bağlantı Teknolojileri	150
7.8. Internet ve TCP/IP	151
7.8.1. TCP/IP Protokol Kümesi	152
7.8.2. Dört-Katmanlı İnternet Mimarisi	153
7.8.3. İnternet Adresleri ve DNS	153
7.8.4. Yeni Nesil Yönlendirme Protokolü / IPv6	155
7.9. Özet	155
7.10. Çalışma Soruları	156
Bölüm 8. SAYI VE KODLAMA SİSTEMLERİ	157
<i>Prof. Dr. Halil ŞENGONCA</i>	
8.1. Sayı Sistemleri	158
8.1.1. On Tabanlı Sayı Sistemi	158
8.1.2. B Tabanlı Sayı Sistemi	159
8.1.3. İkili Sayı Sistemi	159
8.1.4. Sekizli Sayı Sistemi	162
8.1.5. Onaltılı Sayı Sistemi	163
8.1.6. Tabanlar Arası Dönüşürmeler	164
8.2. Bilgisayarda Tam Sayıların Sunumu	166
8.3. Kayan Noktalı Sayıların Sunumu	167
8.4. ABeCesel (Alfabetic) ve Diğer Karakterlerin Sunumu	168
8.4.1. Kodlama Sistemleri	168
8.5. Veri Sıkıştırma	172
8.5.1. Kayıpsız Sıkıştırma	172
8.5.2. Kayıplı Sıkıştırma	172
8.6. Şifreleme ve Şifre Çözme	172
8.6.1. Şifreleme Terimleri	172
8.7. Özet	175
8.8. Çalışma Soruları	175
Bölüm 9. YAZILIM MÜHENDİSLİĞİ	177
<i>Dr. Erhan SARIDOĞAN</i>	
9.1. Sistem ve Yazılım	178
9.1.1. Bilgisayar Sistemi Mühendisliği	178
9.1.2. Sistem Geliştirme Süreci	178
9.1.3. Yazılım Mühendisliği	180
9.2. Yazılım Mühendisliği Yöntembilimleri	180
9.2.1. Yazılım Geliştirme Standartları	180
9.2.2. Yazılım Geliştirme Modelleri	181
9.3. Yazılım Geliştirme Süreci	185
9.3.1. Yazılım İsterleri Çözümlemesi	185
9.3.2. Yazılım Tasarımı	188

9.3.3. Yazılım Gerçekleştirimi	190
9.3.4. Yazılım Testi	191
9.3.5. Yazılım Bakımı	193
9.4. Yardımcı Süreçler	193
9.4.1. Yazılım Nitelik Güvence	193
9.4.2. Yazılım Düzenleşim Yönetimi	194
9.4.3. Yazılım Proje Yönetimi	195
9.5. Özet	195
9.6. Çalışma Soruları	196
Bölüm 10. VERİ YAPILARI VE VERİ MODELLERİ	197
<i>Dr. Rifat ÇÖLKESEN</i>	
10.1. Veri/Veri Yapısı ve Bilgi	198
10.2. Temel Veri Yapıları	199
10.2.1. Karakter	199
10.2.2. Tamsayı	201
10.2.3. Kesirli/Gerçel Sayı	204
10.2.4. Sözcü/Sözcük	205
10.2.5. Dizi/Matris	206
10.3. Tanımlamalı Veri Yapıları	207
10.3.1. Topluluk Oluşturma	207
10.3.2. Ortaklık Oluşturma	208
10.4. Veri Modeli ve Türleri	209
10.4.1. Bağlantılı Liste Veri Modeli	209
10.4.2. Ağaç Veri Modeli	210
10.4.3. Graf Veri Modeli	211
10.4.4. Durum Makinası Veri Modeli	212
10.4.5. Veritabanı İlişkisel Veri Modeli	213
10.4.6. Ağ Veri Modeli	213
10.5. Özet	214
10.6. Çalışma Soruları	215
Bölüm 11. VERİTABANI YÖNETİMİ VE SQL	217
<i>Prof. Dr. Ali YAZICI</i>	
11.1. Veritabanı Yönetim Sistemleri	218
11.2. İlişkisel Veritabanı ve İlişkisel Model	218
11.3. VT Tasarım Adımları	219
11.3.1. Gereksinim Analizi	220
11.3.2. Kavramsal Tasarım	221
11.3.3. Mantıksal Tasarım	223
11.3.4. Sistemin Uyarlanması	224

11.3.5. Sistemin Test Edilmesi ve Bakımı	224
11.4. SQL	224
11.4.1. Tablo Yaratmak	226
11.4.2. Tablolara Veri Girişi	228
11.4.3. Sorulama	230
11.5. Özet	234
11.6. Çalışma Soruları	234
Bölüm 12. DONANIM MÜHENDİSLİĞİ	237
<i>Prof. Dr. Ali OKATAN</i>	
12.1. Saklayıcı Aktarım Dili	238
12.2. Donanım Tanımlama Dilleri	239
12.3. Donanım Benzetim ve Tasarım Yazılımı	242
12.4. Benzetim Ortamları	243
12.5.1. Örnek Bir Mikroişlemci Tasarımı	244
12.5. Algoritmik Yaklaşımla Örnek Donanım Tasarımı	245
12.6. VLSI Tasarımı	250
12.7. Özet	251
12.8. Çalışma Soruları	251
KAYNAKÇA	253
Ek-A: EMO (Elektrik [Elektronik/Bilgisayar]) Mühendisliği Odası	255
Ek-A: IEEE (Elektrik ve Elektronik (Bilgisayar) Mühendisleri Enstitüsü	257
DİZİN	267

ÖNSÖZ

Kendini yenileyebilmenin ve gelişmenin temel aracı olan araştırma/geliştirme, aynı zamanda kaliteli eğitim/öğretim vermenin de ilk koşuludur. Dolayısıyla, üniversite öğretimde dersler, konularda araştırma/geliştirme yapan ve birikimlerini, elde ettiği sonuçları yazıp yayımlayan “alanıçı” öğretim üyelerince verilmelidir. Bu kitapta Bilgisayar Mühendisliği, Bilgisayar Bilimleri, Endüstri Mühendisliği, Elektrik Elektronik Mühendisliği Bölümleri için bilgisayar mühendisliği konuları hakkında referans niteliğindedir. Aynı zamanda, Bilgisayar Mühendisliği kavramlarına ve konularına hakim olmak, temel bilgilerini belirli referansa oturtmak isteyen meslek yüksek okullarının ilgili bölümleri için de bir başvuru kaynağıdır.

Kitap olayı, “*bir aşk işi sanırım*”; herbiri konusunda uzman ve deneyimli olan ve farklı üniversitelerden hocalarımızın biraraya gelip belki de uzun yıllar gündemde kalacak, gençliğin yararlanacağı böylesi bir eser ortaya çıkartmaları, çok yoğun olmalarına karşın, zaman ayırmaları, sanırım yazma aşkı, kitap aşkı ve gençliğe yararlı olmadan başka birşey olmasa gerek. Böyle bir eserin “Bilgisayar Mühendisliğine Giriş” kitabının ilgili mühendislik bölümlerine ve nitelikli eğitim veren meslek yüksek okullarının ilgili bölümlerine, meslekleri konusunda referans noktası oluşturacağına yürekten inanıyorum.

Bu kitabın amacı, “Bilgisayar Mühendisliği” kapsamına giren konulara ait temel bilgileri vermek ve bireye öğretim boyunca verilen dersler hakkında temel bilgileri/kavramları önceden kazandırmaktır. Böylece Bilgisayar Mühendisliği veya Bilimleri konusunda öğretim gören bireyler, hem meslekleri konusunda ön bilgiler ile donatılmış olacak hem de öğretim sürecinde hangi derste hangi konuların anlatılacağı hakkında bilgileri olacaktır.

Kitabın hazırlanması sırasında, “Algoritma ve Akış Şemaları” bölümünü yazan sayın Prof. Dr. Yahya Karslıgil hocamız, maâlesef kitabı hazırlaması sürecinde rahatsızlandı; ve hocamızın bölümünü bu baskiya da koyamadık; hocamıza acil şifalar diliyoruz ve sonraki baskılarda bölümünü ekleyeceğimizi umuyorum.

Bölümleri farklı yazarlar tarafından hazırlanan bu eserde çok dikkatli olunmaya çalışıldı; terim birlaklılığı ve konu bütünlüğü olabildiğince arttırıldı. Ayrıca gereksiz tekrarlar en aza indirildi. İlerideki baskılarda bölüm eklenebilir, çıkartılabilir veya güncellenebilir. Bu konuda yapıçı eleştirilerinizi bekliyoruz... Sevgilerimle.

Dr. Rifat ÇÖLKESEN

BİLGİSAYAR ÖLÇÜ BİRİMLERİ

Veri, Hız ve Ölçü Birimleri

Bu sayfayı hazırlayan Ege KİPMAN'a teşekkür ederiz.

Veri Birimleri

<i>Bit – b</i>	: 0 veya 1 değerini alabilen en küçük veri birimi.
<i>Nibble - N</i>	: 4-bit'in yan yana gelmesiyle oluşan veri birimi.
<i>Byte - B</i>	: 8-bit'in yan yana gelmesiyle oluşan veri birimi.
<i>Word</i>	: 32-bit'in yan yana gelmesiyle oluşan veri birimi.
<i>Double Word</i>	: 64-bit'in yan yana gelmesiyle oluşan veri birimi.
<i>Quad Word</i>	: 128-bit'in yan yana gelmesiyle oluşan veri birimi.
<i>Kilobyte* – KB</i>	: 1024-Byte'in yan yana gelmesiyle oluşan veri birimi.
<i>Megabyte* – MB</i>	: 1024 Kilobyte'in yan yana gelmesiyle oluşan veri birimi.
<i>Gigabyte *– GB</i>	: 1024 Megabyte'in yan yana gelmesiyle oluşan veri birimi.
<i>Terabyte *– TB</i>	: 1024 Gigabyte'in yan yana gelmesiyle oluşan veri birimi.
<i>Petabyte* – PB</i>	: 1024-terabyte'in yan yana gelmesiyle oluşan veri birimi.
<i>Exabyte* – EB</i>	: 1024-Petabyte'in yan yana gelmesiyle oluşan veri birimi.
<i>Zettabyte* – ZB</i>	: 1024-Exabyte'in yan yana gelmesiyle oluşan veri birimi.
<i>Yottabyte* – YB</i>	: 1024-Zettabyte'in yan yana gelmesiyle oluşan veri birimi.

Hız Birimleri

<i>Hertz - Hz</i>	: Bir saniyedeki devir sayısı.
<i>Kilohertz* - kHz</i>	: Saniyedeki 1.000 Hertz' ten oluşan hız birimi.
<i>Megahertz* - MHz</i>	: Saniyedeki 1.000 kHz'ten oluşan hız birimi.
<i>Gigahertz* - GHz</i>	: Saniyedeki 1.000 MHz' ten oluşan hız birimi.
<i>Floating-point Operations Per Second – Flops</i>	: Saniyede yapıla reel sayılı işlem sayısı.
<i>MegaFlops* – Mflops</i>	: Saniyedeki 1.000.000 Flops'tan oluşan hız birimi.
<i>GigaFlops* – Gflops</i>	: Saniyedeki 1.000 MFlops'tan oluşan hız birimi.
<i>Bits per second – bps</i>	: Saniyede aktarılan bit sayısı.
<i>Bytes per second – Bps</i>	: Saniyede aktarılan Byte sayısı.
<i>Millions Instructions Per Second - MIPS</i>	: Saniyedeki komut sayısı (milyon olarak).
<i>Revolutions per minute - rpm</i>	: Dakikadaki devir sayısı.
<i>Refresh rate</i>	: Ekranda saniyedeki görüntü yenileme hızı.
<i>Bit rate</i>	: İletilen bit sayısı/bit oranı.

Ölçü Birimleri

Inch : Bir inch 2,54 cm'e eşittir.

Dots Per Inch (dpi) : *Inch* başına düşen nokta sayısı.

Pixel : Noktacık; ekrandaki en küçük nokta.

Resolution : Çözünürlülük; Yazıcılarda *inch* başına düşen nokta sayısı (Örneğin 600 dpi); ekranda ise yatay ve düşey eksendeki toplam noktacık sayısı (Ör: 640 x 480)

Dot Pitch : İki pixel/noktacık arasındaki uzaklık.

Ön-Ek Tablosu

Ön Ek	Simge	Onlu taban	İkili taban
yocto-	y	10^{-24}	-
zepto-	z	10^{-21}	-
atto-	a	10^{-18}	-
femto-	f	10^{-15}	-
pico-	p	10^{-12}	-
nano-	n	10^{-9}	-
micro-	μ	10^{-6}	-
mili-	m	10^{-3}	-
centi-	c	10^{-2}	-
deci-	d	10^{-1}	-
deka-	D	10^1	-
hecto-	h	10^2	-
*kilo-	k veya K	10^3	2^{10}
*mega-	M	10^6	2^{20}
*giga-	G	10^9	2^{30}
*tera-	T	10^{12}	2^{40}
*peta-	P	10^{15}	2^{50}
*exa-	E	10^{18}	2^{60}
*zetta-	Z	10^{21}	2^{70}
*yotta-	Y	10^{24}	2^{80}

BİLGİSAYAR MÜHENDİSLİĞİNE GİRİŞ I

Bilgisayar Mühendisliği Nedir?

- Bilgisayar mühendisliği;
 - bilgisayar sistemlerinin yapısı,
 - tasarıımı,
 - geliştirilmesi ve
 - bu sistemlerin kullanımlarını içeren bir mühendislik dalıdır.

Bilgisayar Mühendisi Ne İş Yapar?

- Bilgisayar Mühendisi kavramı artık daha genel bir alanı nitelemektedir.
- Teknolojinin gelişmesiyle de farklı uzmanlık alanları ortaya çıkmıştır.
- Genel olarak aşağıdaki başlıklar halinde incelenebilir.
 1. Yazılım
 2. Veri tabanı
 3. Bilişim güvenliği
 4. Bilgisayar ağları
 5. Donanım/Elektronik
 6. Web tasarıımı ve programcılığı
 7. Çoklu ortam (Multimedya)
 8. Teknik destek

...

1- YAZILIM

- Bilgisayar mühendislerinin en bilinen çalışma alanlarından birisi yazılım sektörüdür.
- Yazılım geliştirme işi genelde adına “**Yazılım yaşam döngüsü**” denilen bir süreç içerisinde yürürl.
- Bilgisayar mühendisleri;
 - planlama,
 - analiz,
 - tasarım,
 - kodlama,
 - test/kalite yönetimi ve
 - kurulum gibi aşamalardan oluşan bu sürecin her tarafında yer alabilirler.

Yazılım - İş Tanımları (1)

- **Planlama/Analiz Aşaması,**
 - İş Analisti
 - Müşterilerin ihtiyacını anlamaya yönelik çalışmalar yapan pozisyondur.
 - Sistem Analisti
 - Sistem analisti genel olarak ise tüm bir bilgi işlem sistemini tasarlayıp ihtiyaca göre çözümler öneren kişidir.
- **Tasarım Aşaması,**
 - Yazılım Tasarım Uzmanı
 - Yazılımın altyapı bileşenleri dahil tüm yazılım yapısını planlayan kişidir.
 - Yazılım Modelleme Uzmanı
 - UML denilen modelleme ve diyagram çizme dilini kullanarak yazılımın işleyişini tanımlar.

Yazılım - İş Tanımları (2)

- **Geliştirme Aşaması,**
 - Programcı / Geliştirici / Yazılım Uzmanı
 - Programlama dillerini ve teknolojilerini kullanarak ürün geliştiren kişidir.
 - Yazılım Mühendisi
 - Bir yazılım mühendisi yazılımla ilgili tüm aşamalarda bulunabilecek özelliklere sahip kişidir.
 - Yazılım Mimarı / Yazılım Danışmanı
 - Projede kullanılacak teknolojileri seçerler, proje metodolojisi ve süreçlerinde tavsiyelerde bulunurlar, uygulamanın genel tasarım ve yapısını oluşturur ve korurlar, projenin doğru tanımlanmasını sağlarlar, tasarımın dökümantasyonu edilmesi ve kodlama standartlarının belirlenmesi ile uğraşırlar.
 - Ekip Lideri / Proje Yöneticisi
 - Proje Lideri/Yöneticisi yazılım ekibini bir arada tutan ve zaman çizelgelerine uyulması için gerekli motivasyonu sağlayan yöneticidir.

Yazılım - İş Tanımları (3)

- **Test/Kalite Güvence Aşaması**, yazılım projesinin standartlara uygunluğunu ve ihtiyaçları karşıladığı denetleyen çalışmalar demektir.
 - Yazılım Kalite Uzmanı
 - İhtiyaçların ve geliştirilen çözümün doğru belirlenip belirlenmediğini, yazılımin belirli standartlarda olup olmadığını denetleyen kişidir.
 - Test Uzmanı/Test Mühendisi
 - Yazılım hatalarını tespit etme, hataların sebepleri ve sonuçları ile ilgili süreci yönetme işlerini yürütür.
 - Test Yöneticisi
 - Yazılımın testi geçip geçemediği, kabul edilmeye hazır olup olmadığı kararlarını verir.

2- VERİ VE VERİTABANI YÖNETİMİ

- İş hayatında bilgisayarlaşmanın artmasıyla çok miktarda veri depolanmaya başlamıştır..
- Artık bilgileri kaydetmekten çok bu verilerin analiz edilmesi ve işlenmesi önem kazanmaya başlamıştır.
- Bilgisayar mühendisleri verilerin ve içinde tutuldukları veritabanlarının yönetimi ile ilgili bir çok pozisyonda yer almaktadırlar.

İş Tanımları (1)

- Veri Yöneticisi
 - Veri yöneticisi veriyi kullanıcılar tarafından kullanılmaya hazır hale getirir. Bunun için daha çok analiz etme, veri toplama ve dizayn aşamalarında yer alır.
- Veri Analisti
 - Veri ambarı üzerinde çok miktarda veri üzerinde çok karmaşık sorgulama ve analizleri gerçekleştirerek sonuçlar üretir.
- Veri Modelleme Uzmanı
 - Veri modelleme uzmanı; bir taraftan, verilerin şekil ve metin olarak ifade eden modelleri kurarken, bir taraftan da gereken veritabanı tablo tasarımlarını yapar.

İş Tanımları (2)

- Veri Madencisi
 - Aslında veri analistliğine benzeyen bu pozisyon, anlamsız görünen verilerden belli metodolojiler kullanarak anlamlı sonuçlar ve istatistikler çıkarır.
- Veritabanı Yöneticisi
 - Veritabanı basit olarak bilgi depolayan, bu bilgiyi verimli ve hızlı bir şekilde yönetip değiştirebilen bir yazılımdır.
 - Veritabanı yöneticisinin görevleri kısaca şu başlıklar altında toplanabilir:
 - Veritabanı yazılımının Kurulması ve versiyon takibi,
 - Veritabanı oluşturulması,
 - Kullanıcı yönetimi,
 - Yedekleme,
 - Performans ayarları vb.

İş Tanımları (3)

- Veritabanı Mimarı
 - Yeni bir veritabanının tasarılanması ve oluşturulması görevlerini yerine getirir.
 - Veritabanı mimarı, sadece yeni tasarım ve geliştirme işine bakar.
- Veritabanı Güvenlik Uzmanı
 - Veritabanı güvenlik uzmanı sistem güvenliği ve veri güvenliği denilen iki işi yapar.

3- BİLİŞİM GÜVENLİĞİ

- Baş döndürücü hızla gelişen yeni teknolojiler, beraberinde güvenlik tehlikelerinin de hızla ortaya çıkması sonucunu getirmektedir.
- Şirketler gittikçe çoğalan sayısal verilerini korumak istedikleri için bilişim güvenliği de başlı başına bir sektör haline dönüşmüştür.
- Güvenlik uzmanları bugün iş ilanlarında en çok aranan kişilerdir.

İş Tanımları (1)

- **Bilişim Güvenliği Uzmanı / Yöneticisi**
 - Şirketlerin güvenlik altyapısını tasarlayan, kuran, yöneten ve geliştiren kişilerdir.
- **Ağ ve Internet Güvenlik Uzmanı**
 - Var olan tehditleri sürekli olarak takip ve analiz etme, bu tehditlere karşı alınabilecek önlemleri inceleme, gerekli önlemleri devreye alma, önlemlerin düzgün çalıştığını sürekli olarak kontrol etme işlerini gerçekleştirir.
- **Güvenlik Danışmanı**
 - Uzman oldukları alanlara göre şirketlerin güvenlik konusundaki bilgi ihtiyacını karşılar ve gerekli çözümleri önerir ya da üretirler.

4- AĞ YÖNETİMİ

- Bilgisayar mühendisleri ağ kurulumu ve yönetiminin tüm aşamalarında görev alarak işlerin düzenli yürümesini sağlarlar.
- Ağ bir şirket içi ağ olabileceği gibi dışarıya açık bir ağ da olabilir.
- Ağ yönetimi konusunda bir çok iş alanı mevcuttur.

İş Tanımları (1)

- Ağ Analisti
 - İhtiyaçları iyi anlayıp ağ altyapısının tasarımını yapar, kullanılacak donanım ve ağ trafigini en iyi performansla sağlayacak ekipmanı belirler, kurulumun her aşamasında rol alır.
- Ağ Yöneticisi / Sistem Yöneticisi
 - Şirket gereksinimlerine göre network tasarıımı ve yönetimi ve işletim sistemlerinin tasarıımı, kurulumu ve yönetimi konularından sorumludur.
 - Ayrıca yedekleme gibi günlük işlemlerin yönetimi, kullanıcı gereksinimlerini karşılanması ve işlemleri içerir.

İş Tanımları (2)

- Ağ Mühendisi
 - Daha çok fiziksel ağ işleriyle uğraşır.
 - Kablo sistemleri, yerel ve geniş ağ bağlantıları, router vb. ekipmanların kurulumu ve ayarları, yönetim yazılımları, trafik izleme konularında çalışır.
- Ağ Destek Uzmanı
 - Ağ ile ilgili kurulum ve kullanım sorunlarına destek verir.
 - Kablolama, ağ ekipmanları, ağ kurulumu ve bakımı gibi görevlerde çalışır.

5- DONANIM VE ELEKTRONİK

- Bilgisayar mühendisliğinin en çok bilinen bir alanı yazılımsa diğeri de donanımdır.
- Son yıllarda yazılım ve donanım bütünlüğmesine dayanan teknolojiler (VOIP, Embedded Sistemler, Robotik) büyük ivme kazanmıştır.
- Genellikle Elektronik Mühendisliğinin işi gibi görünse de, Bilgisayar Mühendislerinin de donanım alanının alt kollarında uzmanlaşması ve iş bulması mümkündür.

İş Tanımları (1)

- Donanım Teknisyeni/Mühendisi
 - Çip teknolojileri, devre ve kart tasarımları, donanım parçalarının tüm üretim aşamaları, bilgisayar montajı, modem, yazıcı, monitör vb. teknolojiler bu pozisyonun ilgi alanıdır.
- Kontrol Mühendisi
 - Elektrik, elektronik, mekanik ve bilgisayar tabanlı tüm endüstriyel üretim sistemlerinin amaçlanan ve planlanan biçimde çalışmasını sağlar. Otomatik kontrol teorisi ve uygulamaları, endüstriyel otomasyon, ölçme, bilgisayar tabanlı endüstriyel bilişim sistemlerinin tasarımı ve uygulamaları konularında çalışır.

İş Tanımları (2)

- Robotik Mühendisi
 - Robot tasarılar, üretim ve geliştirmelerini yapar, çalışmaları için gerekli yazılım uygulamalarını geliştirir.
- Telekom Mühendisi
 - İletişim alanındaki teknolojileri tasarlayan ve uygulayan kişidir. Mesela cep telefonu santralleri, kablosuz ağ sistemleri, İnternet altyapı sistemleri çalışma alanlarıdır.
- Sistem Tasarımcısı
 - Donanım anlamında sistem tasarımcısı, küçük bir cihazın tasarımından komple bilgi işlem sistemlerinin tasarımına kadar geniş bir yelpazede iş yapan kişidir.

6- WEB / INTERNET

- 90'lı yıllarda itibaren Internet'in hayatımızın her noktasında yer almasının bir sonucu olarak web siteleri ile ilgili işler bilişim mesleklerinin önemli bir alanı haline gelmiştir.

İş Tanımları (1)

- İçerik Yöneticisi
 - Bir web sitesinin içeriğinden sorumlu kişidir.
 - Sitedeki yazılı, sesli ya da görüntülü içeriğin planlanması, eklenmesi, kontrolü ve yönetimi ile uğraşır.
- Web Tasarımcısı
 - Web tasarımcıları, bir web sitesinin görünümünden, tasarımından ve kullanılabiliğinden (usability) sorumlu kişilerdir.
 - Web tasarımcısı olmak hem grafik, hem multimedya hem de programlama konusunda bilgi sahibi olmak demektir.

İş Tanımları (2)

- Web Geliştiricisi
 - Özellikle web tabanlı uygulamalar geliştiren yazılım mühendisleridir. (HTML, XML, ASP, JSP, PHP)
- Webmaster/Web Sitesi Yöneticisi
 - Web siteleri için alan adı almak, hosting(yer) kiralamak, hazırlanmış projeleri yüklemek, çalışır hale getirmek, arama motorlarına kaydettirmek, yönetmek, hatalarıyla uğraşmak, kullanım istatistiklerini inceleyerek gerekli iyileştirmeleri yapmak, yedek almak, bakım ve versiyon güncellemeleri yapmak gibi işleri yaparlar.

7- Çoklu Ortam

- Multimedya ya da diğer bir deyişle çoklu ortam, grafik yanında ses ve video üzerinde çalışmaları da içermektedir.
- 2 ve 3 boyutlu (2D/3D) bilgisayar grafikleri, hem oyunlarda hem de animasyon işlerinde kullanılmaktadır.

İş Tanımları (1)

- 2D/3D Grafik Uzmanı
 - Web siteleri, tanıtım CD'leri, oyunlar, eğlence amaçlı yazılımlar vb. teknoloji ürünlerinde hoşumuza giden resim ve tasarımlar bu uzmanların elinden çıkmaktadır. (Photoshop, AfterEffects, 3D Studio Max)
- Multimedya/Grafik Tasarımcısı
 - Multimedya tasarımcıları hem grafik hem de animasyon konusunda bilgi sahibidir. Ses, görüntü, animasyon, çizim, fotoğraf gibi malzemeleri kullanarak prodüksiyonlar yaparlar.
- Eğitim Tasarımcısı
 - Bilişsel eğitim dediğimiz çok yönlü öğrenme tekniklerini kullanan materyaller tasarlar. Dolayısıyla, e-öğrenme, Internet, multimedya alanlarında kendilerini yetiştirmeleri gereklidir.

8- TEKNİK DESTEK



İş Tanımları (1)

- Teknik Destek Uzmanı
 - Bir ürün ya da teknoloji konusunda şirket personeline veya müşterilere destek hizmeti veren kişilerdir.
- Yardım Masası Uzmanı
 - Müşterilere uzaktan destek verilmesi amacıyla kurulan çağrı merkezleri ve yardım masası bölgelerinde çalışan uzmanlar genelde sesli ya da yazılı destek verirler.
- 1./2./3. Seviye Teknik Destek Mühendisi
 - Bilgisayar mühendisleri bilgi ve tecrübelere bağlı olarak bu üç seviyede de çalışabilir.

9- DİĞER ÇALIŞMA ALANLARI:

- **Eğitmen/Akademik Kariyer**
 - Bilgisayar mühendisleri eğitimcilik kabiliyetleri varsa özel şirketlerde, eğitim merkezlerinde eğitim verebilirler.
 - Ya da akademik kariyer yapmayı tercih edip üniversitede kalabilirler.
- **Dokümantasyon Uzmanı/Teknik Yazar**
 - Yazılım/Donanım ürünleri ile ilgili kullanım kılavuzu, programlama kılavuzu vb. dokümanları hazırlar
- **E-ticaret uzmanı**
 - Bilgisayar mühendisleri de e-ticaret sistemlerinin hem altyapı hem de işleyiş kısımlarında rol alırlar.
- **CIO**
 - CIO(Chief Information Officer) ya da Türkçesiyle Bilişim Direktörü, üst düzey bir yönetici pozisyonudur. CIO'lar şirketlerin bilişim vizyonunu belirler, teknoloji bütçelerini yönetir, stratejik teknoloji yatırım kararlarını verir ve yapılan yatırımların getirisini ölçer.

BİLGİSAYAR MÜHENDİSLİĞİNE GİRİŞ

İçerik

- Bilgisayarda kullanılan veri birimleri
- Bilgisayar Hız birimleri
- Boole Cebri

Bilgisayar Veri Birimleri

Veri Birimi Adı	Ölçüsü
Bit	0 veya 1 olabilen bir veri
Nibble	4 bit'in yan yana gelmesi ile oluşan veri
Byte	$8 \text{ bit} = 2^3$
Word	$32 \text{ bit} = 2^5$
Double word	$64 \text{ bit} = 2^6$
Quad Word	$128 \text{ Bit} = 2^7$
.....
KiloByte(KB)	$1024 \text{ Byte} = 2^{10}$
Megabyte(MB)	$1024 \text{ KB} = 2^{20}$
GigaByte(GB)	$1024 \text{ M} = 2^{30}$
TeraByte(TB)	$1024 \text{ G} = 2^{40}$
Peta(PB)	$1024 \text{ G} = 2^{50}$
Exa(EB)	$1024 \text{ P} = 2^{60}$

Bilgisayar Hız Birimleri

Boole Cebri

- İkili sayı sistemi için geliştirilmiş cebirdir.
- İkili sayı sistemi 0 ve 1 rakamını kullanır.
- İkili sayı sisteminde her bir basamak bit olarak adlandırılır.
- En büyük hane MOST SIGNIFICANT BIT(MSB)
- En küçük hane LEAST SIGNIFICANT BIT(LSB) olarak adlandırılır.

Sayı Sistemleri Arasındaki İlişki

Onluk Taban (Decimal Base)	İkilik Taban (Binary Base)	Sekizlik Taban (Octal Base)	Onaltılık taban (Hexadecimal Base)
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A

İkililik Taban

- 10'luk taban 0,1,..9 rakamlarını kullanan ve insanoğlunun en kolay işlem yaptığı sayı sistemidir.
- $123=1\times100+2\times10+3\times1$
- İkililik taban ise bilgisayarlı sistemler için en uygun sayı sistemidir. 0,1 rakamlarını kullanır.
- $110=1\times4+1\times2+0\times1=6$

İkililik tabanda basamaklar ve 10 tabanına dönüşüm

sayı	1	0	1	1	1	0	0	1	1
Basamakları	256'lar	128'ler	64'ler	32'ler	16'lar	8'ler	4'ler	2'ler	Birler basamağı
10'luk karşılığı	256		64	32	16			2	1

Sayı= $256+64+32+16+2+1=371$ dir.

10'luk tabandan ikilik tabana dönüşüm

Örnek 1: 49 sayısının 2'lik karşılığını bulmaya çalışalım.

- En büyük 32'ler basamağı vardır.
- $49 = 32 + 16 + 1 = (110001)_2$

Örnek 2: 179 sayısının 2'lik karşılığı?

En fazla 128 var.

$$179 = 1 \times 128 + 0 \times 64 + 1 \times 32 + 1 \times 16 + 0 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1 = (10110011)_2$$

$$128 + 32 + 16 + 3 = 179$$

Boole Cebri

- Computer(=Bilgisayar), bilginin işlenmesi , iletilmesi ve saklanması gibi konuları içerir. Ancak bilgisayarlar elektrikle çalıştığı için bilgisayar için iki tür bilgi olabilir. Elektrik var veya yok. İşte bilgisayar biliminin gelişmesine paralel olarak bu mantık üzerine sayı sistemleri ve cebri geliştirilmiştir. Boole Cebri olarak adlandırılan cebir ikilik sayı sistemini temel alır.

Boole Cebri

- Bool Cebrini oluşturan üç unsur vardır:
- 1. Rakamları(0(False),1(True))(Bilgi saklama)
- 2. Aritmetik Operatörleri(VE(.), VEYA(+), NOT('))(Bilgi işleme)
- 3. Teoremleri(.....)

Boole Cebri

- Bool Cebir Aritmetik Operatörleri

A	B	S=A+B	S=A.B	S=A'
0	0	0	0	1
0	1	1	0	1
1	0	1	0	0
1	1	1	1	0

Boole Cebri

- Teoremleri

1. Değişme Özelliği

- $A+B=B+A$

- $A \cdot B = B \cdot A$

2. Birleşme Özelliği

- $A+B+C=(A+B)+C=A+(B+C)$

- $A \cdot B \cdot C = (A \cdot B) \cdot C = A \cdot (B \cdot C)$

Boole Cebri

- 3. Dağılma Özelliği
- $A.(B+C)=A.B+A.C$
- $A+B.C=(A+B).(A+C)$
- Örnek: $1.(1+0)=1.1+1.0$
- $1.(1) =1+0$
- $1 =1$
- Örnek: $1+1.0=(1+1).(1+0)=1.1=1$

Boole Cebri

- 4. Özdeşlik Kuralı

$$A+A=A$$

$$A \cdot A = A$$

Örnek: $1+1=1$, $1 \cdot 1=1$, $0 \cdot 0=0$, $0+0=0$

- 5. VE Kuralı

$$A \cdot 1 = A (\text{Etkisiz eleman})$$

$$A \cdot 0 = 0 (\text{Yutan eleman})$$

Boole Cebri

- 6. Veya Kuralı

$$A+1=1(\text{Yutan eleman})$$

$$A+0=A(\text{Etkisiz eleman})$$

- 7.Tümleyen İfadeleri

$$A+A'=1$$

$$A.A'=0$$

Örnek: $1+1'=1+0=1$

- $1.1'=1.0=0$

Boole Cebri

- 8. Tümleyenin Tümleyeni

$$(A')' = A$$

$$((A+B)')' = A+B$$

- 9. De Morgan Kuralı

$$(A+B)' = A' \cdot B'$$

$$(A \cdot B)' = A' + B'$$

Boole Cebri

- 10. Yutma Kuralı:

$$A+A \cdot B = A$$

$$A \cdot (A+B) = A$$

İsbat: $A+AB=A \cdot 1+AB=A(1+B)=A$

$$AA+AB=A+AB=A \cdot 1+AB=A(1+B)=A$$

- Örnek:

$$1+1 \cdot 0=1+0=1 \text{ veya } 0+0 \cdot 1=0+0=0$$

$$1 \cdot (1+0)=1 \cdot 1=1 \text{ veya } 0 \cdot (0+1)=0 \cdot 1=0$$

Boole Cebri

Örnek: $a(a+b')$ ifadesinin sadeleşmiş halini bulunuz.

$$a.a+a.b'=a+a.b'=a.(1+b)=a.1=a$$

Örnek: $a+(a'+b)'$ ifadesini sadeleştiriniz.

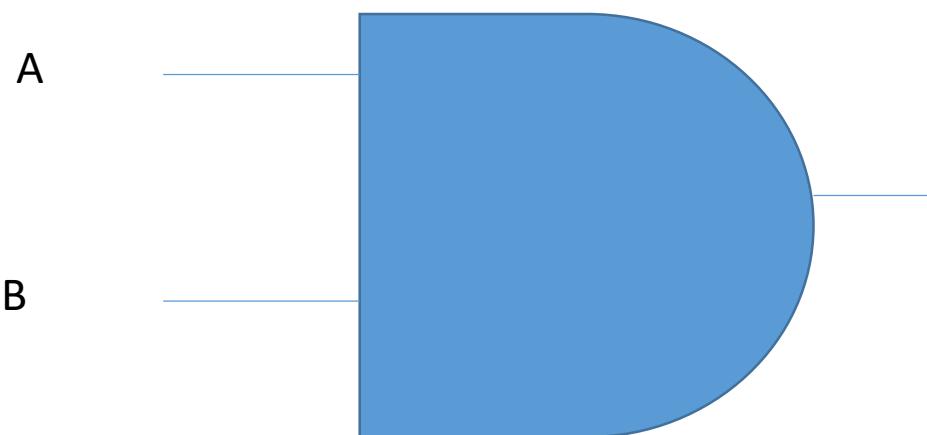
$$a+(a')'.b'=a+ab'=b'$$

Ödev: $x.y+x'.z+y.z+y'z$ ifadesini sadeleştiriniz.

LOJIK KAPILAR(LOGIC GATES)

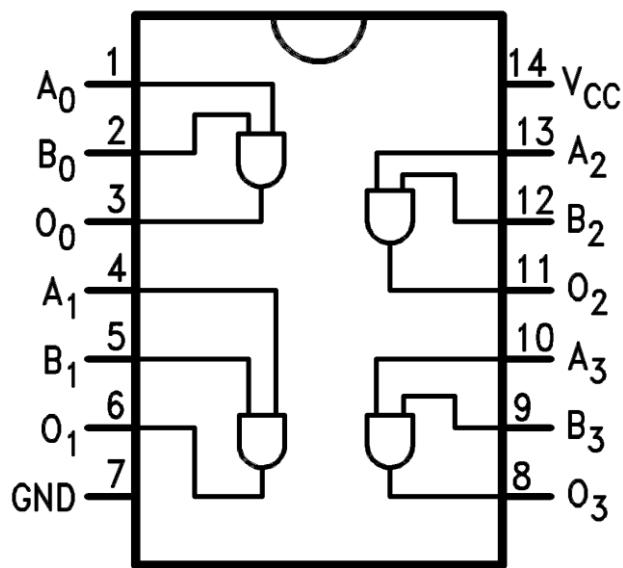
- Lojik Kapılar :Sayısal entegrelerin temelini oluşturan kapılardır:
- VE(AND),
- VEYA(OR),
- DEĞİL(NOT),
- AYRICALIKLI VEYA(YADA)(EXCLUSIVE OR)(XOR)
- VEDEĞİL(NAND)
- VEYADEĞİL(NOR),
- YADADEĞİL(XNOR)
- TAMPON(BUFFER)

VE(AND) KAPISI



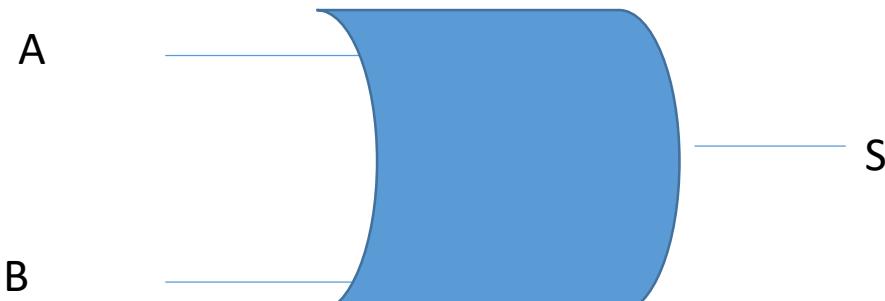
Doğruluk Tablosu
 $S = A \cdot B$

A	B	S
0	0	0
0	1	0
1	0	0
1	1	1



74AC08, 74ACT08
Quad 2-Input AND Gate

VEYA(OR) KAPISI

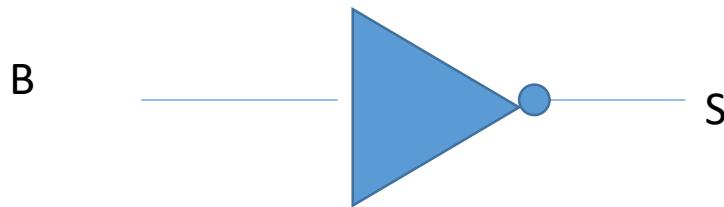


Doğruluk Tablosu
 $S=A+B$

A	B	S
0	0	0
0	1	1
1	0	1
1	1	1

74AC32, 74ACT32
Quad 2-Input OR Gate

DEĞİL(NOT) KAPISI

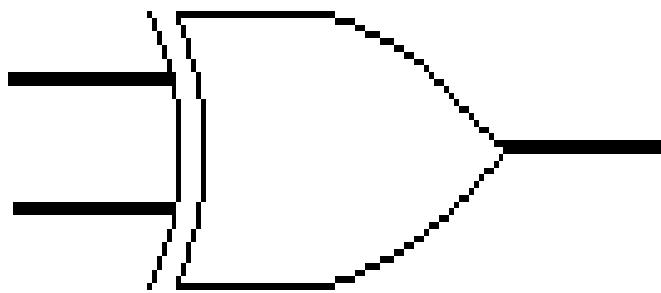


Doğruluk Tablosu
 $S=B'$

B	S
0	1
1	0

7404
Quad 2-Input OR Gate

YADA(Exclusive OR=XOR) KAPISI



Doğruluk Tablosu
 $S = A \oplus B = A'B + AB'$

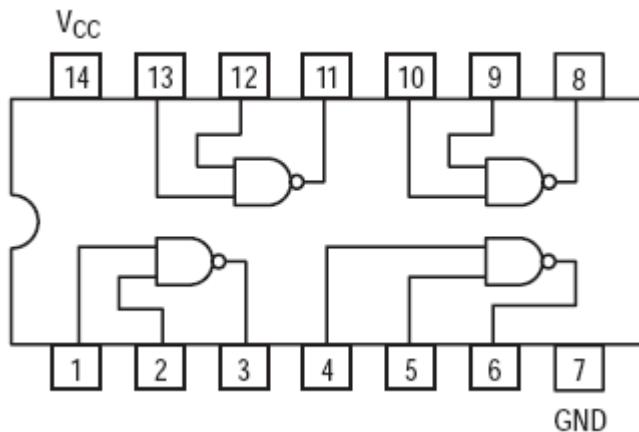
A	B	S
0	0	0
0	1	1
1	0	1
1	1	0

74HC/HCT86 Quad 2-input EXCLUSIVE-OR gate

VEDEĞİL(NOT-AND≡NAND) KAPISI

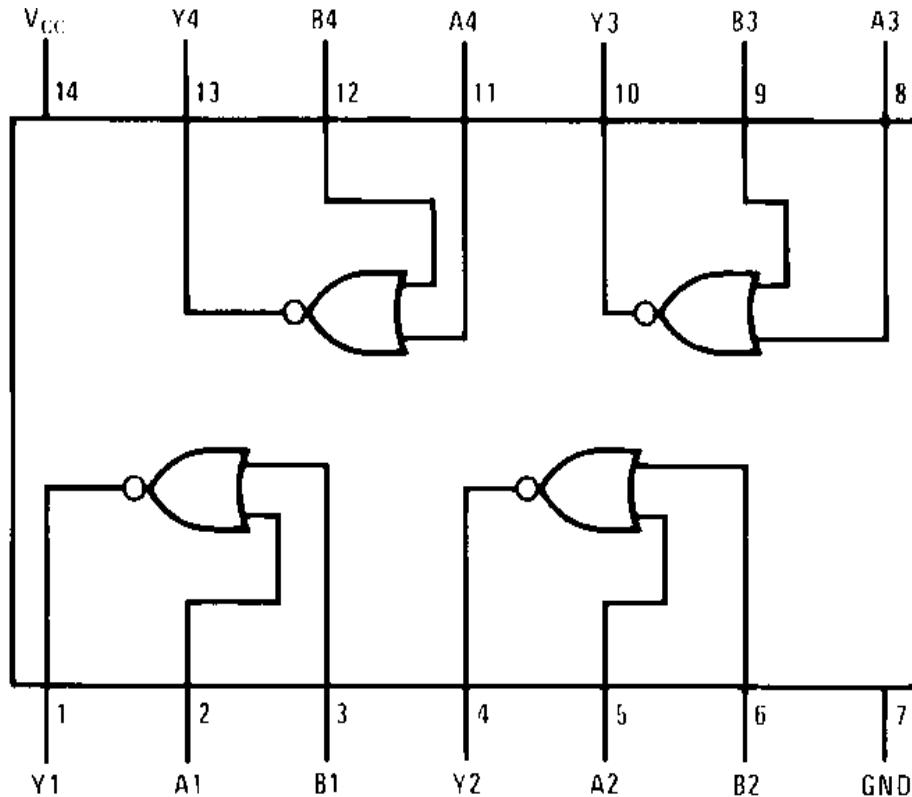
Doğruluk Tablosu
 $S = (A \cdot B)' = A' + B'$

A	B	S
0	0	1
0	1	1
1	0	1
1	1	0



7400 - Quad 2-Input NAND Gate - ON Semiconductor

VEYADEĞİL(NOT-OR=NOR) KAPISI

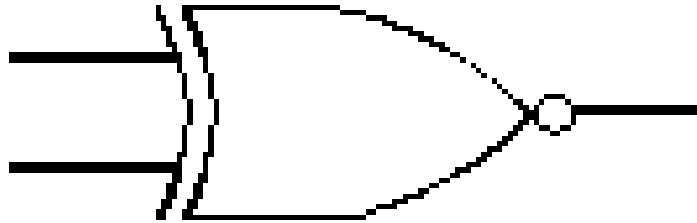


Doğruluk Tablosu
 $S = (A+B)' = A'.B'$

A	B	S
0	0	1
0	1	0
1	0	0
1	1	0

74LS02 - Quad 2-Input NOR Gate -
Fairchild Semiconductor

YADA DEĞİL(Exclusive-NOR=XNOR) KAPISI



Doğruluk Tablosu

$$S = A \otimes B = A' \cdot B' + AB$$

İSBAT:

Yada kapısının tümleyeni alınırsa;

$$S = (A'B + B'A)' = (A'B)' \cdot (B'A)' = (A+B') \cdot (B+A') = AB$$

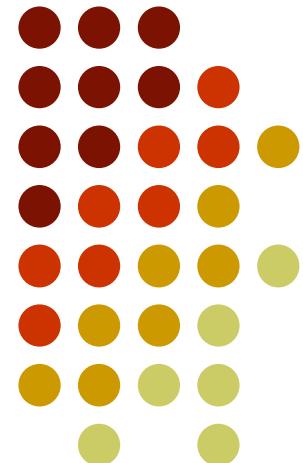
$$+ AA' + BB' + A'B' = A' \cdot B' + AB \text{ bulunur.}$$

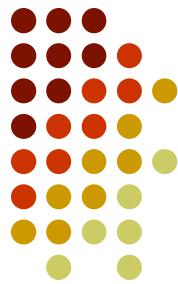
A	B	S
0	0	1
0	1	0
1	0	0
1	1	1

Boole Fonksiyonlarının İndirgemesi

- $F=xyz+x'yz+y'z$ ifadesini sadeleştirdiğimizde;
- $F=(x+x')yz+y'z=yz+y'z=z(y+y')=z$ bulunur.
- Lojik kapılarla tasarım yapılırken mümkün olduğunca az kapı kullanılması ekonomik bir çözümdür. Bu sebeple fonksiyonların indirgenmesi çok önemlidir. Yukarıdaki fonksiyon indirgenmemiş olsa idi iki tümleyen(not) kapısı 3 and kapısı ve 1 or kapısı gereki olacak idi. Oysa fonksiyon aslında tek bir girişin kendisi imiş.

Temel Kavramlar

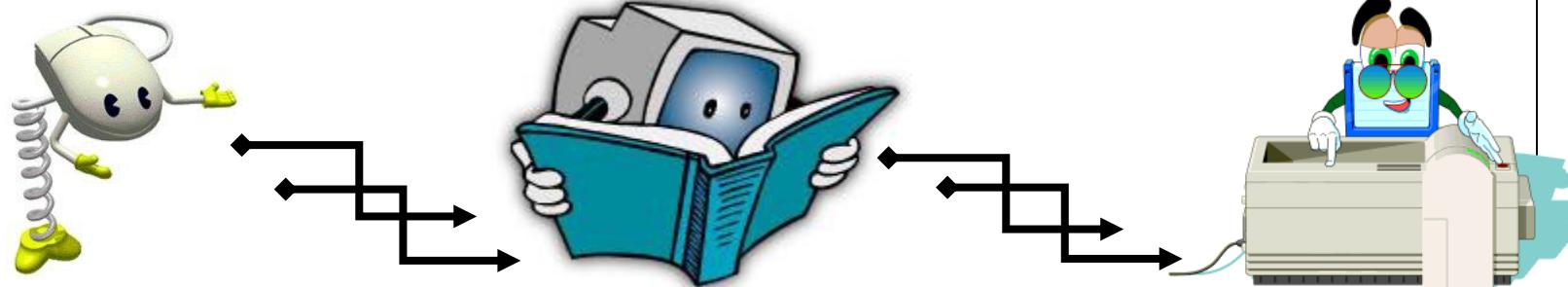
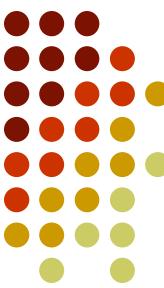




Bilgisayar Nedir?

- Bilgisayar ingilizce “hesaplayıcı (computer)” adını almıştır.
- Girdiğiniz verileri makine diline çevirir.
- Sadece 0 ve 1'lerle çalışır ama bu çalışma çok hızlı ve sürekli dir.
- Evlerde ve iş yerlerinde kullanılan küçük tip bilgisayarlara kişisel bilgisayar (PC: Personal Computer) denir.
- İlk PC IBM tarafından 1981 yılında piyasaya sürülmüştür.

Girdi-İşlem-Çıktı Prensibi



Bilgi işleyen her sistemi en azından üç seviyeye ayırmak mümkündür:

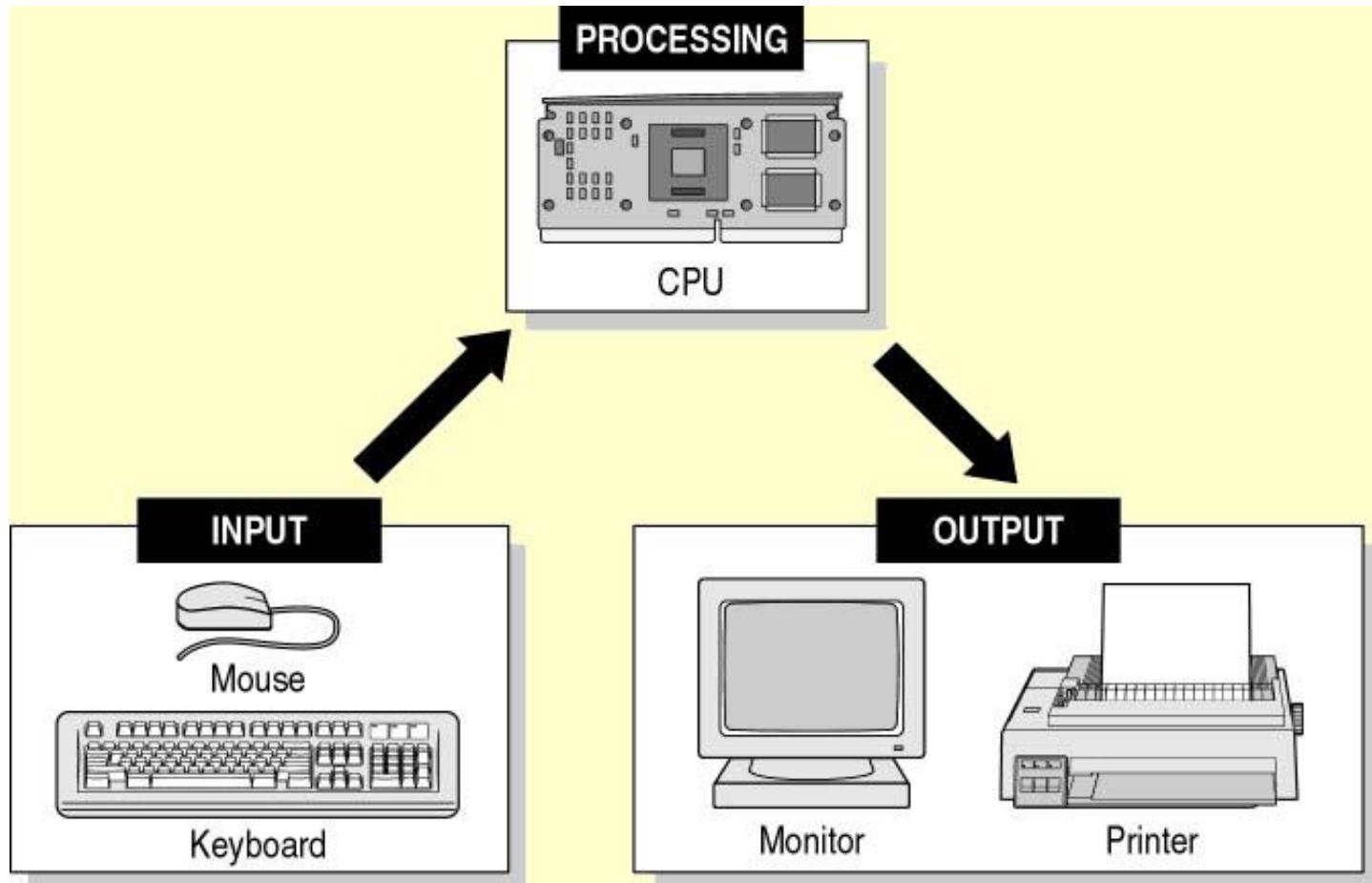
1. **Girdi veya veri kabul etme,**
2. **veri işleme ve**
3. **sonuncu olarak çıktı olarak verilmesi.**

Eğer sistem bunlara ek olarak veriyi tutabiliyorsa veri saklama denen bir dördüncü bileşen daha bunlara eklenir.

PC'nin içinde takılı veya dışarıdan ona bağlı cihazlar "Girdi ve Çıktı (I/O-Input/Output)", "İşlem(Processing)" veya "Saklama (Kaydetme) (Storage)" işlev gruplarından birine dahildir.



İşlemlerin 3 aşaması





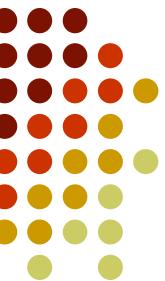
Bilgisayarı Oluşturan Bileşenler

Donanım

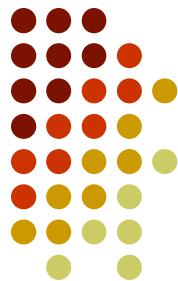
Yazılım

Donanım denilince anlaşılması gereken bilgisayarın fiziki görünümüdür.

Yazılım ise bilgisayarın kullandığı programların genel adıdır.



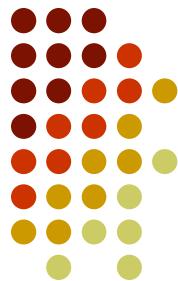
DONANIM



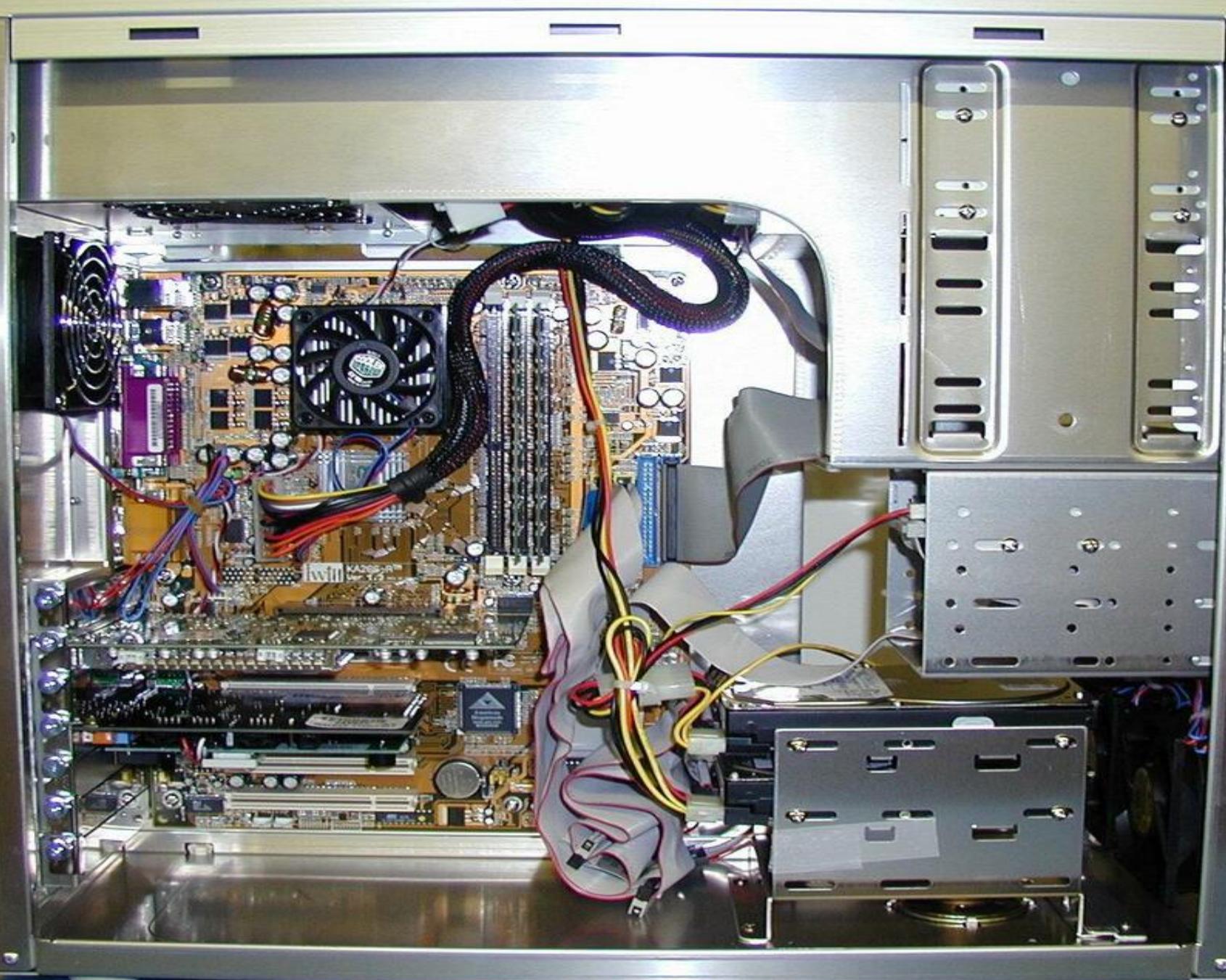
DONANIM (HARDWARE)

- Bilgisayarların elle tutulabilen, gözle görülebilen fiziksel kısımlarına donanım denir.
- Dahili Donanım: Bilgisayar kasasının içinde bulunan donanımlardır.
 - anakart, mikroişlemci, bellek, sabit disk, ...
- Harici Donanım: Kasanın dışında yer alan donanımlardır. 2 farklı türde harici donanım vardır:
 - Girdi aygıtları: klavye, fare, mikrofon, tarayıcı, ...
 - Çıktı aygıtları: ekran, yazıcı, hoparlör, çizici, ...

KASA (AT ve ATX)

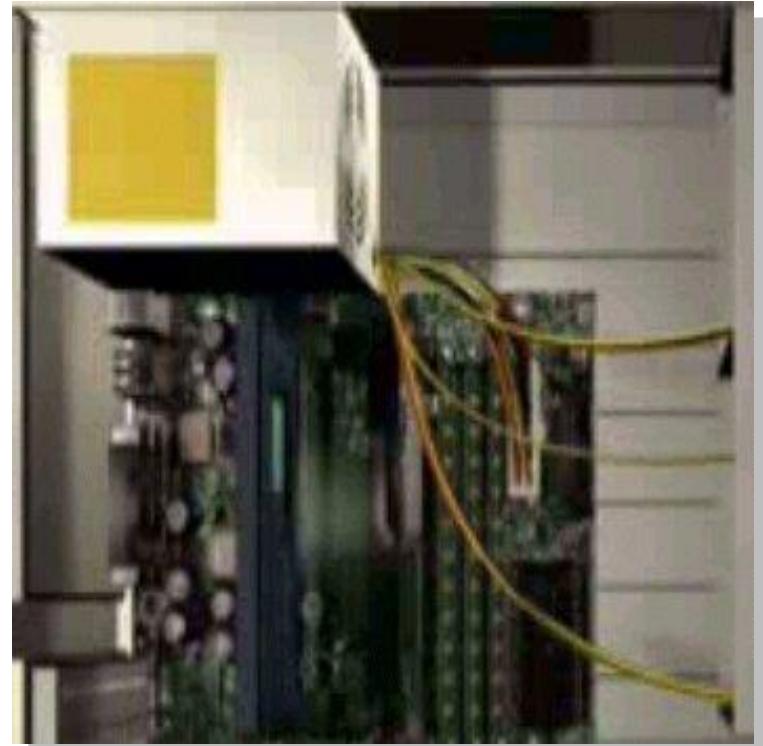


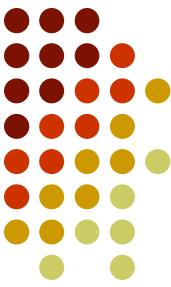
Bilgisayarı oluşturan kartları, merkezi işlemci (CPU), RAM, diğer genişletme kartlarını içinde barındıran ve fiziksel zarara karşı koruyan elektrostatik kutulardır.



GÜÇ KAYNAĞI

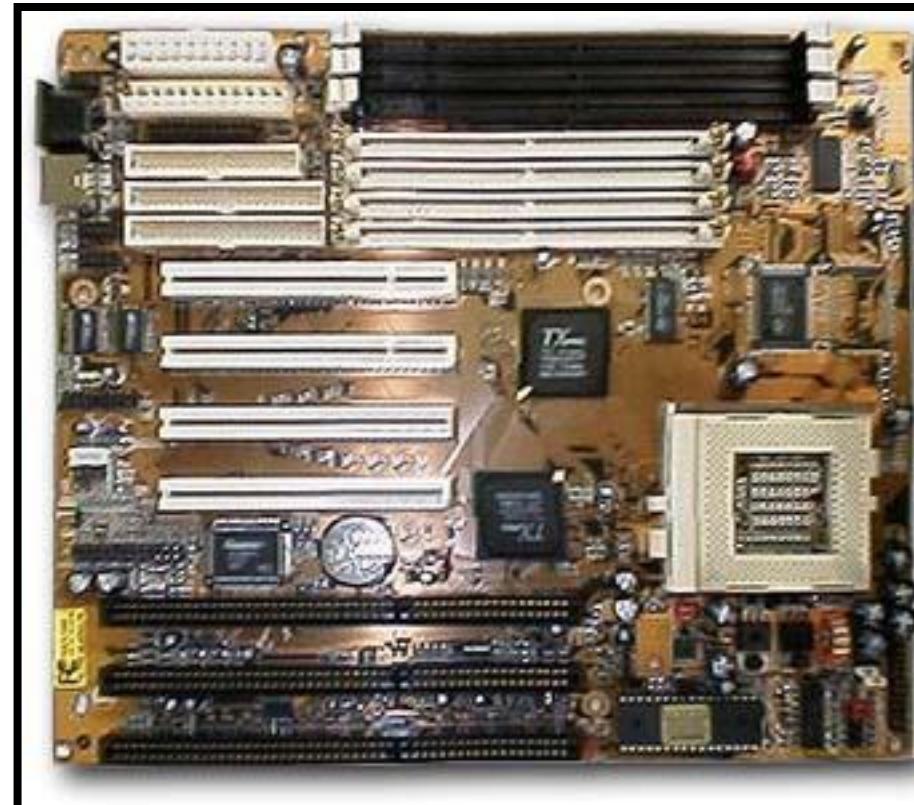
- ❖ Hat gerilimi 220 volt dalgıç akımdır.
- ❖ PC'de çoğu sayısal aygit 5 ve 12 volt kullanmak üzere ayarlanmıştır.
- ❖ Bu dönüşüm süreci bir güç kaynağı tarafından gerçekleştirilir.
- ❖ Güç kaynağı, kasanızın iç tarafında, sağ arkada duran, gümüş veya siyah renkli kutudur.
- ❖ Güç kaynağınız bir PC'ye kaç tane çevre birimi yerleştirebileceğinizi kısmen belirler.



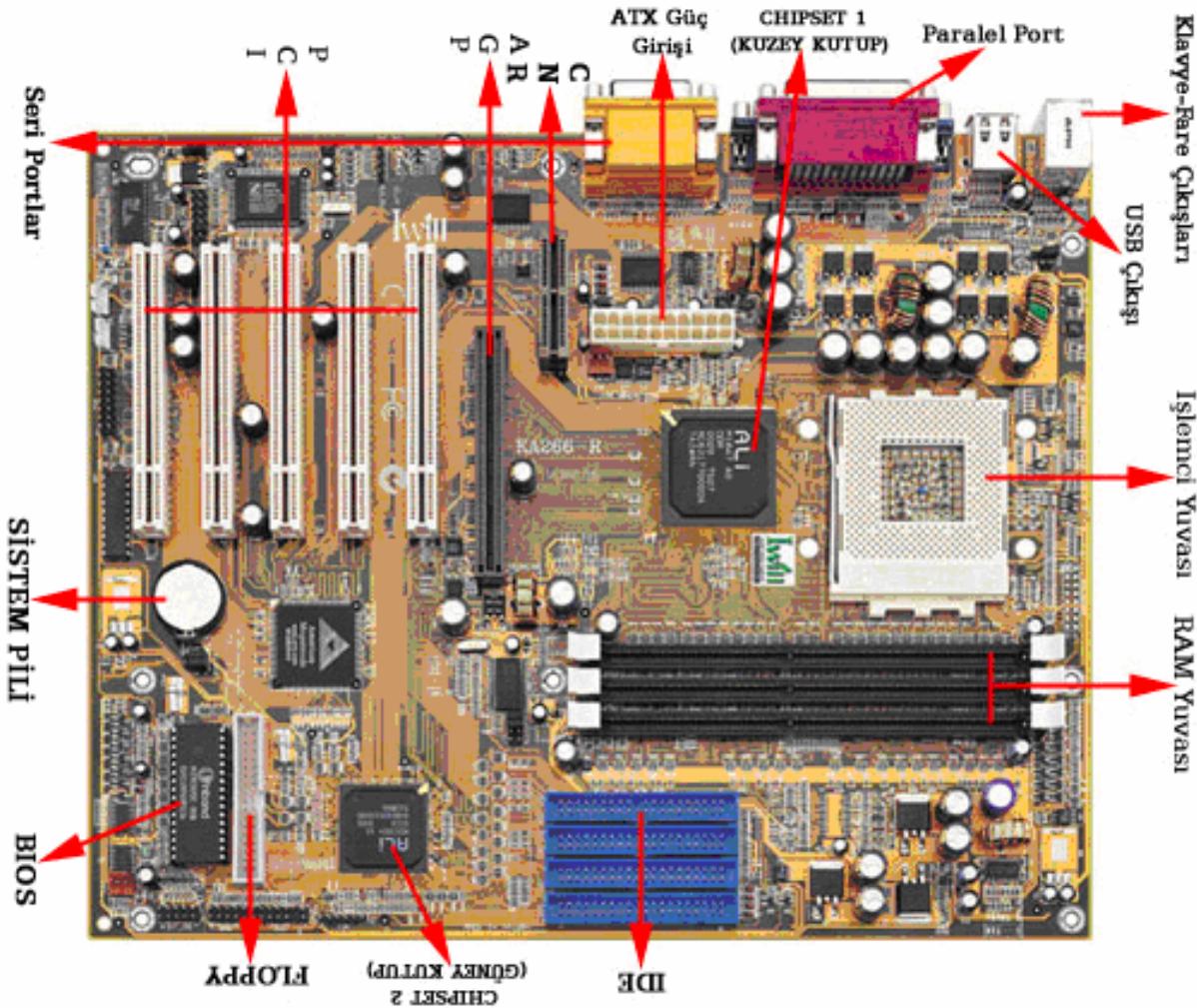
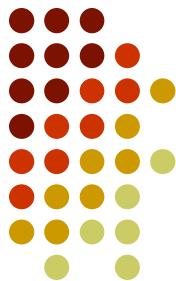


ANA KART

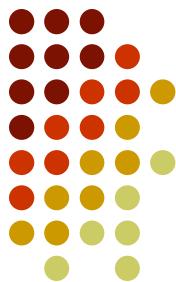
- ❖ Ana kart, fiberglast'tan yapılmış, üzerinde bakır yolların bulunduğu, genellikle koyu yeşil bir levhadır.
- ❖ Ana kart üzerinde;
 - ❖ mikro işlemci,
 - ❖ bellek,
 - ❖ genişleme yuvaları,
 - ❖ BIOS ve
 - ❖ diğer yardımcı devreler yer alır.
- ❖ Anakart'a bütün kartların anası denir;
 - ❖ Çünkü PC'nin diğer bileşenleri bir şekilde anakarta bağlanmaktadır ve birbirleri ile anlaşmak için anakartı bir platform olarak kullanırlar
 - ❖ Yani PC'nin "sinir sistemi" anakart üzerinde yer almaktadır.



Anakart (Mainboard)

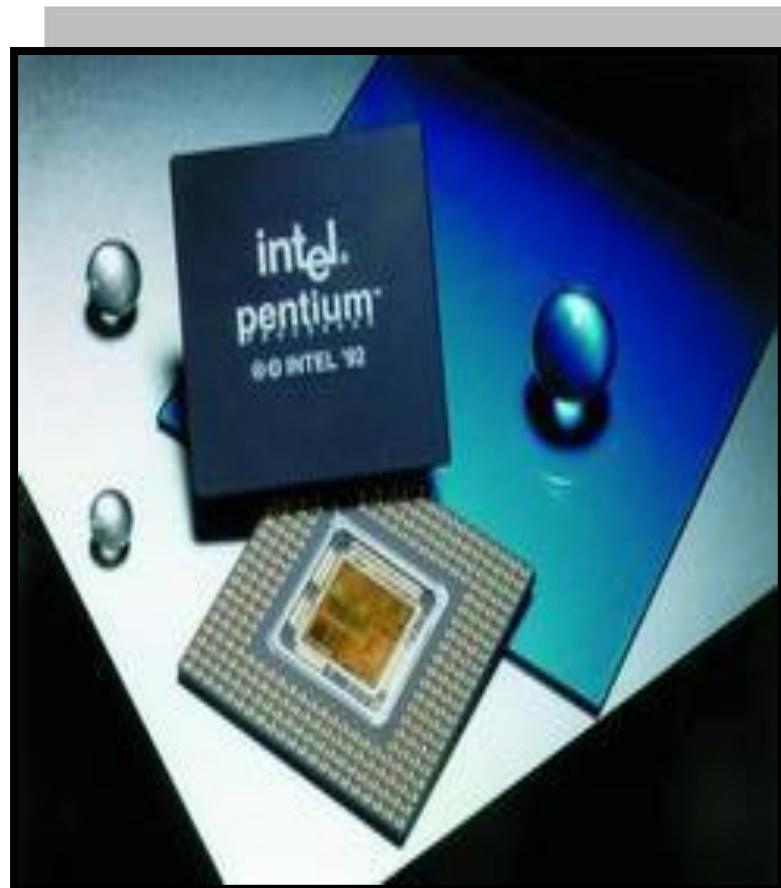


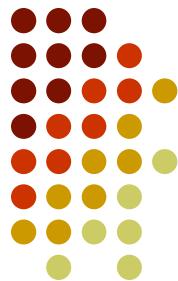
CD Sürücü, Disket Sürücü ve Disk Sürücü kablolarla ana karta bağlanır.
Kablolar türleri: IDE kablo, SCSI kablo, Disket sürücü kablosu



CPU (Merkezi İşlemci Birimi)

- ❖ Bilgisayarın program komutlarını bellekten aldıktan sonra kodlarını çözen ve karşılığı olan işlemleri yerine getiren merkez birimdir.
- ❖ Bilgisayarın en önemli öğesidir.
- ❖ Bu nedenle beyine benzetilir.
- ❖ Bilgisayarın yapabileceği tüm komutları verir.
- ❖ Tüm herşeyi ve her birimi yönetir.



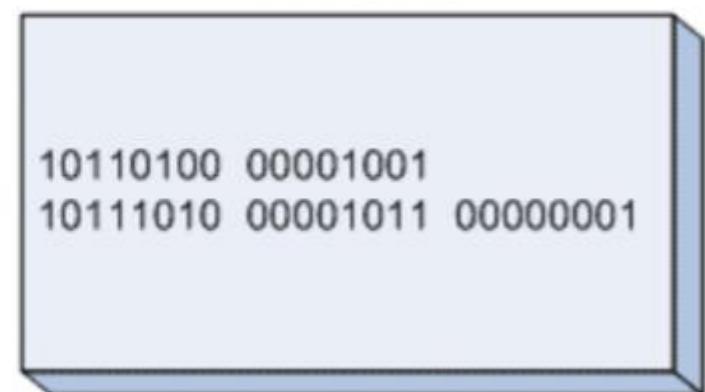
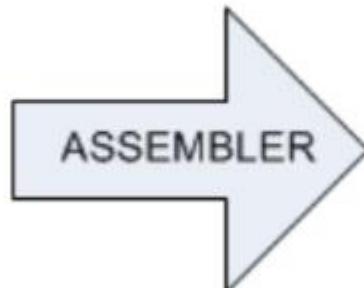
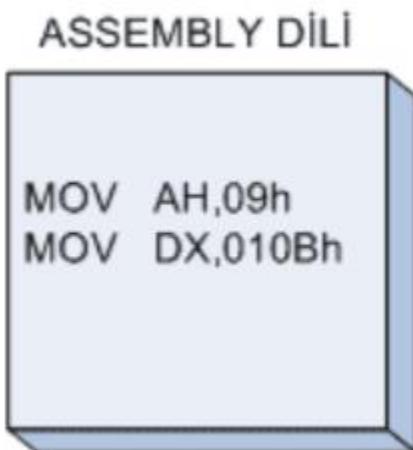


Makine Dili

Mikroişlemci sadece ikili sayı sisteminde yazılan komut kodlarını anlar.

Assembly dilinde yazılan programları makine diline çevirmek için Assembler adı verilen çeviriçi(derleyici) programlar kullanılır.

Aşağıda verilen şekilde Assembly dili, Makine dili ve Assembler blok olarak görülmektedir.





BELLEK

- ❖ Bellek bilginin saklandığı içsel depo olarak tanımlanabilir.
İkiye ayrılır;

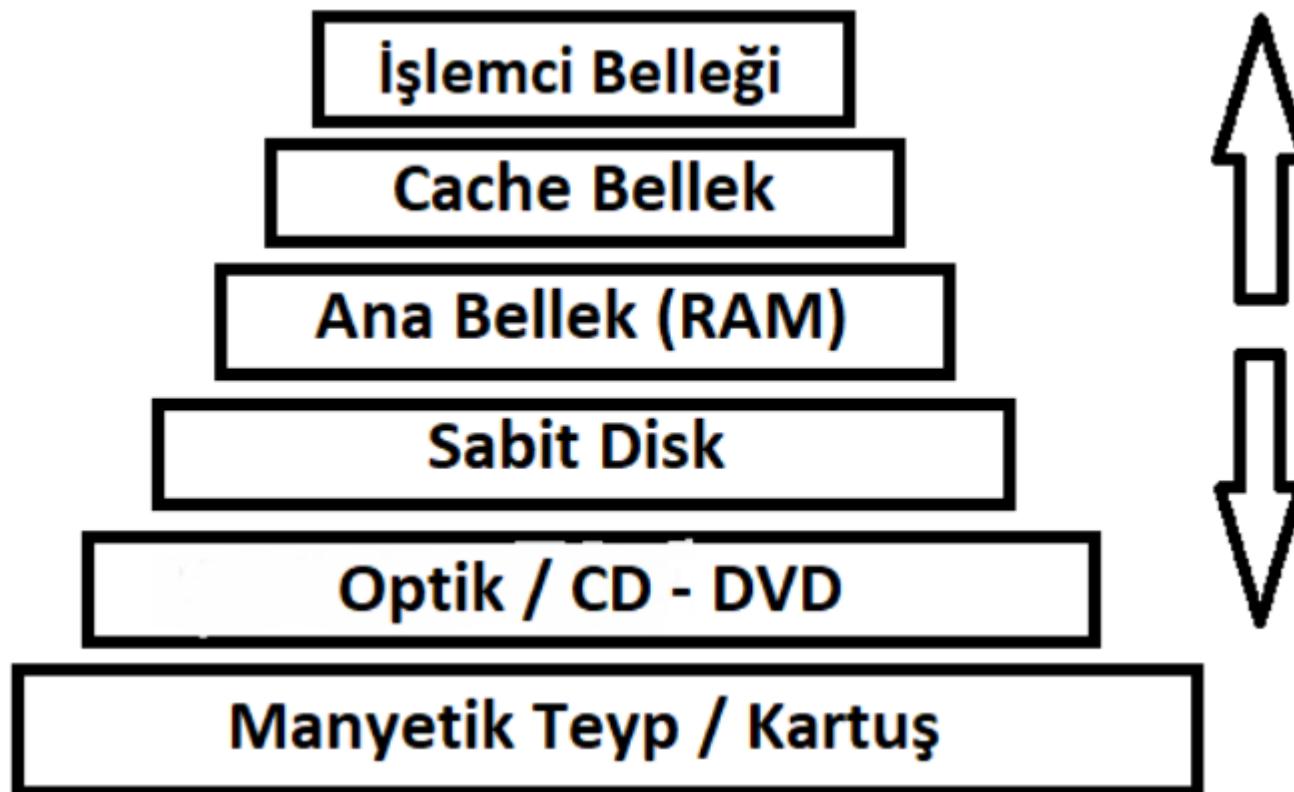
1. Ana Bellek;

- a. **ROM (Read Only Memory)**: Silinmez bellektir. Bu bellekten sadece okuma yapılır. Temel sistem bilgilerini içerir. ROM'ların veri okuma kapasitesi RAM'e göre düşüktür.
- b. **RAM (Random Access Memory)**: Geçici ve silinebilir bellektir.

2. Yardımcı Bellek; RAM içinde yazılan programların, depolama verilerinin saklandığı ve tekrar geri çağrıma olanağının bulunduğu bellek türüdür. Bunlar manyetik ortamlardır (Harddisk, Compact Disk, Disket, Teyp vb.)



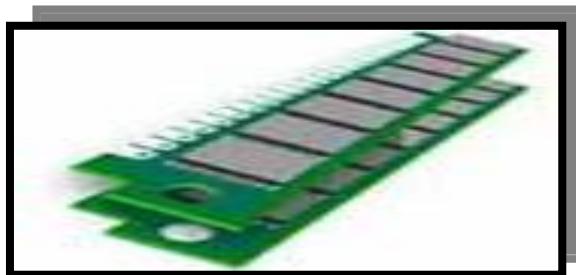
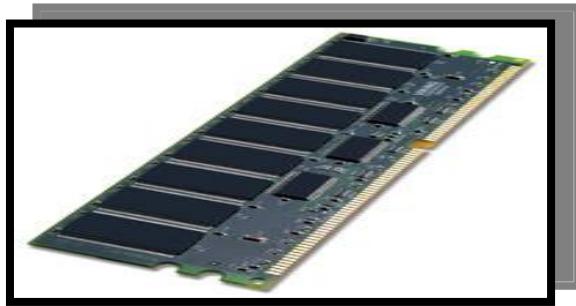
BELLEK HİYERARŞİSİ





RAM (Random Access Memory)

- ❖ Bilgisayarınızda o anda çalışan bir programların, gerekli bilgilerin saklanarak daha sonra gerektiğinde kullanım için alındığı alana RAM denir.
- ❖ Diğer bir deyişle bir geçici bellek görevindedir.
- ❖ Bilgiler gerektiğinde kullanılır. Gerekmediği zaman silinir.
- ❖ RAM üzerindeki bilgiler kısa ömürlüdür. Bilgisayarınızı kapattığınızda bilgiler silinir.
- ❖ Bilgilerimizi uzun ömürlü olarak saklamak istiyorsak, manyetik alana kayıt yapan sabit diskleri kullanmamız gerekmektedir.



RAM Türleri

- ❖ Sadece anakarta uygun RAM türü kullanılabilir.
- ❖ Eskiden kullanılan Extended Data Out (EDO-RAM), Synchronous Dynamic (SD-RAM) ve Rambus Dynamic (RD-RAM) gibi RAM türleri zamanla yerini Double Data Rate SDRAM (DDR-RAM) türüne bırakmıştır.
- ❖ Günümüzde 533-2000 MHz hızları arasında değişen DDR2 ve DDR3 tipi RAM'ler yaygın olarak kullanılmaktadır.



SD-RAM



DDR-RAM

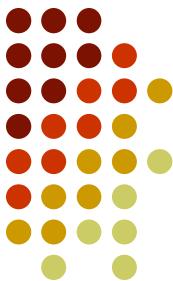


EDO-RAM



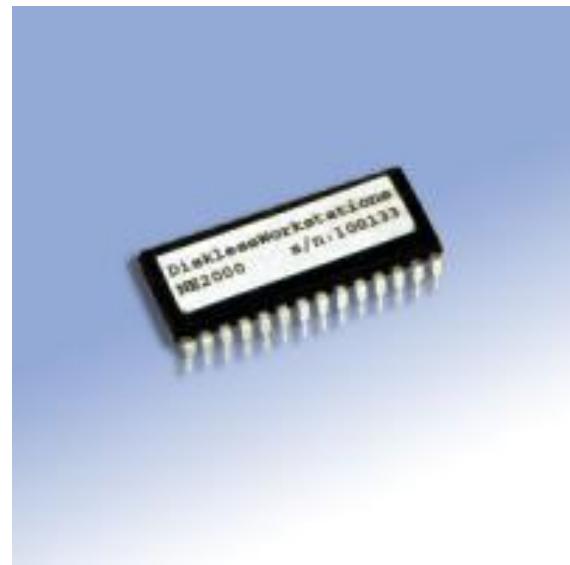
RD-RAM





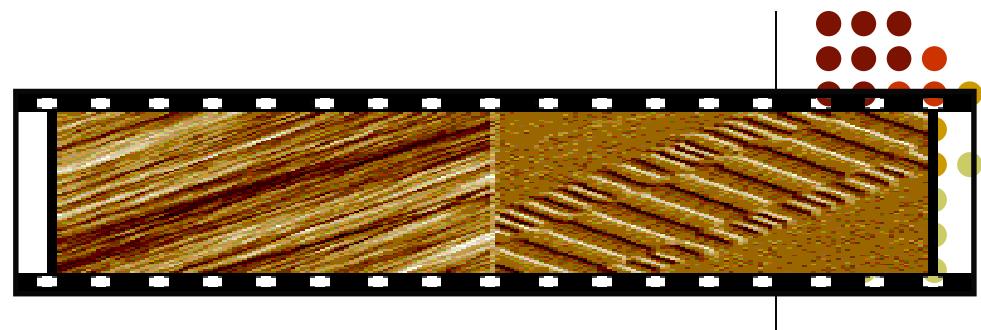
ROM (Read Only Memory)

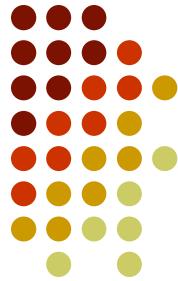
- ❖ Üretimleri esnasında özel bilgiler ile programlanmış hafıza tipleridir.
- ❖ ROM chipleri sadece bilgisayarlarda değil birçok elektronik cihazda da kullanılmaktadır.
- ❖ Temel olarak beş adet ROM tipi bulunmaktadır
 - ❖ ROM (Read Only Memory)
 - ❖ PROM (Programmable Read Only Memory)
 - ❖ EPROM (Electrically Programmable Read Only Memory)
 - ❖ EEPROM (Electrically Erasable Programmable Read Only Memory)
 - ❖ Flash memory
- ❖ Farklı ROM tipleri olmakla birlikte hepsi için temel iki özellik bulunmaktadır.
 - ❖ Bu tip hafızalarda elektrik olmasa dahi bilgiler kaybolmazlar.
 - ❖ Bu tip hafızalarda tutulan bilgiler ya değiştirilemezdir, yada değiştirilmeleri için özel işlemler gerekmektedir.



SABİT DİSK (HDD)

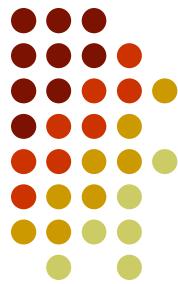
- Sabit disk bilgilerin saklanıldığı bir depodur istenilen bilgiler ve dosyalar sabit diskten okunur, belleğe aktarılır ve işlenebilmesi sağlanır.
- Sabit diskleri birbirinden ayıran en önemli özellik veri depolama kapasiteleridir.(Örn: 40 mb, 40 gb, 80 gb vb.)





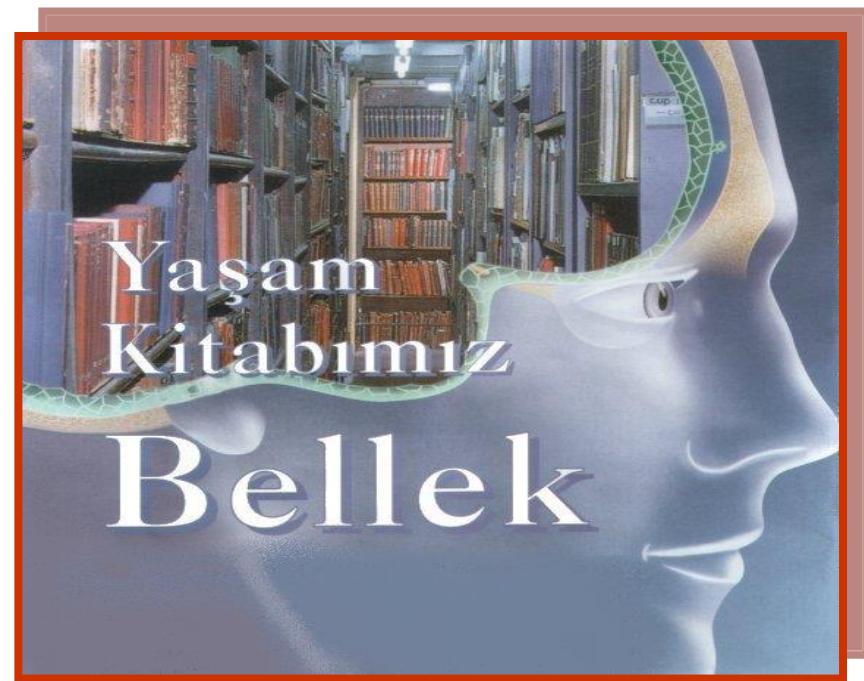
SABİT DİSK (HDD-Hard Disk Drive)

- Harddisk manyetik bir kayıt birimidir.
- Sabit disk, vakumlu (havası alınmış ve sürtünmenin en aza indirildiği) bir metal kutu içerisinde yerleştirilmiş bir veya daha çok sayıda üzeri manyetik olarak yazılabilen bir film kaplı alüminyum disklerden oluşur.
- RPM olarak bilinen sabit disk hızı, RPM ne kadar yüksek olursa o kadar hızlı çalışması anlamına gelmektedir. (Hızlı çalışması daha hızlı bir şekilde veri aktarması demektir)
- Milyonlarca küçük mıknatıştan oluşan sabit diskte yanyana duran bu mıknatısların birbirlerine göre durumları birer "0" ve "1" rakamlarını oluşturmaktadır.
- 1 Gb 'lık diskte 8,589,934,592 (8 milyar 589 milyon 934 bin 592) adet küçük mıknatıs bulunur ve bunların saniyede milyonlarcasının yön değiştirmesiyle veriler okunabilir veya değiştirilebilir.



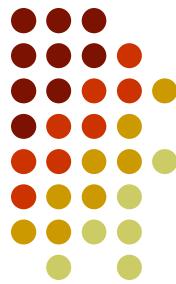
Bir HDD Nelerden Oluşur?

- Sabit diskler bir merkez etrafında dönen birkaç diskten ibarettir. Bu diskler alüminyum yapısında olup üzerleri manyetik alan oluşabilmesi için düzgün demir oksit ile kaplanmıştır.
- HDD'de disklerdeki bilgileri okuma/yazma için kullanılan
 - sürücü motoru
 - okuma/yazma kafaları
 - adım motoru
 - kontrol devresi bulunur.



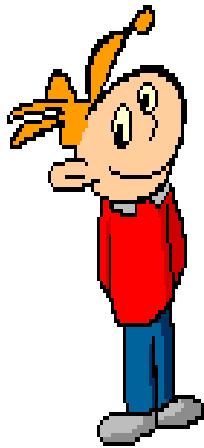
**Belleğin
birimi
NEDİR?**





Bellek Birimleri

- En küçük bellek birimine **Bit** denir.
- 1 Bit, 0 ya da 1'den (kapalı devre=0, açık devre=1) oluşur.
- Bir karakteri ifade edebilmek için 8 bit'in biraraya gelmesi ile oluşan **Byte** kullanılır.
 - 1024 Byte = 1 KiloByte (KB)
 - 1024 KB = 1 MegaByte (MB)
 - 1024 MB = 1 GigaByte (GB)
 - 1024 GB = 1 TeraByte (TB)



BİRİMLER



Byte

**Kilobyte
(KB)**

**Megabyte
(MB)**

**Gigabyte
(GB)**

**Terabyte
(TB)**

Bir karakter
rakam yada
sembol

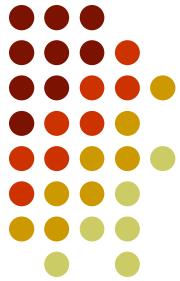
Bin karakter,
yada satır aralıklı
yazılmış
bir sayfalık metin

Bir milyon karakter
veya bir roman

Bir milyar karakter,
ya da
bin tane roman

Bir trilyon karakter,
ya da
bir kütüphane

CD Sürücü (CD-ROM)



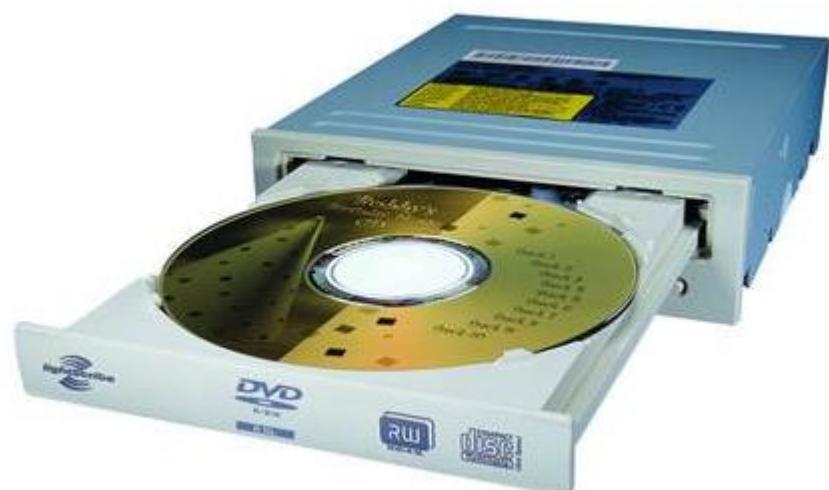
- CD-ROM olarak adlandırılan CD sürücüler, fiyatlarının ucuzlamasıyla birlikte 90'lı yılların başlarında PC'lerde yaygın olarak kullanılmaya başlandı. CD'lerin kapasitesi başlarda 650MB iken daha sonra 700MB'a çıktı.
- İlk olarak 1998'de geliştirilen CD yazıcıları bilgi okumanın yanında boş CD'lere bilgi yazabilme yeteneğine de sahiptirler. 2 tür boş CD vardır:
 - CD-R: sadece bir defa bilgi yazılabılır, içindeki bilgiler silinemez
 - CD-RW: birçok defa bilgi yazılabılır, içindeki bilgiler silinebilir
- CD sürücülerinin ve yazıcılarının üzerinde yazan numaralar bilgi okuma ve yazma hızlarını gösterir. 1X, saniyede 150KB bilgi anlamına gelir. ($52X = 7.800\text{KB/s}$)
 - 52x32x52 türünde bir yazıcı, tüm diskleri 52 hızlı okur, 32 hızla CD-RW diske yazar, 52 hızla CD-R diske yazar.





DVD Sürücü ve Yazıcı

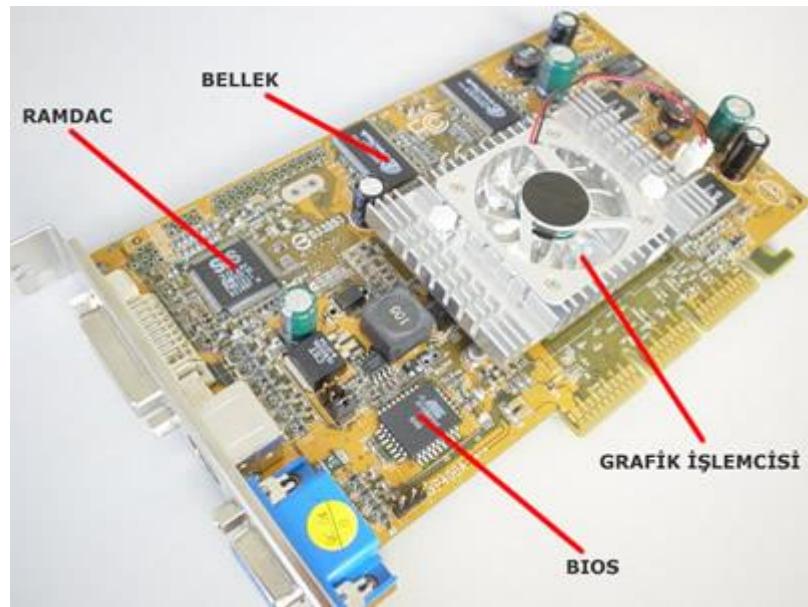
- 90'lı yılların sonlarına doğru ortaya çıkan DVD diskler, CD'lere göre daha yüksek kapasiteye sahiptirler. Tek katmanlı DVD diskler 4.7GB, çift katmanlı (double layer) olanlar ise 8.5GB bilgi saklayabilirler.
- DVD sürücü ve yazıcılar fiyatlarının ucuzlamasıyla 2000'li yıllarda yaygınlaştılar.
- DVD'de 1X saniyede 1385KB anlamına gelmektedir. 16X DVD sürücü ve yazıcılar 22MB/s hızları ile 52X CD'lerden 3 kat hızlıdır.





EKRAN KARTI

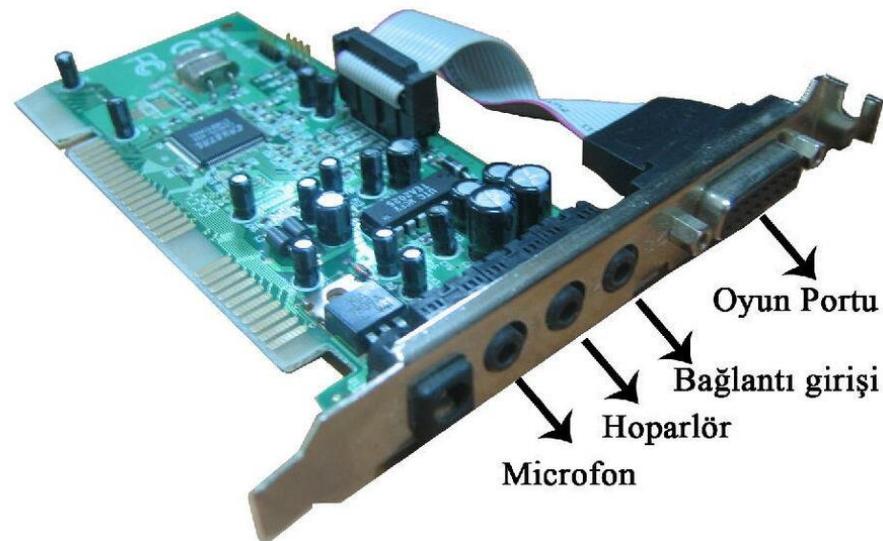
- Bilgisayarın görüntü çıkış birimi ekrandır.
- Yani görüntü ekran üzerinde oluşur, ancak ekranın veri girişi, yada bilgisayarın görüntü oluşturmak için kullandığı ara birim, ekran kartı yada görüntü kartı adını alan birimdir.
- Bilgisayarın oluşturduğu sayısal bilgiler ekrana video sinyaline çevrilerek gönderilir. Bu işlemi ekran kartı yapar.
- Grafik kartı da denen bu birim ayrı bir kart olabileceği gibi bazen ana kart üzerine tümleşik olarak üretilmiş olabilir





SES KARTI

- Ses kartı ses kaydedilmesine, bilgisayardan başka cihazlara ses gönderilmesine yarayan donanımdır.
- Ses kartları da ekran kartları gibi bazı anakartlara tümleşik (on board) olarak gelebilmektedir.
- Hoparlör sayesinde ses çıkışı, mikrofon sayesinde ses girişi yapabiliriz.

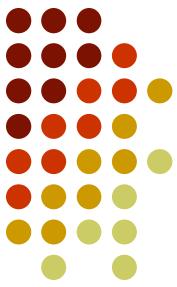




ETHERNET (AĞ) KARTI

- Bilgisayarların kablolarla birbirine bağlanmasıında ağ kartları kullanılır.
- Sayısal verileri elektrik sinyallerine dönüştürüp kablolarla diğer bilgisayarlardaki ağ kartlarına iletirler. O da bu sinyalleri sayısal veriye dönüştürüp bilgisayara iletir.
- Kablo ile 10 Mbit/s hızında veri aktaran kartlara Ethernet, 100 Mbit/s hızında olanlara Fast Ethernet, 1000 Mbit/s hızında olanlara ise Gigabit Ethernet denir.
- WiFi olarak bilinen 802.11g kablosuz bağlantı hızı 54Mbit/s'tır.





Monitör (Ekran) Çıktı Aygıtı

- Bilgisayardan gelen görüntü sinyallerini görmemizi sağlayan aygittir.
- Televizyonlar ile aynı prensipte çalışırlar. CRT ve LCD olmak üzere iki çeşidi vardır:
 - CRT (Cathode Ray Tube): İçinde bulunan görüntü tüpü nedeniyle büyük ebatta ve ağırdır.
 - LCD (Liquid Crystal Display): Sıvı kristaller ile görüntüyü oluşturur. İnce ve hafiftir.



Yazıcı Çıktı Aygıtı

- Yazılıları, grafikleri ve resimleri kağıda aktarmak için kullanılan aygittır.
- 3 çeşidi vardır:
 - Nokta vuruşlu
 - Bir iğnenin kağıda temas eden bir mürekkep şeridine vurması ile çıktı alır.
 - Gürültülü ve yavaş çalışır.
 - Fatura basmak gibi işlerde kullanılır.
 - Mürekkep püskürtmeli:
 - Faklı renklerdeki mürekkepleri kağıdın üzerine püskürterek çıktı alır
 - Renkli fotoğrafların basılmasında kullanılabilir.
 - Lazer:
 - Kağıdın üzerinde yazılacak olan verinin yerlerini lazer ile elektrostatik olarak işaretler ve o bölgelere toner adı verilen tozu yapıştırır.
 - Hızlı çıktı alması sebebiyle ofislerde kullanılır.
 - Renkli çıktı alabilen modelleri de vardır (pahalıdır).

Klavye ve Mouse Girdi Aygıtları



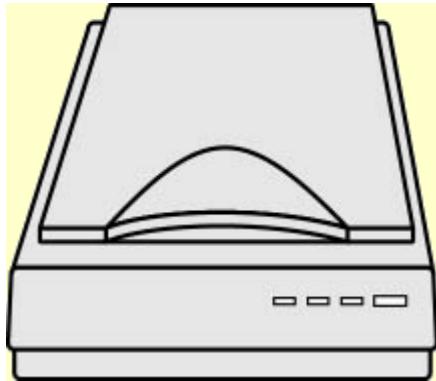
Klavye: Üzerinde karakterler, rakamlar ve gereklili olan diğer tuşların bulunduğu gelişmiş bir girdi aracıdır.



Mouse: Grafik ortamlarda kullanılan, ekrandaki nesneleri işaretlemek ve seçmek için kullanılan girdi aracıdır.

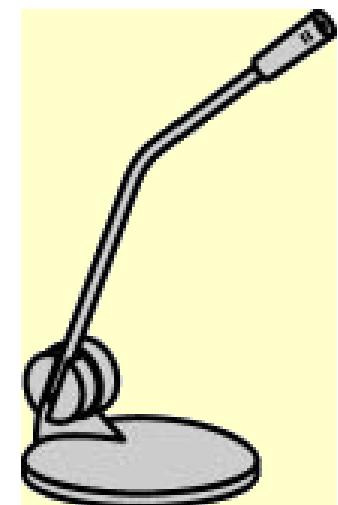


Tarayıcı ve Mikrofon Girdi Aygıtı



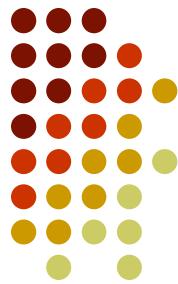
Tarayıcı: Kağıt üzerindeki yazı, grafik ve fotoğrafları sayısal bilgiye dönüştürür. Bu sayısal bilgiler daha sonra kaydedilebilir.

Mikrofon: Sesini ses kartına aktarır. Ses kartı da sesin sayısal bilgiye dönüştürülmesini sağlar. Bu şekilde ses bilgisayar ortamında kaydedilebilir.



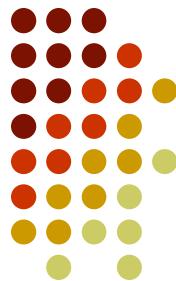


YAZILIM



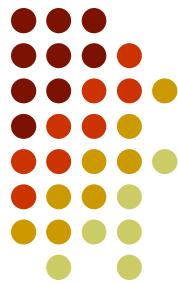
Yazılım

- Bilgisayarın kullanılmasını sağlayan, bilgisayara istenilen işlemlerin yaptırılması için işlenen deyimlere verilen genel bir addır.
- İki ana başlık altında incelenebilir:
 - Sistem Yazılımları
 - Uygulama Yazılımları.



Sistem Yazılımları

- Son kullanıcının doğrudan kullanmadığı, bilgisayarın donanımına müdahale edilmesini ve bu donanımın kullanılmasını sağlayan programlardır.
- Sistem yazılımlarından bazıları:
 - İşletim Sistemi
 - Hizmet programları
 - Aygıt Sürücüleri
 - Dil Çeviriciler



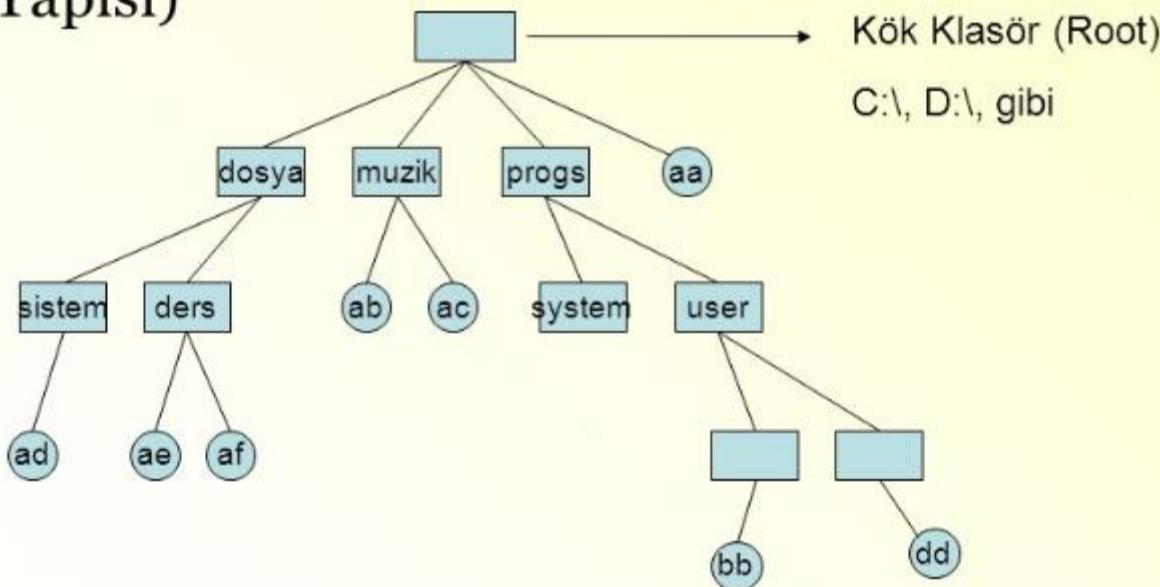
İşletim Sistemi

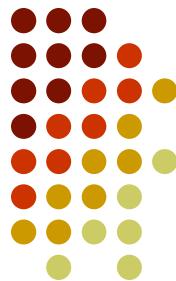
- Bilgisayar kaynaklarını koordine ve kontrol ederken, kullanıcı ile bilgisayar arasında da bir kullanıcı arayüzü sağlar ve uygulamaları çalıştırır.
- Üç ana grupta toplanabilirler:
 - Gömülü İşletim Sistemleri (Embedded Operating Systems): İşe özgü olarak geliştirilir.
 - Ağ İşletim Sistemleri (Network Operating Systems): Novell, UNIX, Linux, MS-Windows NT, Solaris, AIX
 - Bağımsız İşletim Sistemleri (Stand-alone Operating Systems). MS-Windows, MacOS, Linux

İşletim Sistemlerinde depolama yapısı



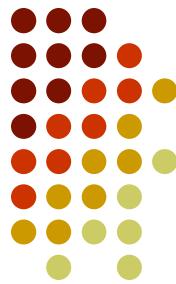
Kök Klasör (Root Directory) : Hiyerarsık düzendeki klasör sisteminde en üstteki klasöre denir. (Tree – Ağaç Yapısı)





Hizmet Programları

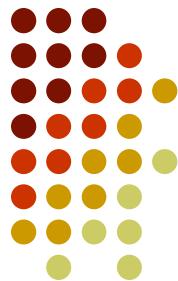
- Anti-virus programları
- Yedekleme programları
- Dosya sıkıştırma programları
- Veritabanı yönetim programları
- ...



Aygıt Sürücüler

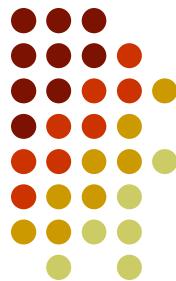
- Bilgisayar sisteminde varolan
 - Klavye,
 - Fare,
 - Ekran,
 - CD sürücü

gibi herhangi bir aygıtın bilgisayarın işletim sistemi tarafından tanınmasını sağlamak üzere aygıt üreticileri tarafından sağlanan yazılımlardır.



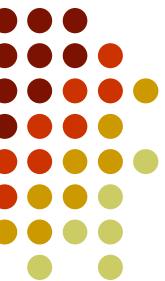
Dil Çeviriciler

- Bilgisayar yalnızca Makina Dili olarak adlandırılan ve 0 ve 1'lerden oluşan bir dil tanıyabilir.
- Herhangi bir bilgisayar programlama dilinde yazılmış olan bir program, Makina Dili'ne çevrilmek zorundadır.
- Çeviriciler bu işlemi gerçekleştirir.
- İkiye ayrırlar
 - Yorumlayıcı(Interpreter)
 - Derleyici (Compiler)



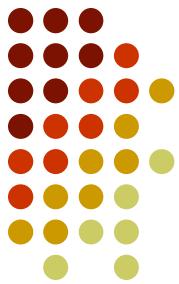
Uygulama Yazılımları

- İkiye ayrırlar.
- Genel Amaçlı:
 - Herkesin kendi ihtiyacı doğrultusunda kullanabileceği, genel kullanıma hizmet eder şekilde hazırlanmış programlardır.
 - Kelime-işlem programları, web görüntüleyiciler, tablo işleme programları, sunum hazırlama programları
- Özel amaçlı:
 - Kullanıcıların kendi istekleri doğrultusunda geliştirdikleri yazılımlardan, mimari tasarım programlarına, web uygulamalarından değişik mühendislik hesaplarını yapan programlara kadar çok geniş bir yelpazeyi kapsarlar.



Bilgisayar Ağları

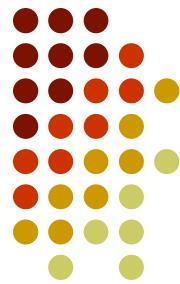
Bilgisayar Ağı kullanmanın avantajları



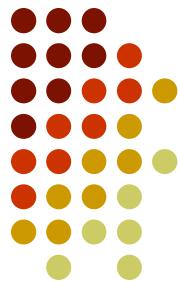
- Donanım maliyetini düşürür
- Uygulamalar paylaşılır
- Verinin paylaşılması sağlanır
- Veri yönetiminin merkezileşmesi
- Güvenlik
- Çalışma gruplarını birbirine bağlar
- Esnek erişim imkanı



Bilgisayar Ağı kullanmanın dezavantajları



- Dosya Erişimi yavaş
- Kullanıcı izlenebilir
- Saldırganlar kişisel dosyalara erişebilir
- Ağ çöktüğü zaman kullanıcılar çalışmaya bilir



Bilgisayar Ağlarının Temelleri

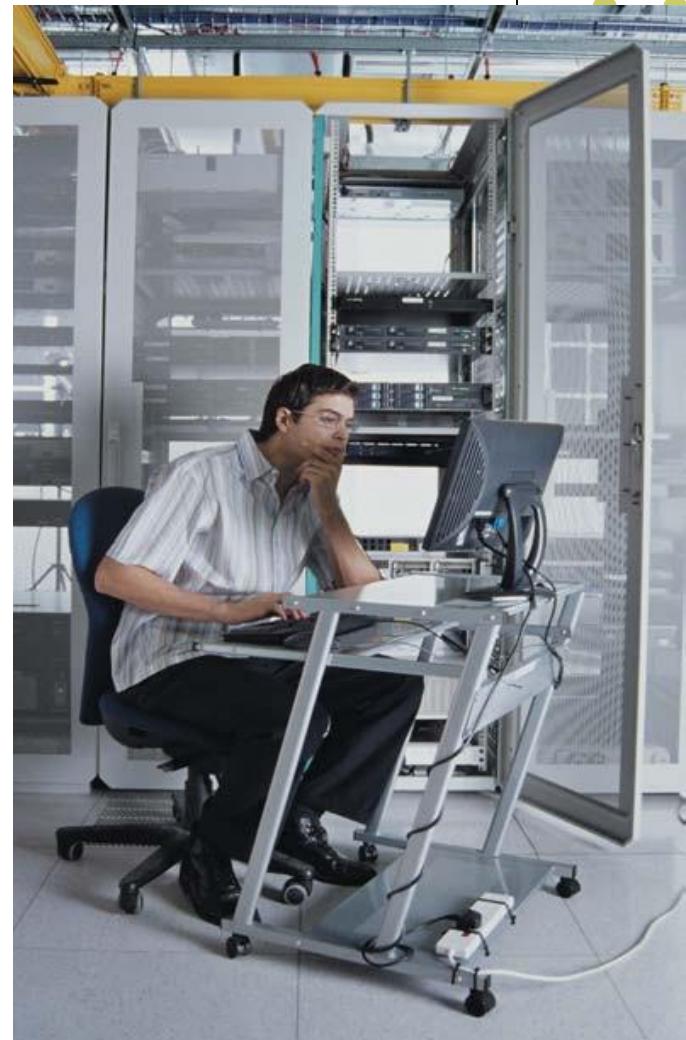
- **Bilgisayar Ağı:** Karşılıklı veri aktarımı ve kaynak paylaşımı yapmak üzere iki veya daha fazla bilgisayarın birbiri ile bağlantısı
- **LAN(Local Area Network):**
Yerel Alan Ağı
 - Küçük coğrafi alanda çalışır. Ör. Bir grup binada
- **WAN(Wide Area Network):**
Geniş Alan Ağı
 - Geniş bir coğrafi alanda çalışır





Ağ Yöneticisi

- Büyük bilgisayar ağları network yöneticisi tarafından yönetilir
 - Yeni yazılım kurma
 - Bakım/Onarım
 - Kullanıcı desteği
 - Kullanıcılarla etkileşim
 - Problem çözme

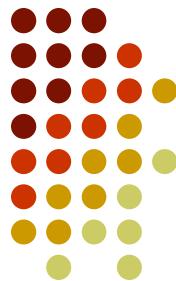




İletişim Cihazları

- İletişim Cihazları:
Bilgisayarlar,
modemler,
yönlendiriciler
(routers), anahtarlar
(switches), kablosuz
erişim noktası (wireless
access points), NIC
(Network Interface
Cards)





İletişim Cihazları

- **Yönlendirici (Routers)** iki veya daha fazla LAN veya WAN'ı birbirine bağlar
 - Verinin yönlendirileceği en iyi rotayı(hız açısından) belirler
- LAN lerde yönlendiricilere benzer anahtarlar da kullanılır

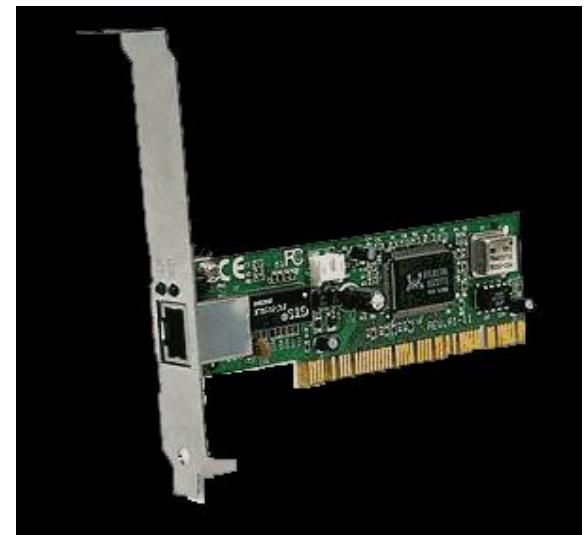




İletişim Cihazları

- **Network Interface Card (NIC)**

- Bilgisayarın ağa bağlanmasıını sağlar
- Bilgisayarın genişleme yuvasına (expansion slot) takılır
- İş istasyonlarında NIC lar: 10Mbps veya 100Mbps
- Sunucularda NIC lar: 1Gbps

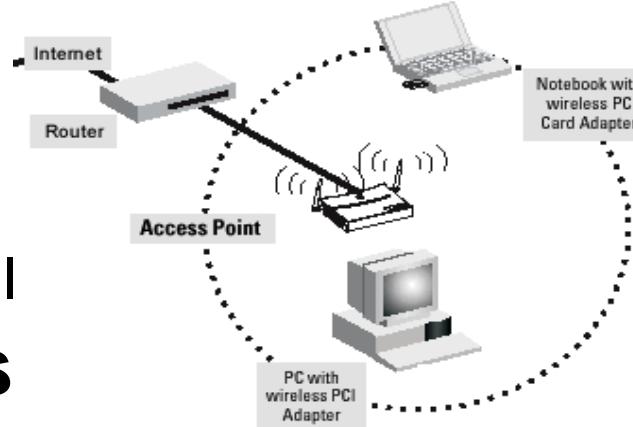


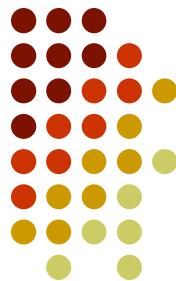


İletişim Cihazları

- **Kablosuz Erişim Noktası
(Wireless Access Points)**

- Kablosuz cihazlar bağlanır
- Genellikle kablolu bir ağa bağlanılır
- Kablolu ağlar arasında bir köprü olabilir
- Laptolar arasında kablosuz bağlantı için kullanılabilir

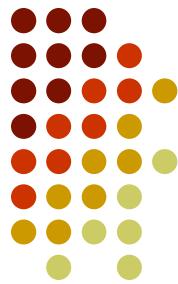




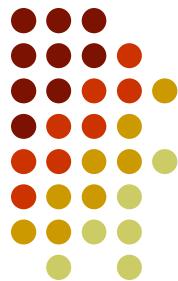
Sunucular (Servers)

- Ağdaki bilgisayarlar özel görevler için kullanılabilir
- Dosya Sunucusu (File servers)
 - Ağ kullanıcılarının erişebileceğи dosyalar yüklenir
 - Yüksek hız, büyük kapasite
- Yazıcı Sunucusu (Print servers)
 - Ağa bağlı yazıcıya erişimi sağlar
- Uygulama Sunucusu (Application Servers)
- Veritabanı Sunucusu (Database Servers)

Yerel Alan Ağları (Local Area Networks) LAN



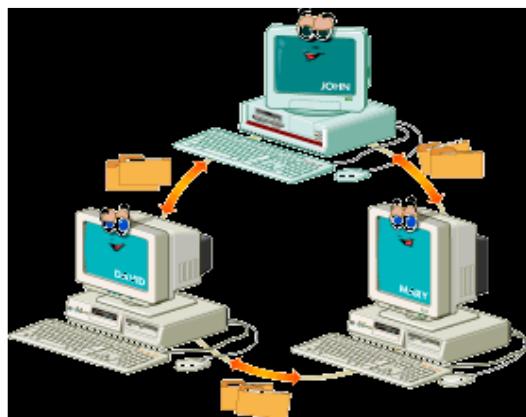
- Bir LAN yakın düğümleri bağlar
 - Bu düğümler iş istasyonu, sunucu, yazıcı, printer, fax makinesi veya yedekleme ünitesi olabilir
Örnekler: Evdeki bilgisayar ağı, bilgisayar laboratuvarı, internet cafe
 - Her düğüm bir **adres** ile ifade edilir
- **Kablosuz LAN lar radyo dalgaları kullanır**
 - Hareketli düğüm noktaları için kullanılır



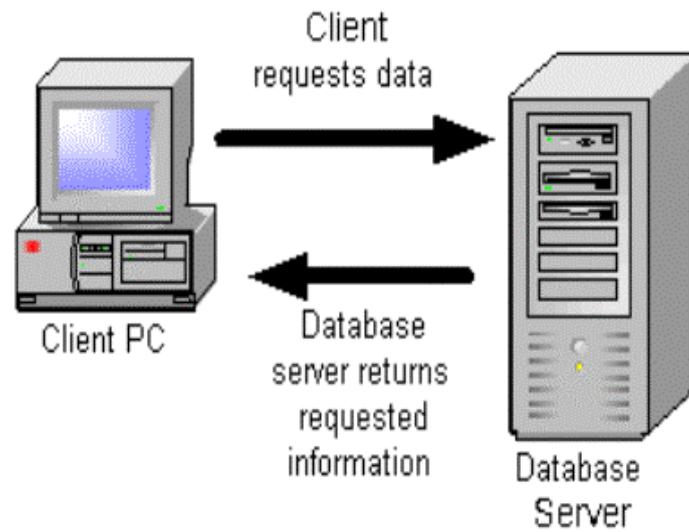
Ağ Modelleri

LAN için iki ana ağ modeli vardır

1) Noktadan Noktaya
(Peer to Peer (P2P))



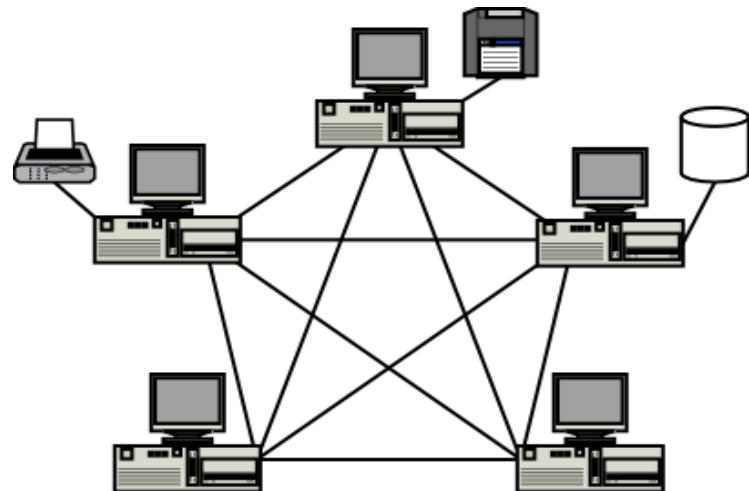
2) İstemci/Sunucu
(Client/server)





Noktadan Noktaya (Peer to Peer) Ağlar

- Ağa bağlı bütün bilgisayarlar eşit hakkı sahiptir
 - Ayrılmış bir sunucu bilgisayar yoktur.
- Bilgisayarlar ağdaki veri ve çevre birimlerini bu düğüme verilen bir takım izinlerle ortak olarak kullanırlar
- Kurmak kolaydır
 - Ev ve küçük ofisler için uygundur
- Büyük Ağlarda yavaşlamaya neden olur

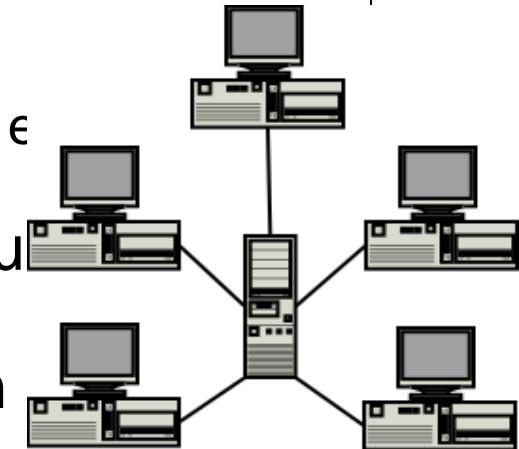


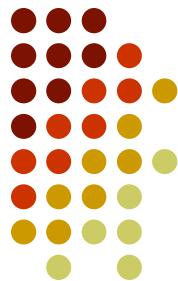
İstemci/Sunucu (Client/Server)

Ağlar



- **Sunucu** ile
 - Dosya, veritabanı, uygulama, çevrebirimleri ve e-posta sunucusu gibi yazılımlara erişebilir
- **İstemci** iş istasyonu tarafından sunucu bilgisayara istek gönderilir
- Sunucu bilgisayar pasiftir ve istemciden gelecek istekleri bekler. İstekleri gerçekleştirir ve sonucu istemci bilgisayara gönderir.
- Ağın büyülüğu sorun oluşturmaz. Ölçeklenebilir (“scalable”).
- Çok sayıda düğüm olsa da sistem yavaşlamaz.





Protokol

- Protokol aşağıda sıralanmış işleri gerçekleştirmek için standart kurallardan oluşur;
 - Veriyi göstermek
 - Veriyi iletmek
 - Doğrulama (authentication)
 - Hata düzeltme (error correction)
 - Ortak protokolü kullanan farklı cihazların düzgün haberleşmesini sağlamak
- Farklı işler için farklı protokoller bulunmaktadır.
- En sık Kullanılan Protokol: TCP / IP

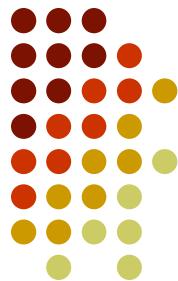


Sayı sistemleri

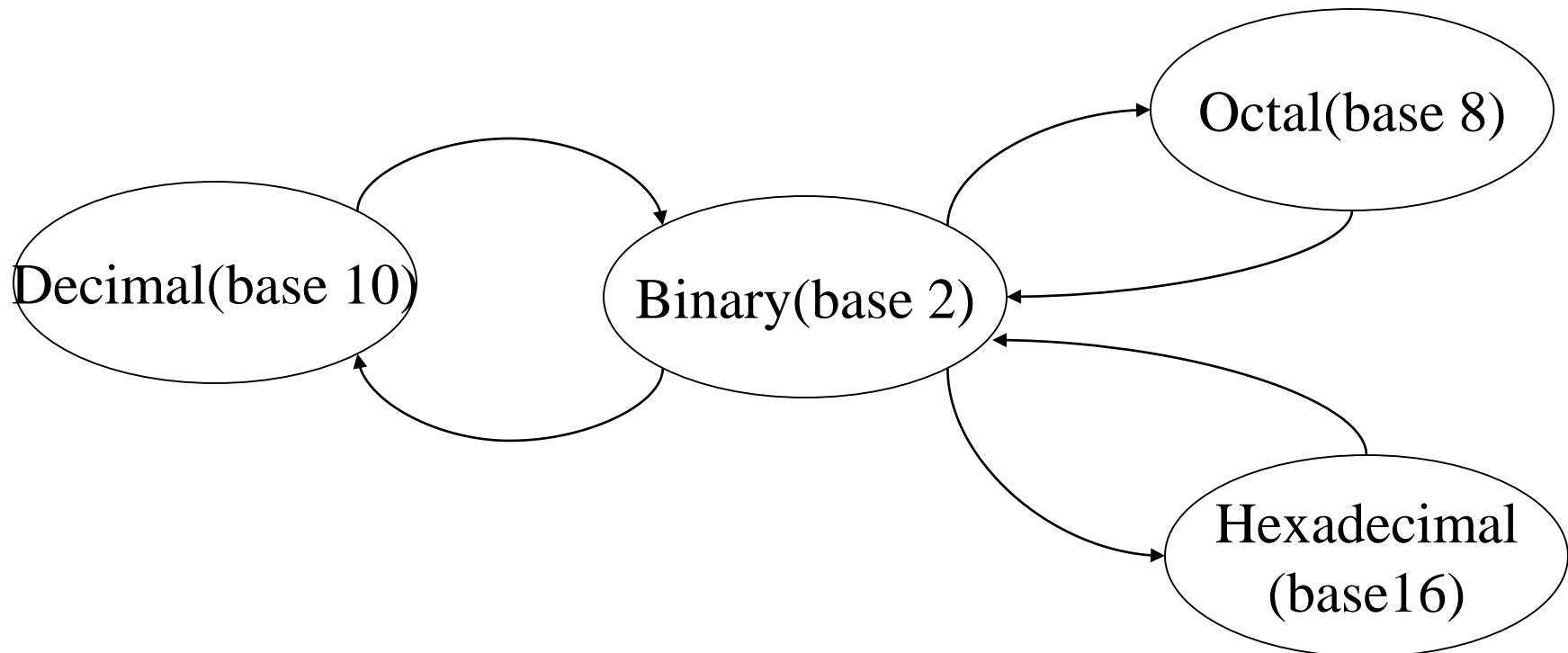


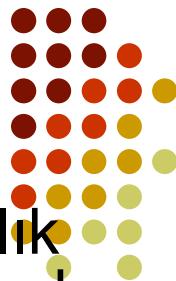
Sayı sistemleri

- Sayı sistemleri Dijital Elektronliğin temelidir.
- Dijital elektronikte dört çeşit sayı sistemi kullanılmaktadır. Bunlar :
 - a) - Desimal Sayı Sistemi
 - b) - Binary Sayı Sistemi
 - c) - Oktal Sayı Sistemi
 - d) - Hexadesimal Sayı Sistemi



Sayı sistemleri ve kodlar (Number systems and codes)



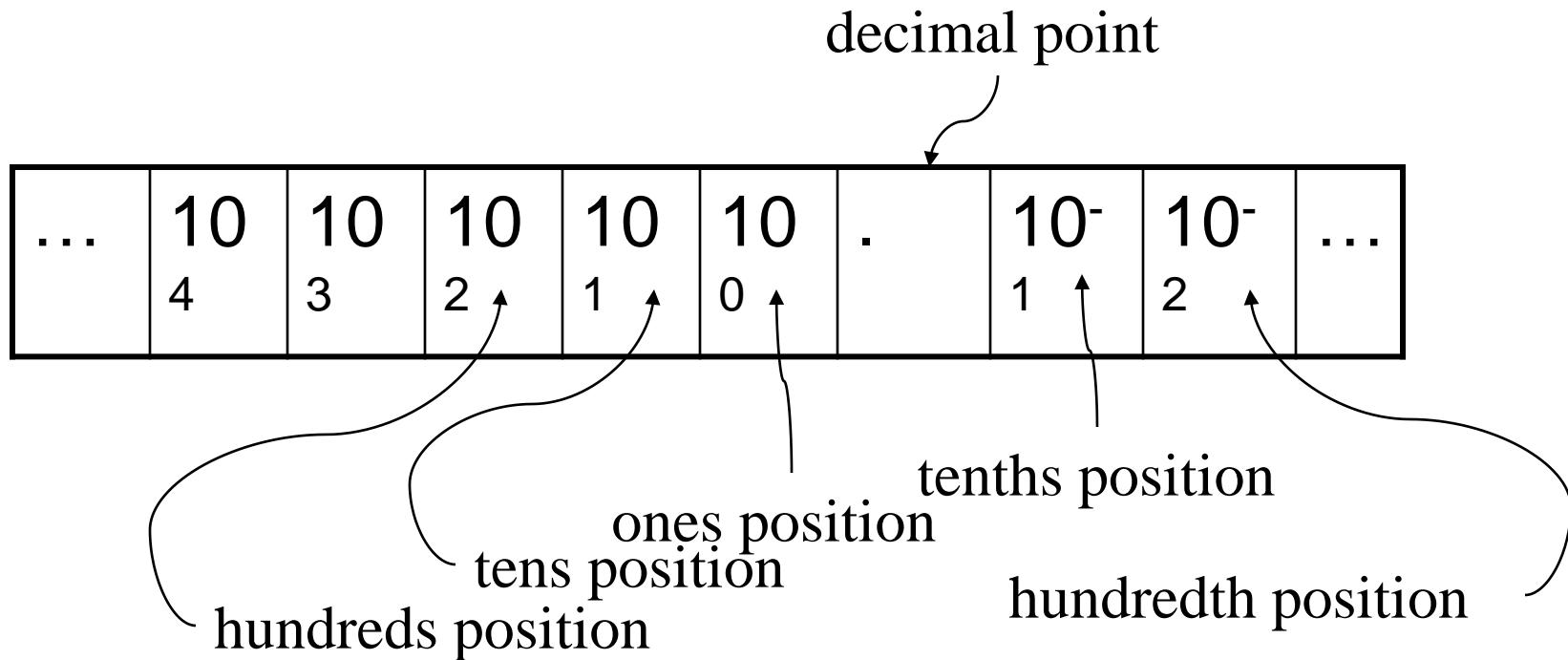


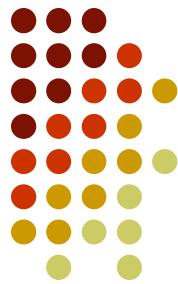
Onluk Sayı Sistemi

- Günlük yaşantımızda kullandığımız sayı sistemi ondalık (decimal) sayı sistemidir. Ayrıca 10 tabanlı sistem olarak da adlandırılır ve bu sistemde on tane sembol kullanılır.
- Taban : 10
- Semboller : 0,1,2,3,4,5,6,7,8,9
- Ondalık sayı sisteminin genel biçimi ve terminolojisi aşağıda verilmiştir.

Tam Kısım	Kesir Kısımı	
2 3 4	. 5 6	
↑	↑	
En büyük Değerli Basamak (Most Significant Digit, MSD)	Ondalık Nokta (Decimal Point, DP)	En Küçük Değerli Basamak (Least Significant Digit, LSD)

Decimal Positional System (Base 10 or radix 10)





Onluk Sayı Sistemi

$$234.56_{10} = \begin{matrix} & \text{Basamak} & \text{Basamak} \\ & \text{Değeri} & \text{Ağırlığı} \\ \downarrow & & \swarrow \\ 2 \times 10^{+2} + & 3 \times 10^{+1} + & 4 \times 10^{+0} + & 5 \times 10^{-1} + & 6 \times 10^{-2} \\ & \uparrow & & & \\ & \text{Taban} & & & \\ & \text{Değeri} & & & \end{matrix}$$

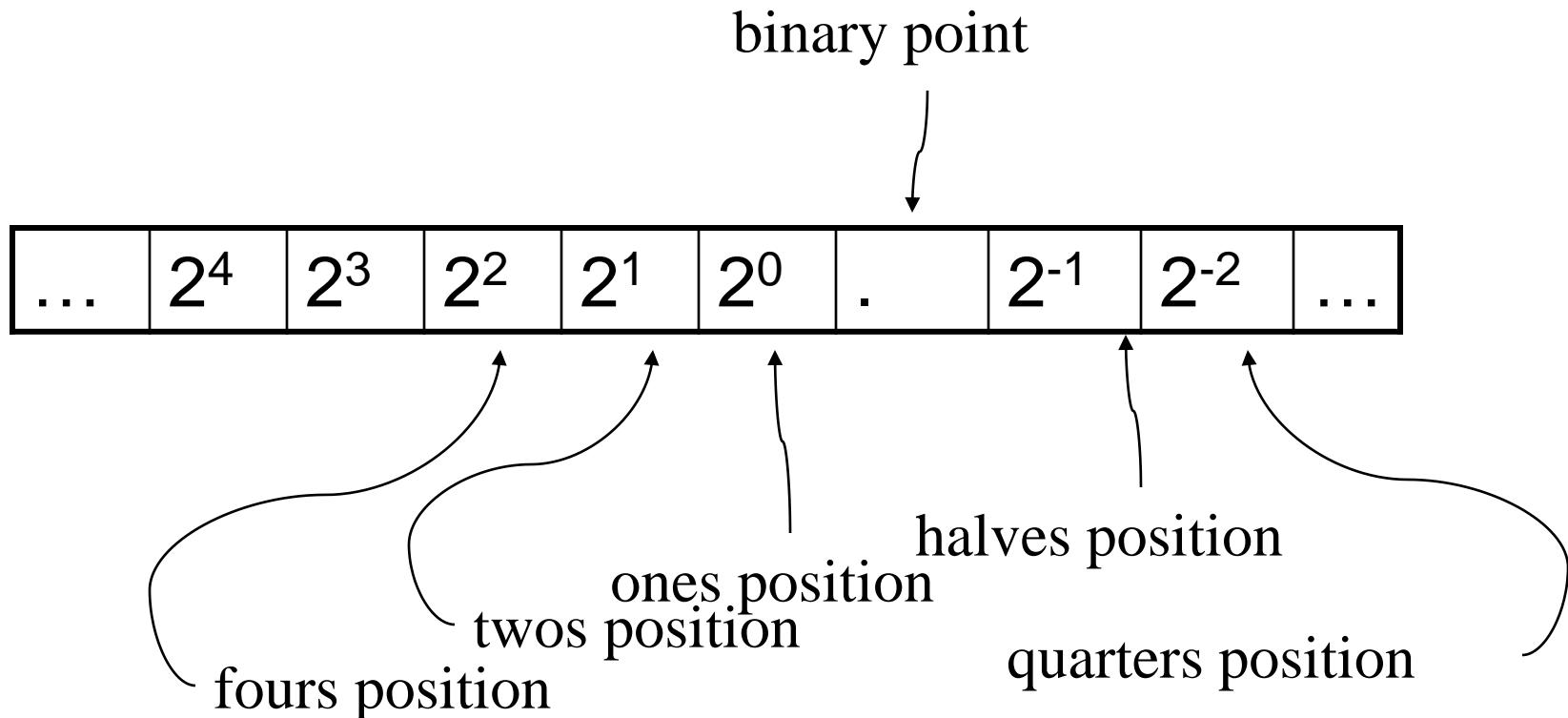


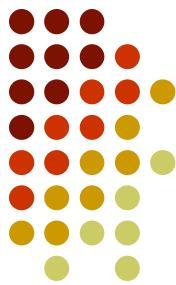
İkili Sayı Sistemi

- İkili (Binary) sayı sistemi, sayısal elektronik sistemlerinde yaygın olarak kullanılır. Günlük yaşamımızda kullandığımız ondalık sayı sisteminden iki yönlü dönüşüm yapılarak kullanılır. Bu sistemde, Boole cebrinde doğru ve yanlışı belirtmek üzere iki tane sembol kullanılır.
- Taban : 2
- Semboller : 0,1
- İkili sayı sisteminin genel biçimi ve terminolojisi aşağıda verilmiştir.

1 0 1 1 . 0 1	 	
↑	↑	↑
En büyük Değerli Bit (Most Significant Bit, MSB)	İkili Nokta (Binary Point, BP)	En Küçük Değerli Bit (Least Significant Bit, LSB)

Binary Positional System (Base 2 or radix 2)





İkili Sayı Sistemi

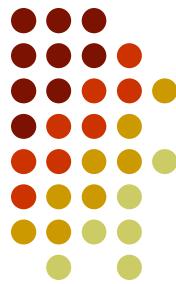
Basamak Değeri Basamak Ağırlığı

↓

$1101.01_2 = 1 \times 2^{+3} + 1 \times 2^{+2} + 0 \times 2^{+1} + 1 \times 2^{+0} + 0 \times 2^{-1} + 1 \times 2^{-2}$

↑

Taban Değeri



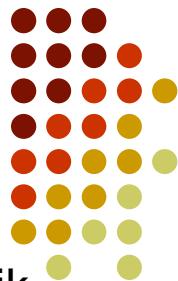
Örnek

Onlu sayılaraya örnek

$$\begin{aligned}346.17_{10} &= (3 \times 10^2) + (4 \times 10^1) + (6 \times 10^0) + (1 \times 10^{-1}) + (7 \times 10^{-2}) \\&= 300 \quad + \quad 40 \quad + \quad 6 \quad + \quad 0.1 \quad + 0.07\end{aligned}$$

İkili sayılaraya örnek

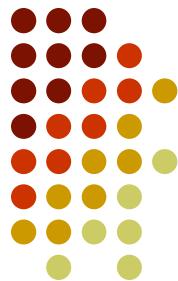
$$\begin{aligned}1101.01_2 &= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} \\&= 8 \quad + \quad 4 \quad + 0 \quad + \quad 1 \quad + \quad 0 \quad + .25 \\&= 13.25_{10}\end{aligned}$$



Sekizli Sayı Sistemi

- Sekizli (Octal) sayı sistemi, sayısal elektronik sistemlerinde ses ve müzik uygulamalarında yaygın olarak kullanılır. Müzikte kullanılan notalara (do re mi fa sol la si do) karşı gelmek üzere sekiz symbol kullanılır. Günlük yaşamımızda kullandığımız ondalık sayı sisteminden iki yönlü dönüşüm yapılarak kullanılır.
- Taban : 8
- Semboller : 0,1,2,3,4,5,6,7
- Sekizli sayı sisteminin genel biçimi ve terminolojisi aşağıda verilmiştir :

7 ↑ En büyük Değerli Basamak (Most Significant Digit, MSD)	0 3 . 1 5 ↑ Sekizli Nokta (Octal Point)	En Küçük Değerli Basamak (Least Significant Digit, LSD)
---	--	--



Sekizli Sayı Sistemi

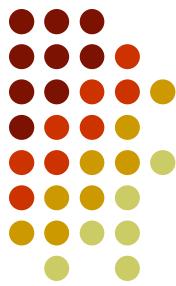
Basamak Değeri Basamak Ağırlığı

↓ ↙

$$703.15_8 = 7 \times 8^{+2} + 0 \times 8^{+1} + 3 \times 8^{+0} + 1 \times 8^{-1} + 5 \times 8^{-2}$$

↑

Taban Değeri



Sekizli sayıların Onlu sayılara Çevirimi (Octal-to-Decimal Conversion)

- Octal-to-Decimal Conversion

$$372_8 = 3 \times (8^2) + 7 \times (8^1) + 2 \times (8^0)$$

$$= 3 \times 64 + 7 \times 8 + 2 \times 1$$

$$= 250_{10}$$

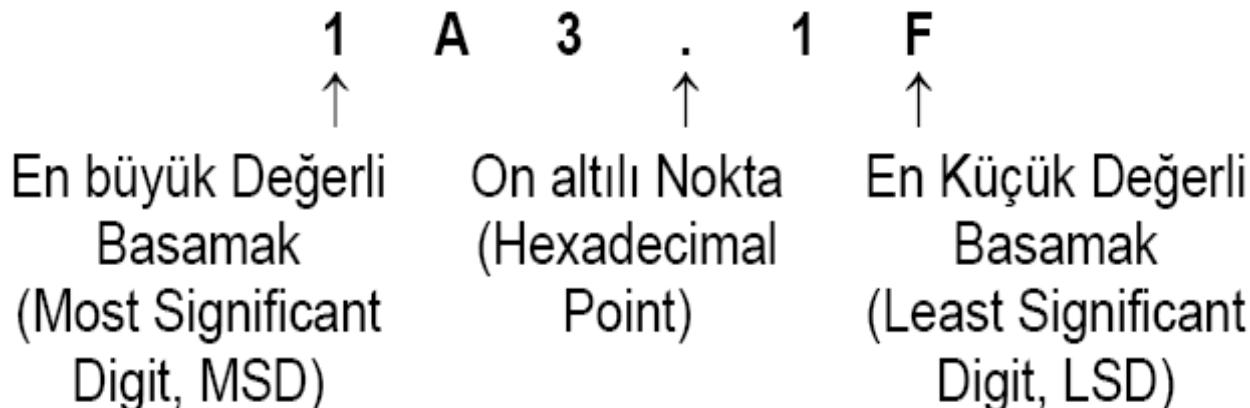
$$24.6_8 = 2 \times (8^1) + 4 \times (8^0) + 6 \times (8^{-1})$$

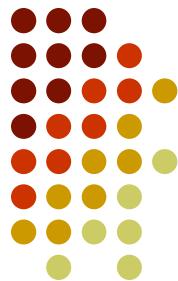
$$= 20.75_{10}$$



Onaltılık Sayı Sistemi

- Onaltılık (Hexadecimal, Hex) sayı sistemi, sayısal elektronik sistemlerinde mikroişlemci temelli uygulamalarda yaygın olarak kullanılır. Günlük yaşamımızda kullandığımız ondalık sayı sisteminden iki yönlü dönüşüm yapılarak kullanılır. Bu sistemde, ondalık sayı sisteminde kullanılan sembollere ek olarak, dokuzdan büyük değerlere karşılık İngiliz alfabetesinin ilk beş harfi ile birlikte on altı tane simbol kullanılır.
- Taban : 16
- Semboller : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
- Onaltılık sayı sisteminin genel biçimi ve terminolojisi aşağıda verilmiştir.





Onaltılık Sayı Sistemi

Basamak Basamak
Değeri Ağırlığı

↓ ↗

$$1A3.1F_{16} = 1 \times 16^{+2} + 10 \times 16^{+1} + 3 \times 16^{+0} + 1 \times 16^{-1} + 15 \times 16^{-2}$$

↑
Taban
Değeri

BİLGİSAYAR PROGRAMLAMA

Algoritma ve Akış Şemaları

Konu Başlıklarısı

- Algoritma tanımı
- Algoritma özellikleri
- Algoritma tasarıımı
- Akış şemaları
- Dallanma simgeleri
- Döngü simgeleri
- Akış şeması tasarıımı
- Akış şeması örnekleri

Algoritma

- Herhangi bir işi yapmak ya da bir problemi çözmek için adım adım uygulanan kurallar dizisine algoritma denir.
- Algoritmada adımlar sırasıyla işlenmektedir.

Algoritma

- Bir algoritmanın en önemli özellikleri:
 - bir giriş verisine karşılık çıkış bilgisinin mutlaka bulunması,
 - sonlu sayıda adım içermesi,
 - her türlü alternatifin düşünülerek sonuca ulaşıldığının garanti edilmesidir.
- Eğer sonlu sayıda adım ile Sonuca ulaşılamaz ise istenmeyen Sonsuz döngüler, eğer bütün alternatifler düşünülmmez ise belirsiz durumlar ortaya çıkabilir ve bir sonuca ulaşılamaz.

Algoritma

- Bir kurallar dizisini algoritma olarak tanımlayabilmek için mutlaka giriş verileri değerlendirlerek çıkış bilgisinin elde edilmesi gereklidir.
- Algoritma, belirli biri işin/problemsonucunu elde etmek için art arda uygulanacak adımları ve koşulları kesin-olarak ortaya koyar. Bu adımlar, ilgili koşullar altında adım adım izlendiğinde bir sonuca ulaşılır.

Algoritmada olması gereken özellikler

- **Etkin ve Genel olma:** her koşulda ve her giriş değerinde doğru sonuca ulaşılabilirdir.
- **Sonlu olma.** Algoritma kesinlikle sonlu sayıda işlem içermeli ve bu işlemlerin süresi de sonlu olmalıdır;
- **Yanılmazlık.** Algoritma tekrar yürütüldüğünde aynı giriş değerleri için aynı sonuç elde edilmelidir.
- **Giriş/Cıkış tanımlı olma.** Algoritmanın giriş ve çıkış değerleri olmalıdır.
- **Başarım.** Algoritma başarımı (disk ve bellek kullanımı) iyi olacak şekilde tasarlanmalıdır.

Akış Şeması

- Sorunun çözümü için oluşturulmuş algoritmanın görsel olarak şekillerle ifade edilmesine akış şeması (flow chart) denilmektedir.
- Akış şemalarında algoritmanın adımları simgeler şeklinde kutular içine yazılmaktadır ve adımlar arasındaki ilişkiler ve akış yönü oklar ile gösterilmektedir.

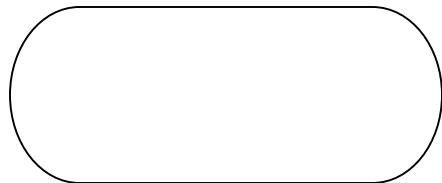
AKİŞ ŞEMASININ AVANTAJLARI

- 1) Birbiri ile ilgili adımlar arasındaki mantıksal ilişkiyi gösterirler.
- 2) İzlenmesi ve anlaşılması kolaydır.
- 3) Şartlara bağlı olarak ortaya çıkan olayların takip edilmesi mümkündür.
- 4) Belli bir standarda göre hazırlandıklarından herkesçe anlaşılabilir ve birden fazla kişi aynı şema üzerinde çalışabilir.

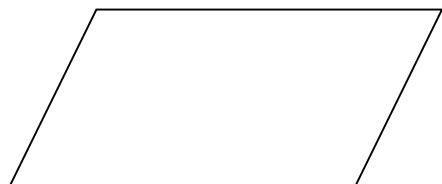
Akış Şeması Hazırlama Kuralları

- 1) Başlangıç ve bitiş uçları tanımlanmalıdır.
- 2) Standart semboller kullanılmalıdır.
- 3) Birbirini kesen akış hatları
kullanılmamalıdır.
- 4) Basit kararlar alınmalıdır.
- 5) Şema belli bir yönde hazırlanmalıdır.

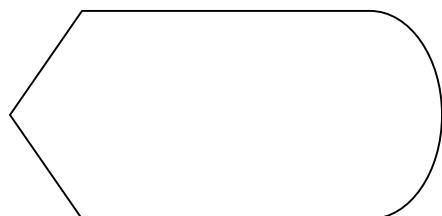
Akış Şeması Elemanları



Akış şemasının başlangıcını
yada bitişini belirtir.

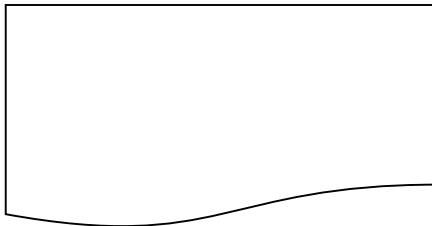


Veri girişi yapılacağını belirtir.



Ekranda görüntüleme
yapılacağını belirtir.

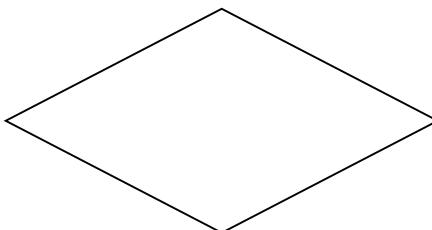
Akış Şeması Elemanları



Yazıcıya çıktı olacağını belirtir.

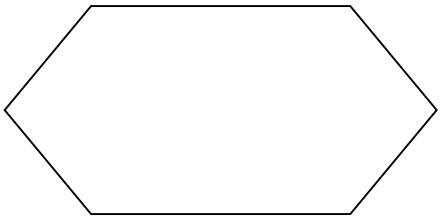


Hesaplama ya da değerlerin
değişkenlere aktarımını gösterir.

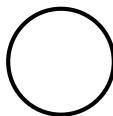


Aritmetik ve mantıksal ifadeler için
karar verme ya da karşılaştırma
durumunu gösterir.

Akış Şeması Elemanları



Yapılacak işler birden fazla sayıda yinelenenecek ise yani iş akışında döngü var ise bu sembol kullanılır.



İki nokta arası ilişkiyi gösterir. Döngü sonunu göstermek için ya da akış şemasının başka bir yere bağlantısını göstermek amacıyla kullanılır.



Fonksiyon çağrırlacağini belirtir.



İş akışının yönünü belirtir.

Akış Şemaları

- Akış şemaları içerik ve biçimlerine göre genel olarak üç grupta sınıflandırılabilir.
 1. Doğrusal akış şemaları
 2. Mantıksal akış şemaları (Koşullu)
 3. Döngü içeren akış şemaları

Doğrusal akış şemaları

- İş akışları, giriş, hesaplama, çıkış biçiminde olan akış şemaları bu grup kapsamına girer.
- Yapısında karar alma ya da döngü ifadeleri içermeyen akış şemalarıdır.

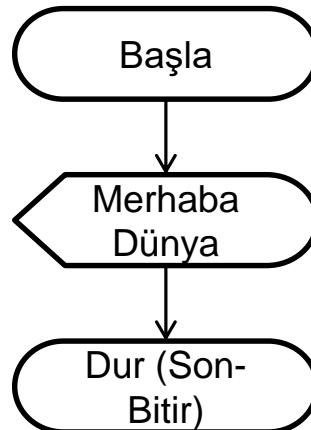
Örnek-1

(Ekrana Merhaba Dünya Yazdırma)

Algoritması

```
Adım 1: Başla  
Adım 2: Ekrana 'Merhaba Dünya' yazdır.  
Adım 3: Dur
```

Akış Şeması (doğrusal)



C Kodu

```
#include <stdio.h>  
void main()  
{  
    printf("Hello World");  
}
```

Örnek-2

(İki Sayıyı Toplama ve Görüntüleme)

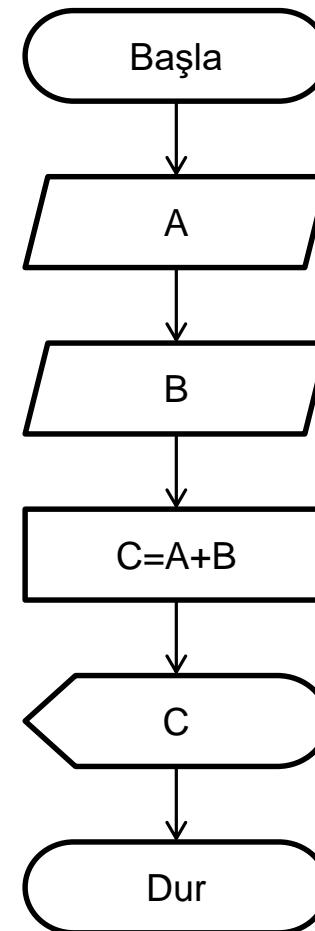
Algoritması

Adım 1: Başla
Adım 2: Birinci sayıyı gir
Adım 3: İkinci sayıyı gir
Adım 4: İki sayıyı topla
Adım 5: Toplam sonucunu ekranda görüntüle
Adım 6: Dur

Değişkenler:
A: Birinci sayı
B: İkinci sayı
C:Toplam sonucu

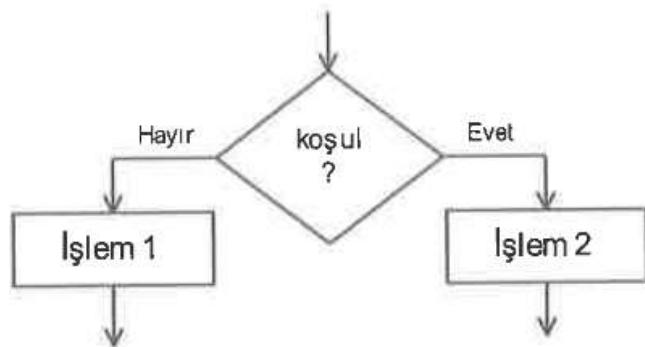
Adım 1: Başla
Adım 2: A'yı gir
Adım 3: B'yı gir
Adım 4: C=A+B
Adım 5: C'yi ekranda görüntüle
Adım 6: Dur

Akış Şeması (doğrusal)

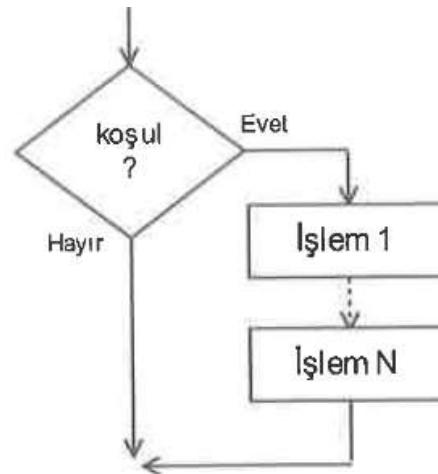


Mantıksal (Koşullu Dallanma)

- Geniş ölçüde mantıksal kararları içeren akış şemalarıdır.
- Hesap düzenleri genellikle basittir.
- Verilen koşulun doğru yada yanlış olmasına göre iş akışı yönlendirilir.



a) Koşulun durumuna bağlı iki farklı işlem var



a) Olumsuz koşulda yapı alacak yok, olumluda N tane işlem var

Örnek-3

(Klavyeden Girilen 2 sayıdan birincisi büyük ise çarpma, aksi durumda toplama yapan algoritma ve akış şeması)

Algoritması

Değişkenler:

X: Birinci sayı

Y: İkinci sayı

S:Toplam sonucu

Adım 1: Başla

Adım 2: X'i gir

Adım 3: Y'yi gir

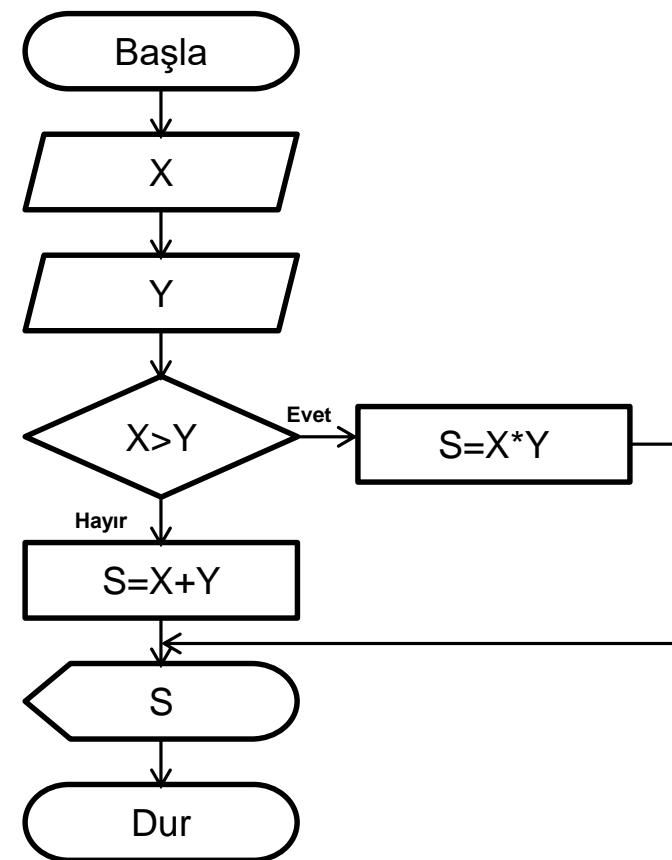
Adım 4: $X > Y$ ise $S = X * Y$ hesapla adım 6'ya git

Adım 5: $X \leq Y$ ise $S = X + Y$ hesapla

Adım 6: S'i görüntüle

Adım 7: Dur

Akış şeması



Örnek: İki sayıdan büyük olanı bulma

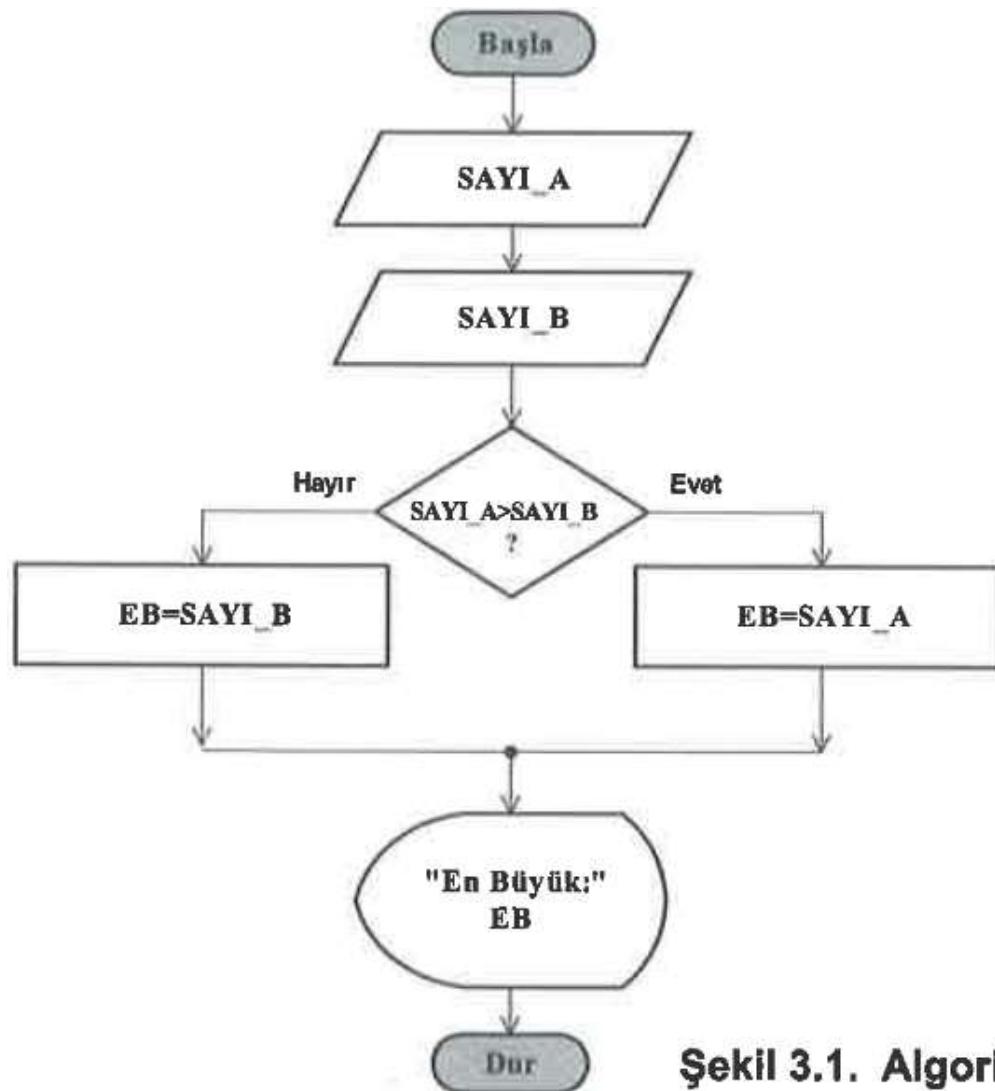
Algoritma bulBuyuk;

/* Bu algoritma iki sayı okur ve büyük olanı bulur; sonucu ekrana yazdırır. */

- ① **Oku (Sayı A);** /* birinci sayıyı oku */
- ② **Oku (Sayı B);** /* ikinci sayısı oku */
- ③ **Karşılaştır (A>B) ?** /* sayıları karşılaştır; A, B'den büyük mü? */
 Evet ➔ EB=A
 Hayır ➔ EB=B
- ④ **Yaz (EB);** /* büyük olan EB'de bunu ekrana yazdır */
- ⑤ **Dur;**

Algoritma 3.1. İki sayıdan büyüğünü bulan algoritma

Örnek: İki sayıdan büyük olanı bulma



Şekil 3.1. Algoritma 3.1'in akış şeması

Döngüsel Akış Şemaları

- Sorunun çözümü için, çözümde yer alan herhangi bir adım ya da aşamanın birden fazla kullanıldığı akış şemalarına denir.
- İş akışları genel olarak giriş ya da başlangıç değeri verme, hesaplama, kontrol biçiminde olmaktadır.

Örnek-4

(Klavyeden 5 kişinin doğum yılını girip 2013 yılındaki yașını hesaplayan algoritma ve akış şeması)

Algoritması

Değişkenler:

dy : kişinin doğum yılı

yas: kişinin yaşı

i : sayaç (5 kişi olup olmadığını
kontrol eder)

Adım 1: Başla

Adım 2: i=1;

Adım 3: dy'yi gir

Adım 4: yas=2013-dy

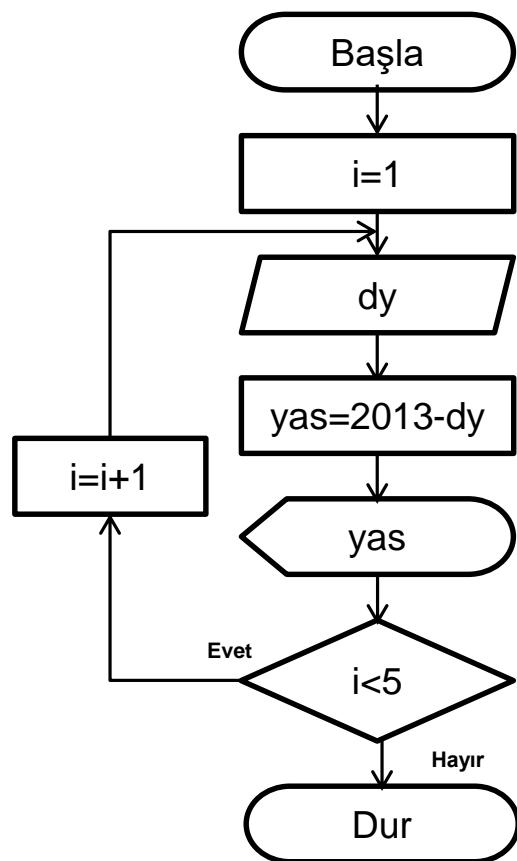
Adım 5: yas'ı ekrana yazdır

Adım 6: i<5 ise i'yi 1 arttır, adım 3'e git

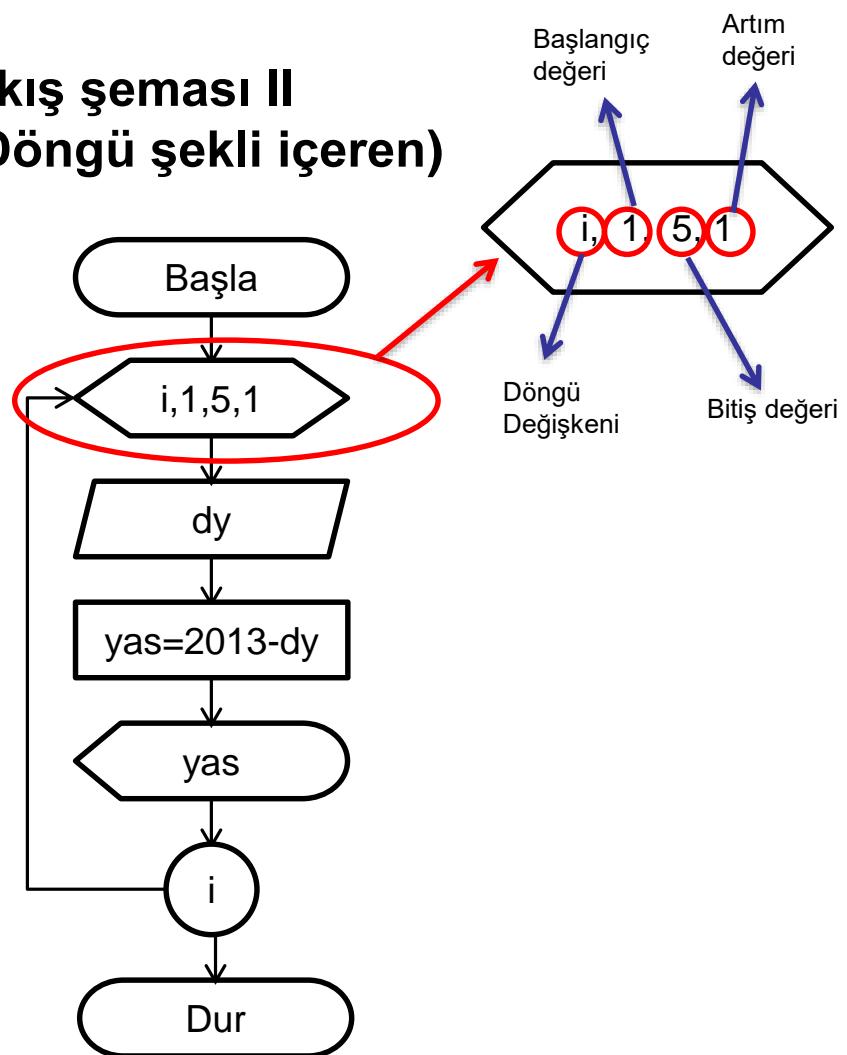
Adım 7: Dur

Örnek 4 (Devam)

Akış şeması I
(Döngü ifadesi içermeyen)



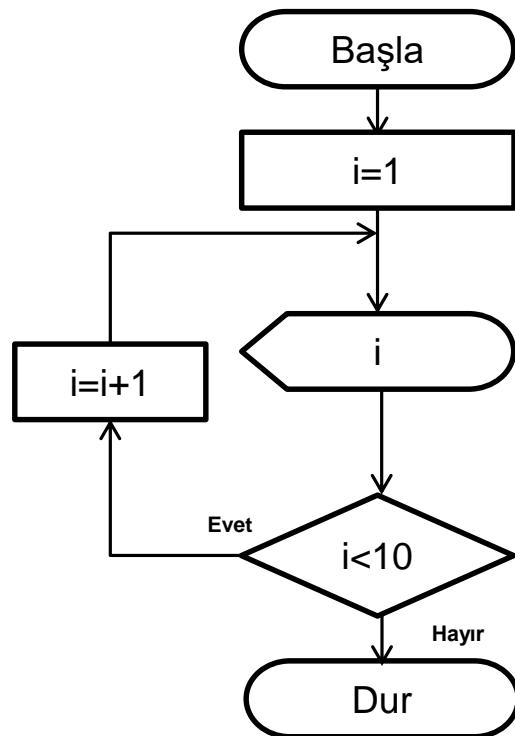
Akış şeması II
(Döngü şekli içeren)



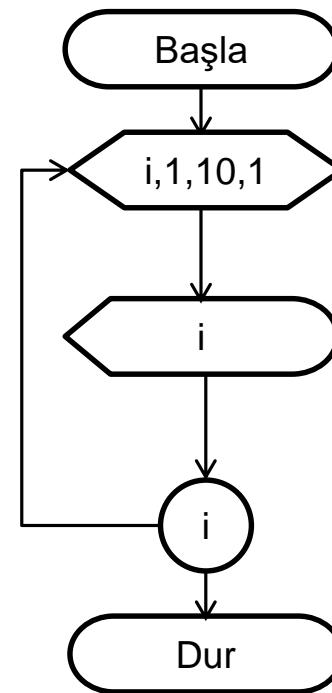
Örnek 5

(1'den 10'a kadar sayıları ekranında görüntüleyen akış şeması)

Kontrol İfadesiyle



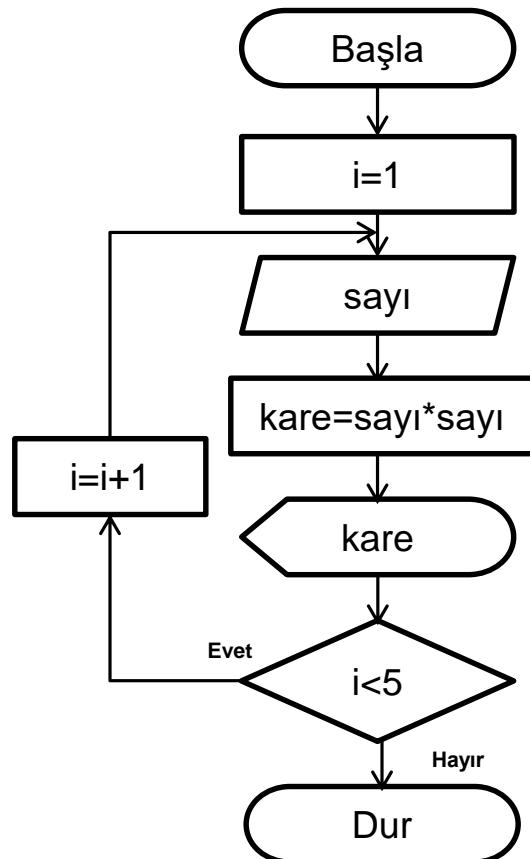
Döngü İfadesiyle



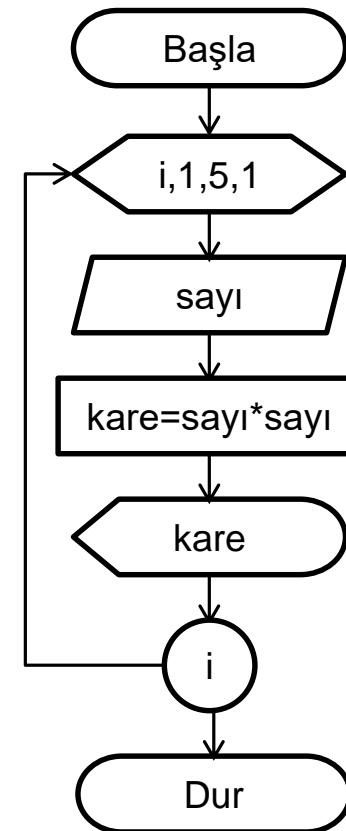
Örnek 6

(Girilen 5 sayının karelerini görüntüleyen akış şeması)

Kontrol İfadesiyle



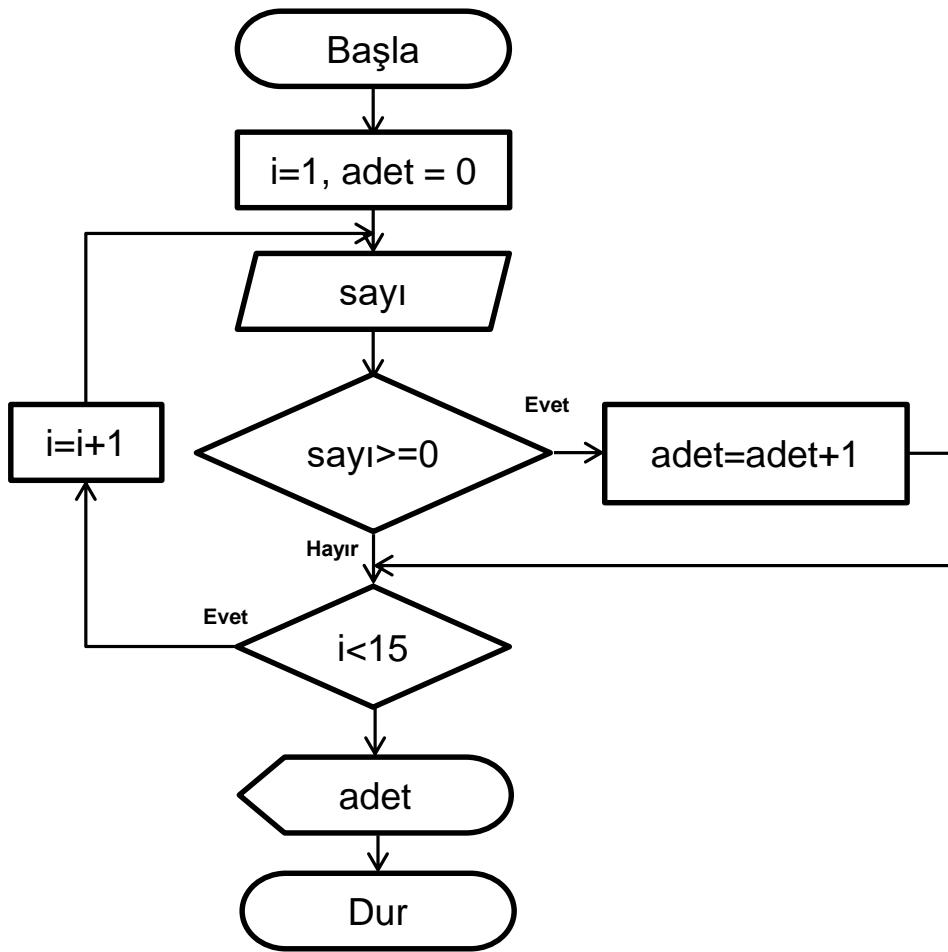
Döngü İfadesiyle



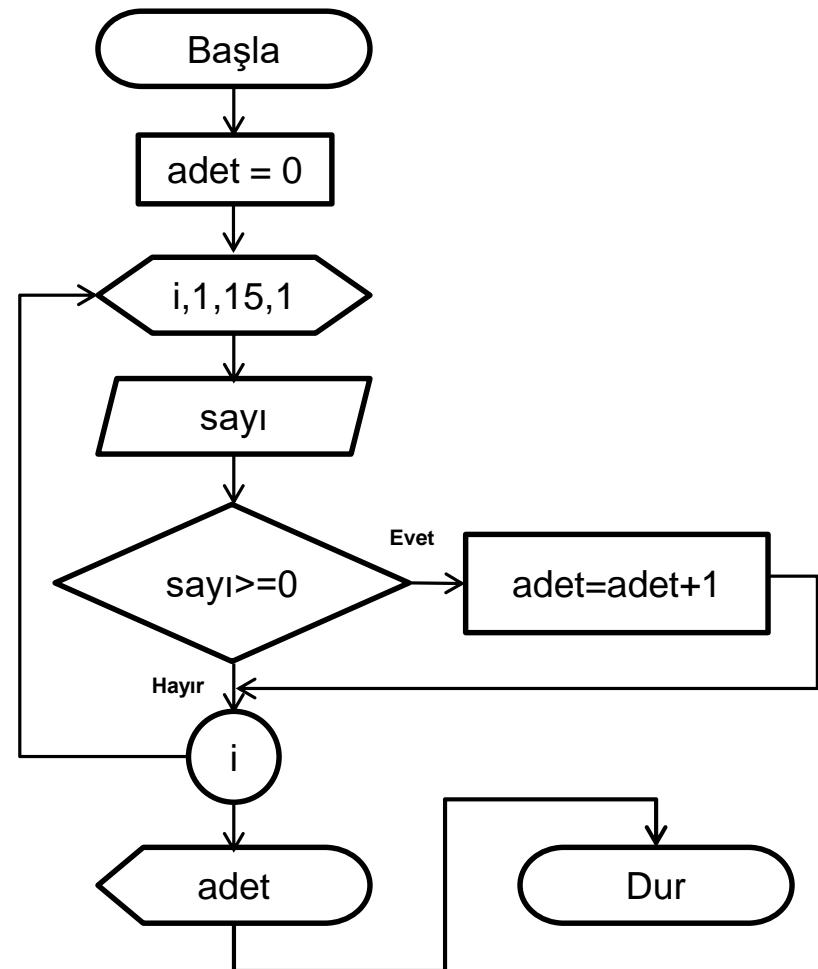
Örnek 7

(Girilen 15 sayıdan pozitif olanların adedini bulup görüntüleyen akış şeması)

Kontrol İfadesiyle



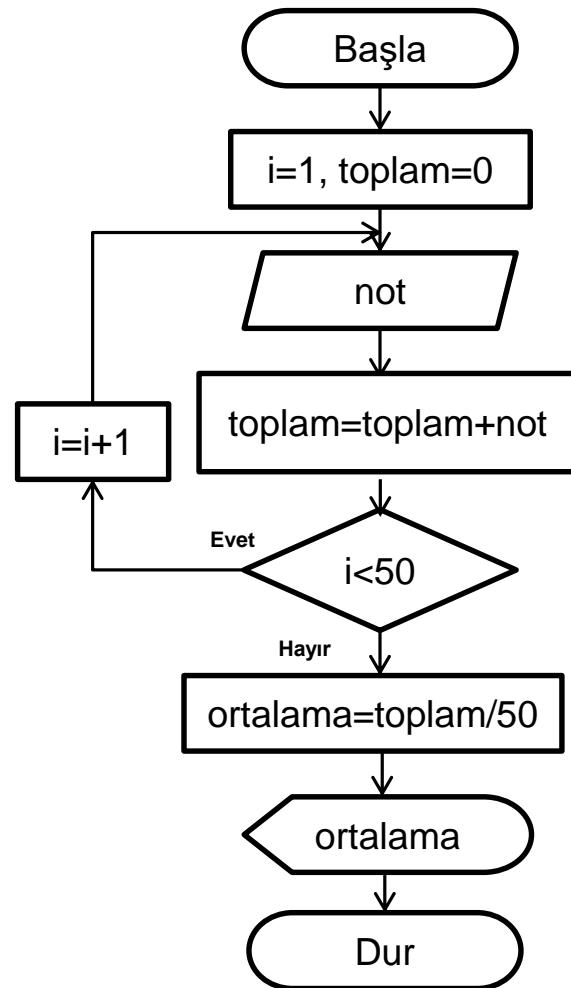
Döngü İfadesiyle



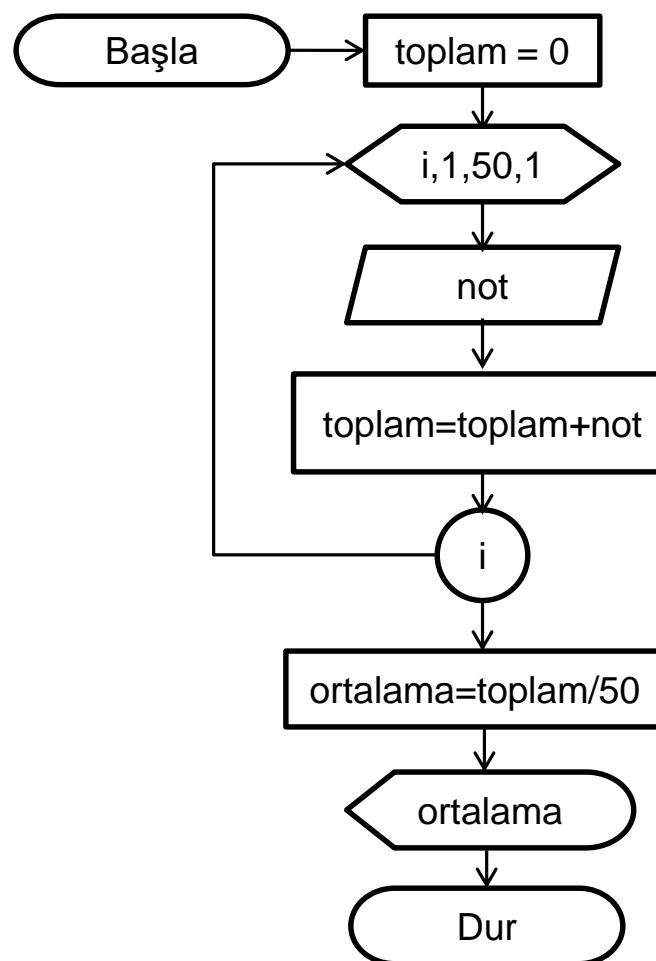
Örnek 8

(50 öğrencinin notlarının ortalamasını bulan akış şeması)

Kontrol İfadesiyle



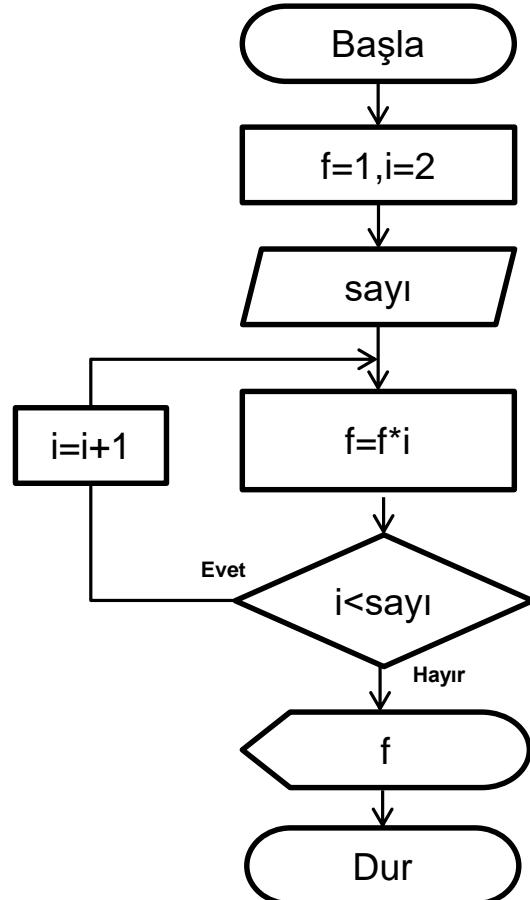
Döngü İfadesiyle



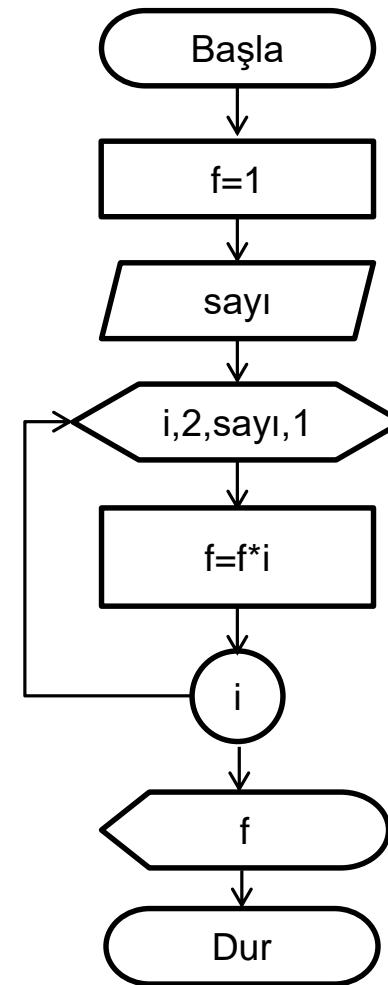
Örnek 9

(Girilen sayının faktöriyelini hesaplayan akış şeması)

Kontrol İfadesiyle



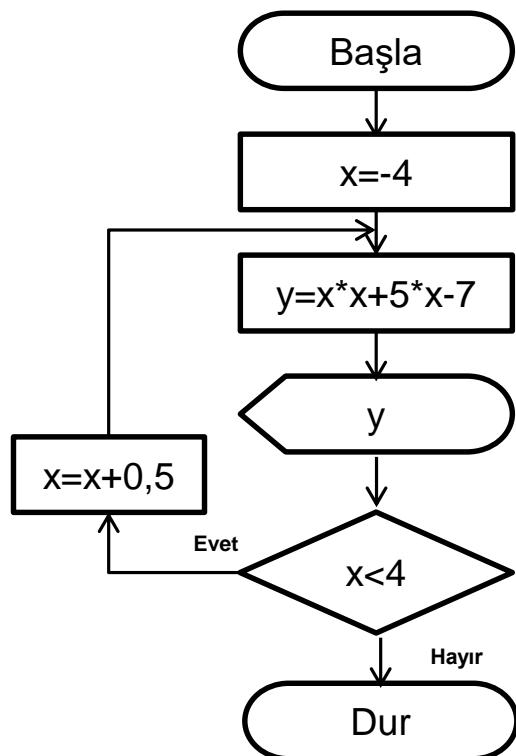
Döngü İfadesiyle



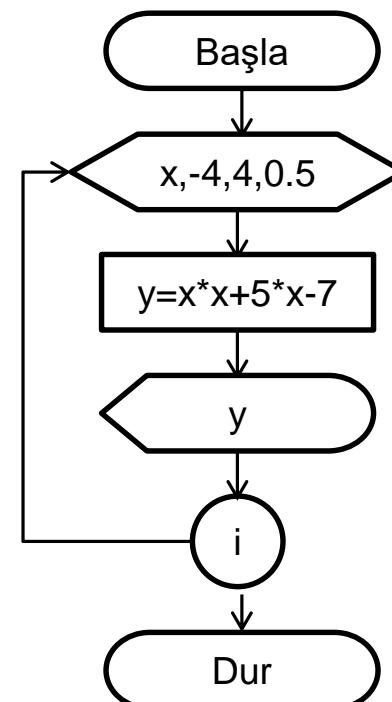
Örnek 10

($y=x^2+5x-7$ denkleminin $x=[-4,4]$ aralığındaki çözümlerini bulan ve görüntüleyen akış şeması, (x 'in artım değeri 0,5'tir.))

Kontrol İfadesiyle



Döngü İfadesiyle

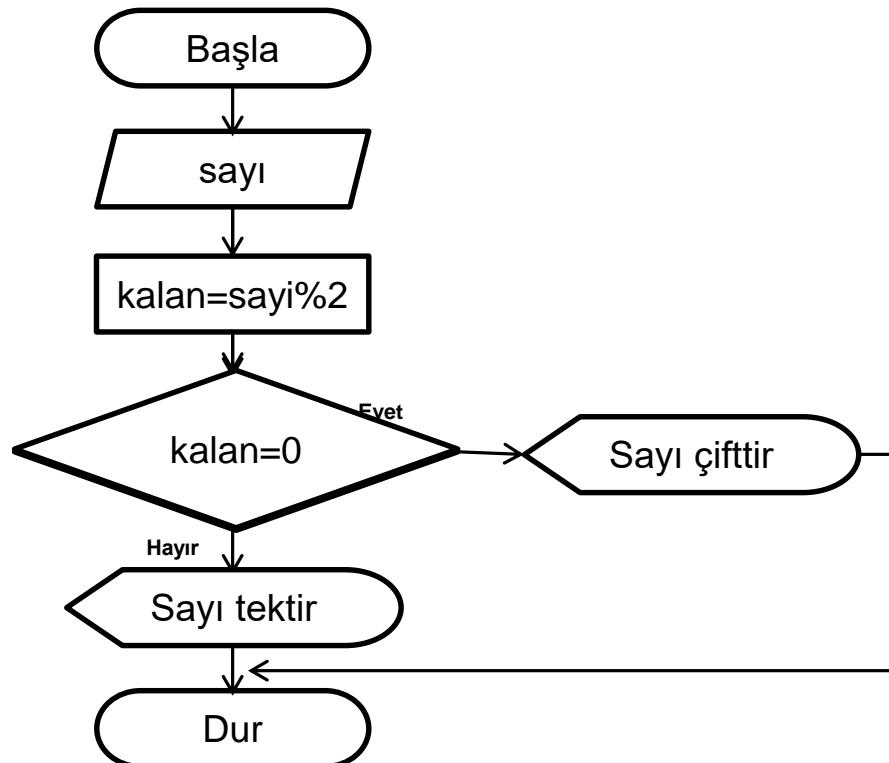


Örnek 11

(girilen sayının tek yada çift olduğunu bulup uygun mesajı görüntüleyen akış şeması)

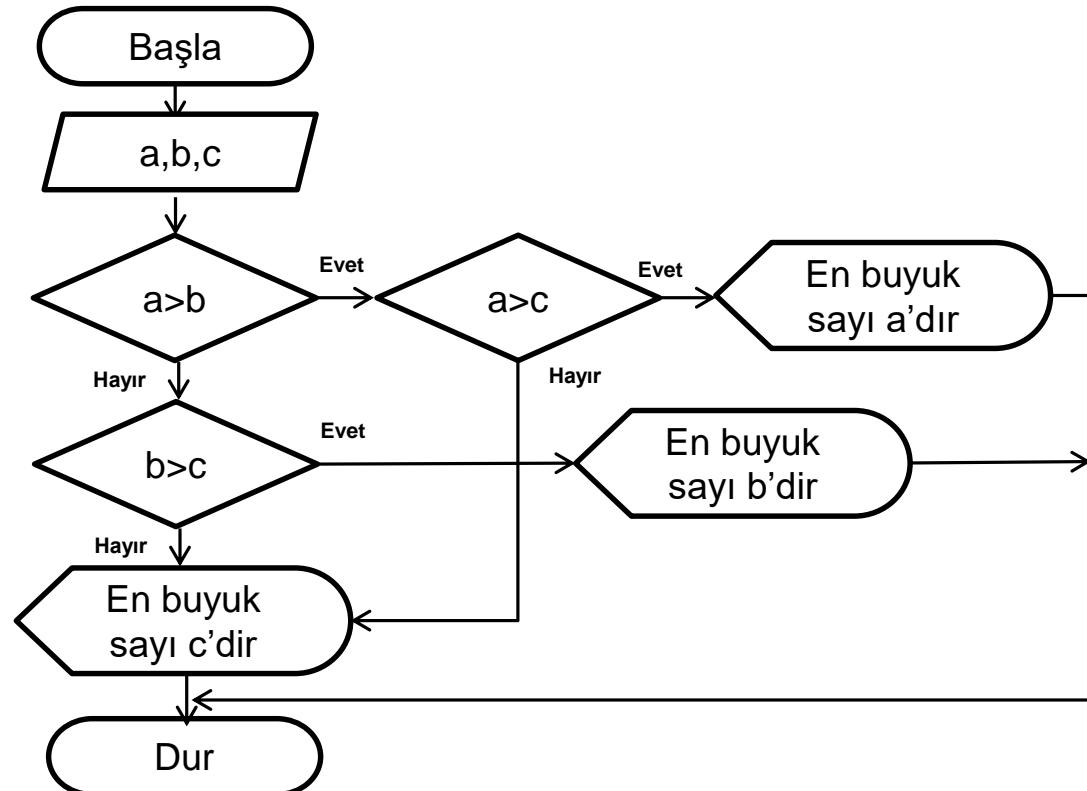
% simgesi mod alma işlemidir. **sayi mod 2 = 0** ise çift sayı olduğunu değilse tek sayı olduğunu ekrana gösteriyoruz.

kalan adında bir değişken kullanmadan, kontrol ifadesi içinde **sayi % 2 = 0** yazmamız da mümkün olabilirdi.



Örnek 12

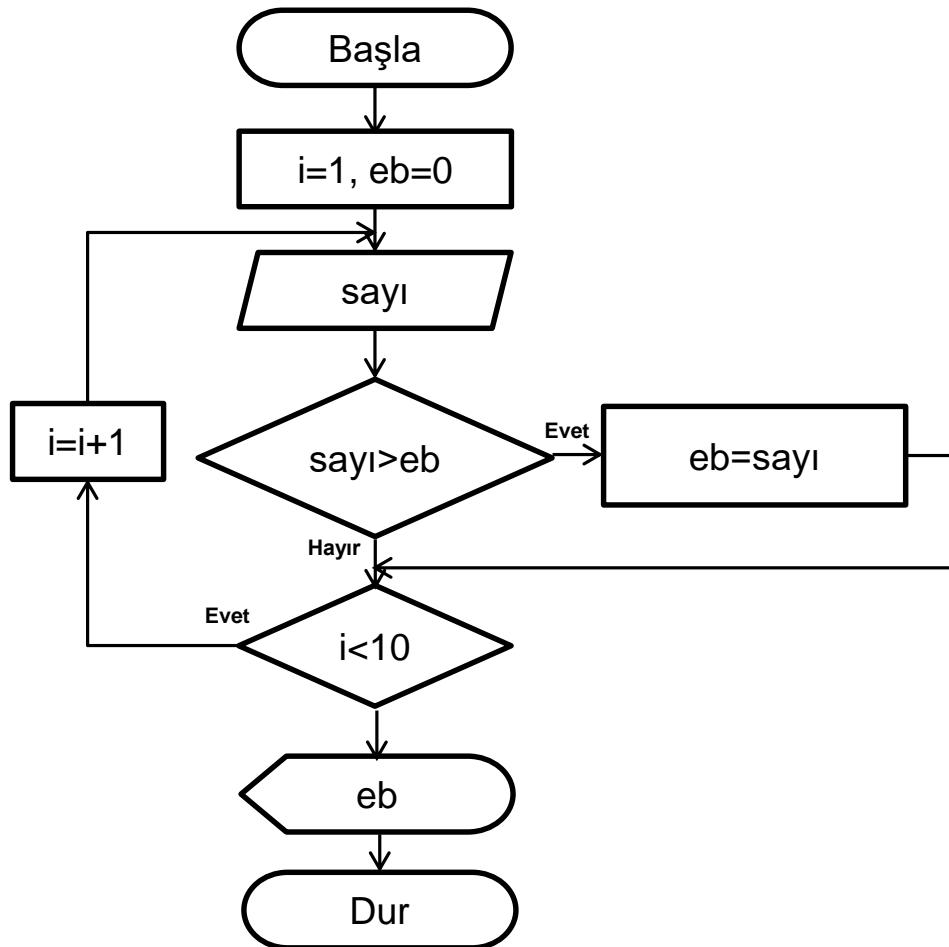
(girilen 3 sayıdan hangisinin en büyük olduğunu bulan akış şeması)



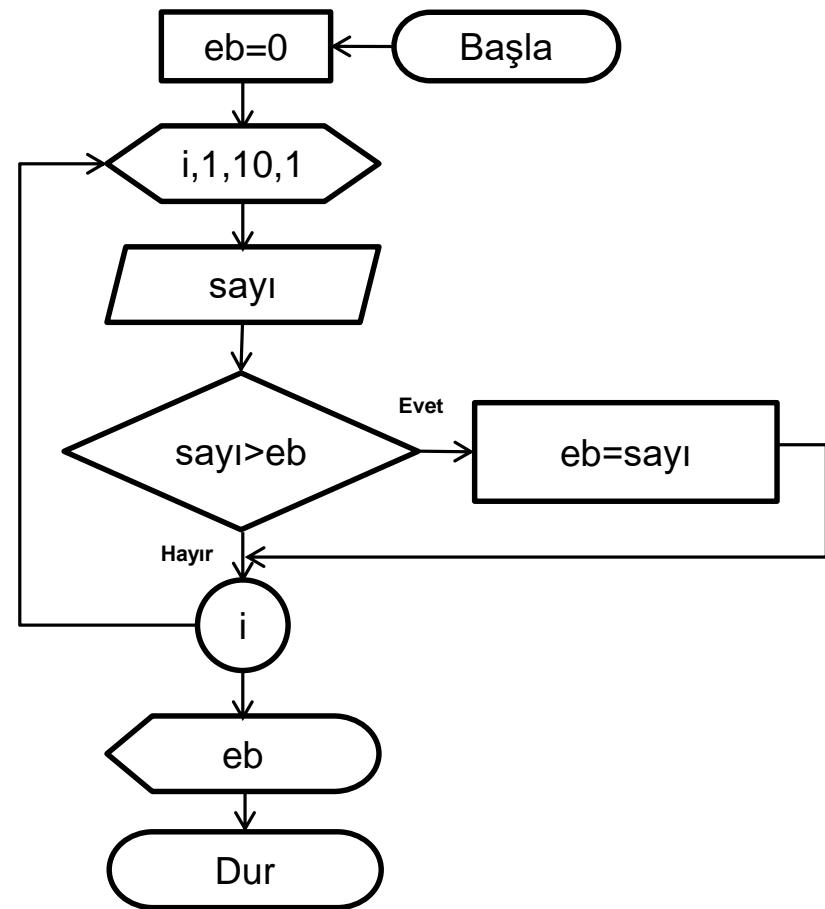
Örnek 13

(girilen 10 sayıdan en büyüğünü bulan ve görüntüleyen akış şeması)

Kontrol İfadesiyle



Döngü İfadesiyle



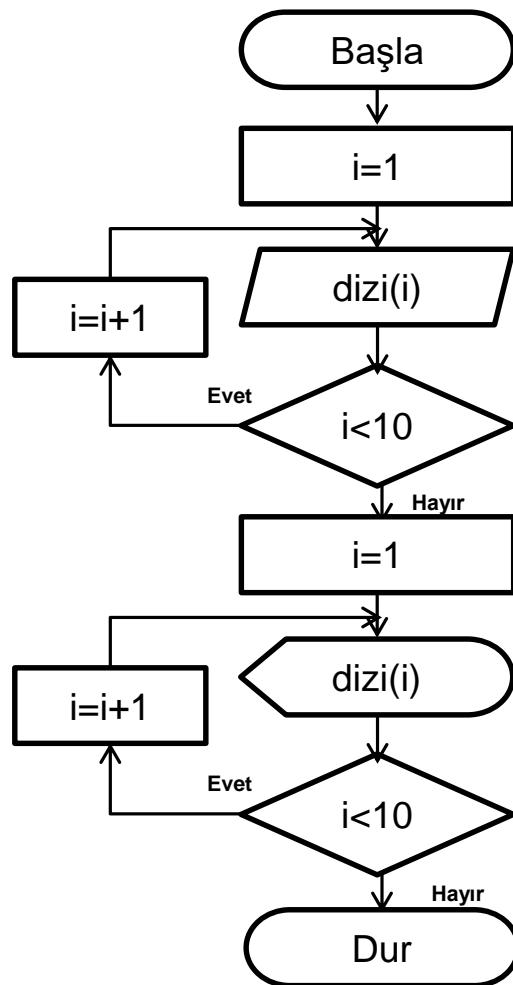
Dizi Kavramı

- Diziler aynı tipteki elemanların yan yana sıralanmasıyla elde edilen bir bilgi kümesidir.
- İki boyutlu (örn: matris) yada daha çok boyutlu diziler de olabilir.
- Diziler akış şemalarında indisli gösterilirler:
 - $\text{dizi}(0)$, $\text{dizi}(1)$, $\text{dizi}(2)$, , $\text{dizi}(i)$
 - ilk elemanın indis akış şemalarında 1 olarak gösterilse de, C programlama dilinde 0'dır.

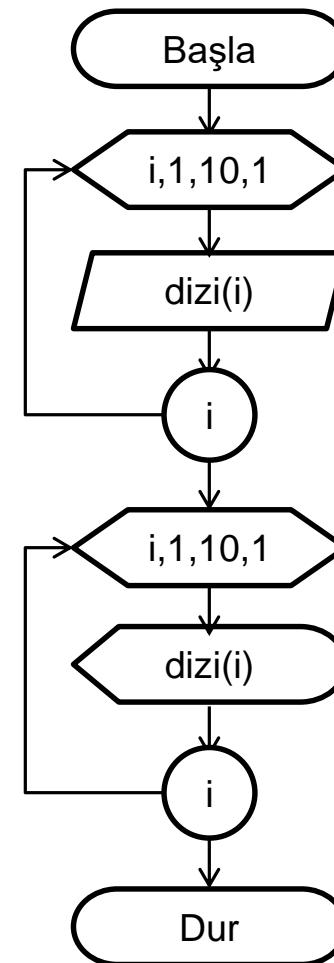
Örnek 15

(10 elemanlı bir diziye bilgi girişi yapan ve diziyi görüntüleyen akış şeması)

Kontrol İfadesiyle

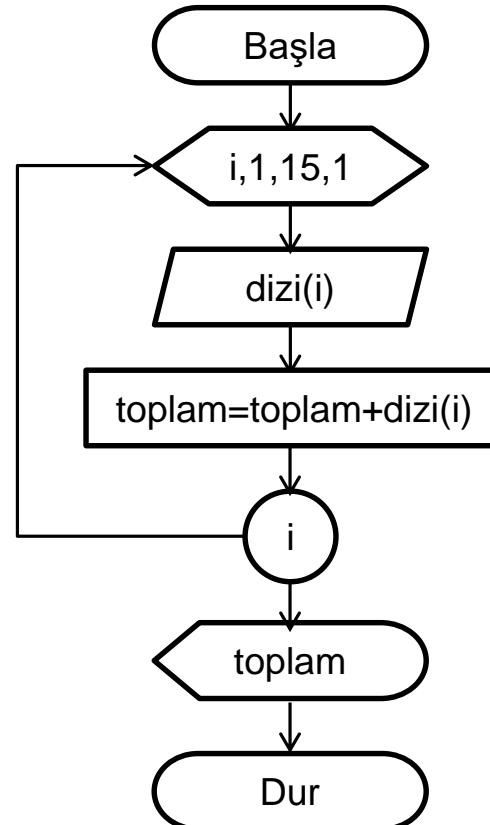


Döngü İfadesiyle



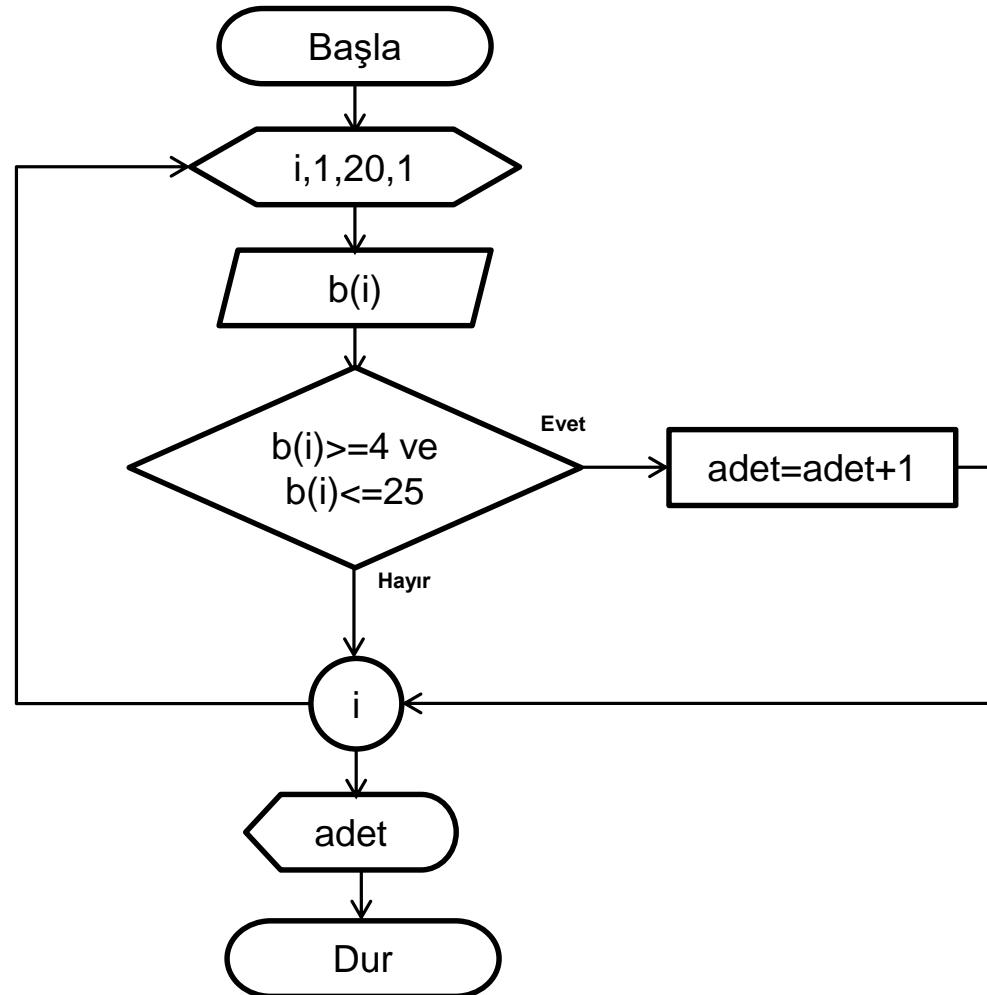
Örnek 16

(15 elemanlı bir sayı dizisine bilgi girişi yapılarak elemanlarının toplamını bulan akış şeması)



Örnek 17

(20 elemanlı bir sayı dizisine bilgi girişi yapılarak 4 ile 25 arasında olanların adedini bulan ve görüntüleyen akış şeması)



Kaba Kod (Pseudo Code)

(Sözde Kod)

- Kaba-kod, algoritmanın yarı programlama dili kuralları, yarı konuşma dili metinleri ile ortaya koyulması/tanımlanmasıdır.
- Gerçek kod ise, algoritmanın herhangi bir programlama diliyle gerçekleştirilmiş halidir.
- Kaba-kod ile verilen bir problem yazılımcılar tarafından istenen bir Programlama dili ile kodlanabilir.
- Kaba-kod herhangi bir programlama dilinden bağımsız olup herhangi bir kurala da sahip değildir.

Kaba Kod Örneği

Kaba Kod Örneği

Algoritma EnKucukBul;

/* Bu algoritma N elemanlı bir A dizinin en küçük değerli elemanını bulur. */

Adım

- ①. [Dizi boş mu?] /* dizide eleman olup olmadığı sinanır */
if N<1 then Begin
 Yaz ('Dizide eleman yok');
 GeriDön;
- ②. [İlk İşlemler]
EKE←A[1]; /* Başlangıçta dizinin ilk elemanı en küçük kabul edilir.*/
i←2; /* İkinciden başlanacağı için dizi indis değeri 2 olmaktadır. */
- ③. [Dizinin elemanlarını EKE ile karşılaştır]
i≤N olduğu sürece 5. adım dahil tekrarla
- ④. [Dizinin bir sonraki elemanı ile EKE ile karşılaştır]
if A[i]<EKE then EKE←A[i]; /* Dizinin i. elemanı daha küçükse onu
 EKE kabul et. */
- ⑤. [indis değerini bir artır.]
i←i+1;
- ⑥. [İşlemi Bitir]
GeriDön;

Algoritma 3.2. Dizinin en küçük değerli elemanını bulan algoritma

İç içe karşılaştırmalı ifadeler -1 (nested statements)

İç içe if (C-örneği)

```
#include <stdio.h>

int main () {

    /* local variable definition */
    int a = 100;
    int b = 200;

    /* check the boolean condition */
    if( a == 100 ) {
        /* if condition is true then check the following */
        if( b == 200 ) {
            /* if condition is true then print the following */
            printf("Value of a is 100 and b is 200\n");
        }
    }
    return 0;
}
```

İç içe karşılaştırmalı ifadeler -2 (case statements)

İç içe karşılaştırma – case (C-örneği)

```
#include <stdio.h>
int main () {

    /* local variable definition */
    char grade = 'B';

    switch(grade) {
        case 'A' :
            printf("Excellent!\n" );
            break;
        case 'B' :
            printf("Well done\n" );
            break;
        case 'C' :
            printf("You passed\n" );
            break;
        case 'F' :
            printf("You failed.\n" );
            break;
        default :
            printf("Invalid grade\n" );
    }
    return 0;
}
```

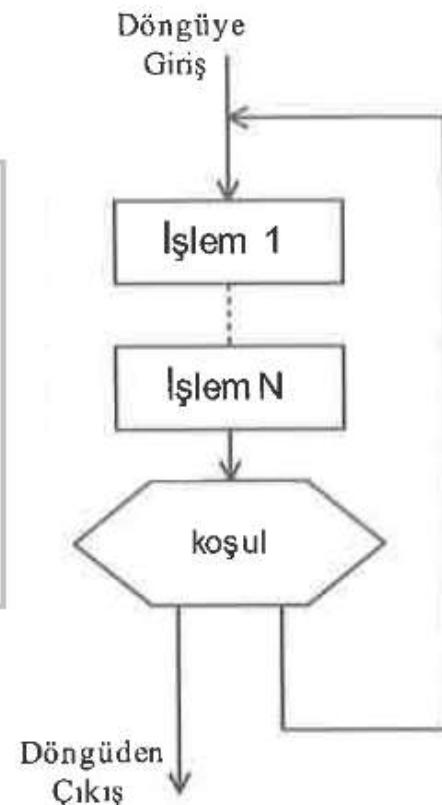
Döngü Çeşitleri

1. Repeat-Until (Do-while) Döngüleri

Repeat döngüsünde koşul ne olursa olsun ifadelerden oluşan işlem bloğu en az bir sefer çalışır. Eğer koşul sağlanamıyorsa sonsuz döngüye girilebilir. Bu nedenle bu döngü kullanılırken dikkat edilmelidir.

Repeat
İfadeler₁
İfadeler₂
...
İfadeler_n
Until (Koşul)

```
do {  
    İşlem 1;  
    ...  
    İşlem N;  
} while(koşul);
```



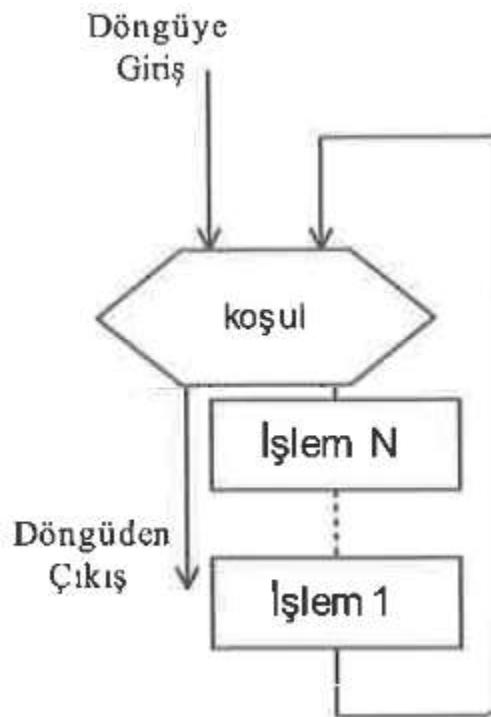
Döngü Çeşitleri

2. While Döngüleri

- While döngüsü de tekrarlı ifadeleri üzere geliştirmiş bir başka döngüdür. Bu döngü yapısında koşul, döngünün ilk satırında yer alır. Koşul doğru olduğu sürece döngünün bloğunda yer alan komutlar işlenir.

While *Koşul* Do
 İfade(i)
 Repeat;

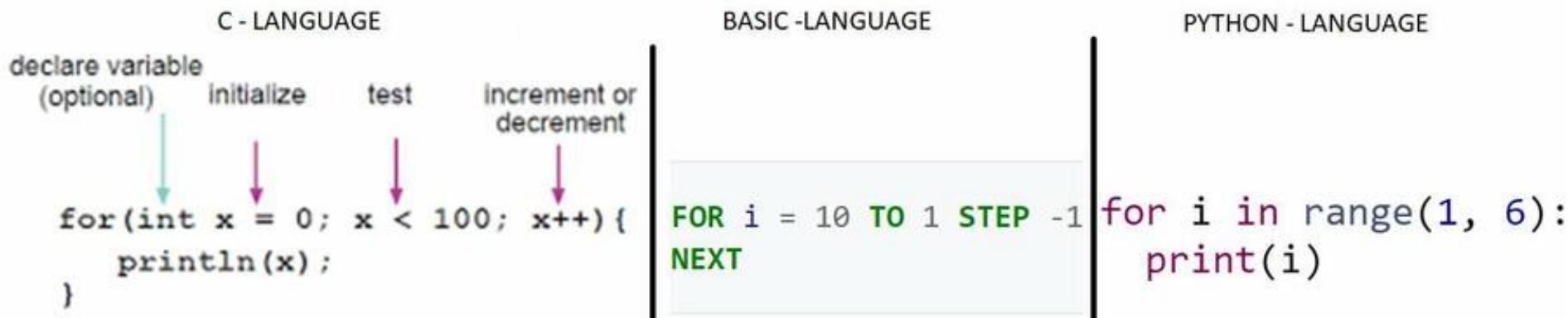
```
while (koşul) {  
    İşlem 1;  
    ...  
    İşlem N;  
}
```



Döngü Çeşitleri

3. for Döngüleri

- Kaç defa döneceği önceden bilinen durumlarda kullanılan bir döngü türüdür.
- Bu döngüde dönme adedini tutan bir döngü sayacı (örnekte x) değişkeni kullanılır. Bu değişken başlangıç olarak verilen değerden başlar, eşit değerli artımlarla veya azalışlarla test koşulu TRUE olduğu sürece devam eder. Bu esnada döngü bloğunda bulunan tüm ifadeler çalıştırılır.



Döngülerde yanlış kullanım

Akış Diyagramları dışında tasarım yöntemleri

- Metinsel tanımlama
- N-S(Nassi-Schnederman)şemaları
- W-O (Warnier-orr) Diyagramları
- Nesneye Yönelik Yaklaşım
(Gelecek dönem derste işlenecek)

Algoritma ve Akış Şeması

Alıştırma Soruları

1. 1 ile 10 arasındaki sayıların karelerinin toplamını ekranda gösterin
2. Girilen pozitif tamsayının kaç basamaklı olduğunu bulup ekranda gösterin
3. Girilen 20 tamsayıdan çift sayıların toplamının tek sayıların toplamına oranını ekranda gösterin
4. Girilen sayının 5'in kuvveti olup olmadığını ekranda gösterin

Programlama Dilleri

Programlama Dili Nedir?

- Programlama Dili, istenilen hesaplamaları yapmak için, elde edilen veriyi saklamak için ve girdi/çıktı aygıtlarına veri gönderme/alma gibi işlemleri yapmak için kullanılan dildir.
- Doğal dillerde olduğu gibi programlama dillerinde de belirli bir yazım kuralı (sentaks) vardır.
- Programlama dilleri ile sadece bilgisayarlar üzerinde çalışan uygulamalar değil, işlemcisi ve belleği bulunan diğer elektronik cihazlarda çalışan uygulamalar da yazılır.

Yazılım Geliştirme Süreci

1. Gereksinim Analizi
2. Yazılım Tasarımı
3. Kodlama (Programlama dili ile)
4. Sertifikasyon (Belgelendirme)
5. Bakım

Programlama Dillerinin Tarihçesi

1800'ler

- ***Ada Augusto Lovelace*** (1815-1852) Bernoulli sayılarının Charles Babbage'ın Analitik Makinesi (1822)(Analytical Engine) ile hesaplanması sağlayan bir yöntem geliştirmiştir.
- Bu hesaplama yöntemi, birçok tarihçi tarafından dünyadaki ilk bilgisayar programı olarak değerlendirilse de, buhar enerjisi ile çalışan analitik makine için geliştirilen bu yöntem, aslında bir programlama dili ile geliştirilen bir program değildir.



Programlama Dillerinin Tarihçesi

1940'lar

- Konrad Zuse 1943-1945 yılları arasında Plankalkül adında bir programlama dili geliştirmiştir. Fakat o yıllarda bu dil için bir derleyici tasarılmamıştır.
- 1945 yılında John von Neumann iki temel ilke ortaya atmıştır:
 - Basit donanım, karmaşık komutlar (programa göre donanım değişikliği gerekmeli)
 - Koşullu denetim yapıları (if-then gibi)
- 1949 Short Code programlama dili

```
a = (b+c)/b*c
```

was converted to Short Code by a sequence of substitutions and a final regrouping:

```
X3 = ( X1 + Y1 ) / X1 * Y1    substitute variables  
X3 03 09 X1 07 Y1 02 04 X1 Y1  substitute operators and parentheses.
```

Programlama Dillerinin Tarihçesi

1950'ler

- 1951 yılında UNIVAC I projesinde çalışan Grace Hopper ilk derleyiciyi tasarlamıştır.
- FORTRAN (**FOR**mula **TRAN**slator) John W. Backus liderliğindeki bir grup tarafından 1954-1955 yıllarında IBM 704 bilgisayarı için tasarlanmıştır (ilk derleyicisi 1957'de).
- LISP (**LIS**t **P**rocessor) John McCarthy tarafından 1956-1958 yılları arasında geliştirilmiştir (ilk derleyicisi 1959'da).
- ALGOL (**ALG**orithmic **L**anguage), FORTRAN'daki bazı eksik noktaları gidermek için 1958'de geliştirilmiş, 1960 ve 1968'de iki farklı uyarlaması yapılmıştır.
- COBOL (**C**Ommun **B**usiness **O**riented **L**anguage) Grace Hopper tarafından 1959 yılında geliştirilmiştir (ilk derleyicisi 1961'de).

Programlama Dillerinin Tarihçesi

1960'lar

- 1964 yılında John George Kemeny ve Thomas Eugene Kurtz tarafından BASIC (Beginner's All-purpose Symbolic Instruction Code) dili geliştirilmiştir.
- PASCAL dili Niklaus Wirth tarafından 1968-1969 yılları arasında geliştirilmiş ve 1970'te yayınlanmıştır.

Programlama Dillerinin Tarihçesi

1970'ler

- C programlama dili 1972 yılında Bell Laboratuarlarında Dennis Ritchie tarafından geliştirilmiştir (Dennis Ritchie daha önce Ken Thompson ile birlikte UNIX işletim sistemi ve B programlama dili üzerinde de çalışmıştır).
- 1979 yılında Bjarne Stroustrup tarafından geliştirilmeye başlanan C++ dili 1983 yılında tamamlanmıştır.

Programlama Dillerinin Tarihçesi

1980'ler

- 1986 yılında Bertrand Meyer tarafından geliştirilen Eiffel programlama dili C++ gibi nesneye-yönelik bir dildir.
- 1987 yılında Larry Wall tarafından geliştirilen Perl dili, metin işleme ve görüntü tanıma söz konusu olduğunda kullanılabilecek en güçlü dillerden biridir.

Programlama Dillerinin Tarihçesi

1990’lar

- Windows tabanlı uygulama geliştirmeyi sağlayan Visual Basic 1.0, Microsoft tarafından 1991 yılında piyasaya sürüldü. (1998 yılına kadar 5 faklı sürüm daha geliştirildi).
- Java dili, Sun Microsystems mühendislerinden James Gosling tarafından geliştirildi ve 1995 yılında ilk sürümü (1.0) yayınlandı.
- Dinamik web sayfalarının oluşturulmasında kullanılan PHP (Personal Home Page) dili Rasmus Lerdorf tarafından 1995’té yaratıldı.

Programlama Dillerinin Tarihçesi

2000'ler

- 2000'li yıllarda Web Tabanlı Programlama giderek yaygınlaşmış, ASP, JSP ve PHP dilleri sürekli olarak yenilenmiştir.
- .NET çatısı altında çalışan C# (C Sharp), VB.NET ve ASP.NET programlama dilleri 2001 yılında Microsoft tarafından piyasaya sürülmüştür.
 - .NET çatısı (.NET Framework), Microsoft tarafından geliştirilen, açık Internet protokoller ve standartları üzerine kurulmuş komple bir "uygulama" geliştirme platformudur.

Programlama Dillerinin Sınıflandırılması

1. Genel Sınıflandırma

- Temel (Imperative) Programlama Dilleri
 - Fortran, C, Cobol, Basic, Pascal
- Veriye Yönelik Programlama Dilleri
 - Lisp, Apl, Snobol, Icon
- Nesneye Yönelik (Object Oriented) Programlama Dilleri
 - Simula, C++, Ada95, Java, Visual Basic (VB)

Programlama Dillerinin Sınıflandırılması

2. Uygulama Alanlarına Göre Sınıflandırma

- Bilimsel ve Mühendislik Dilleri
 - Fortran, C, Pascal
- Sistem Programlama Dilleri
 - C, Assembler
- Veri Tabanı Dilleri
 - Dbase, Clipper
- Yapay Zeka Dilleri
 - Prolog, LISP
- Genel Amaçlı Programlama Dilleri
 - C, Pascal, Basic

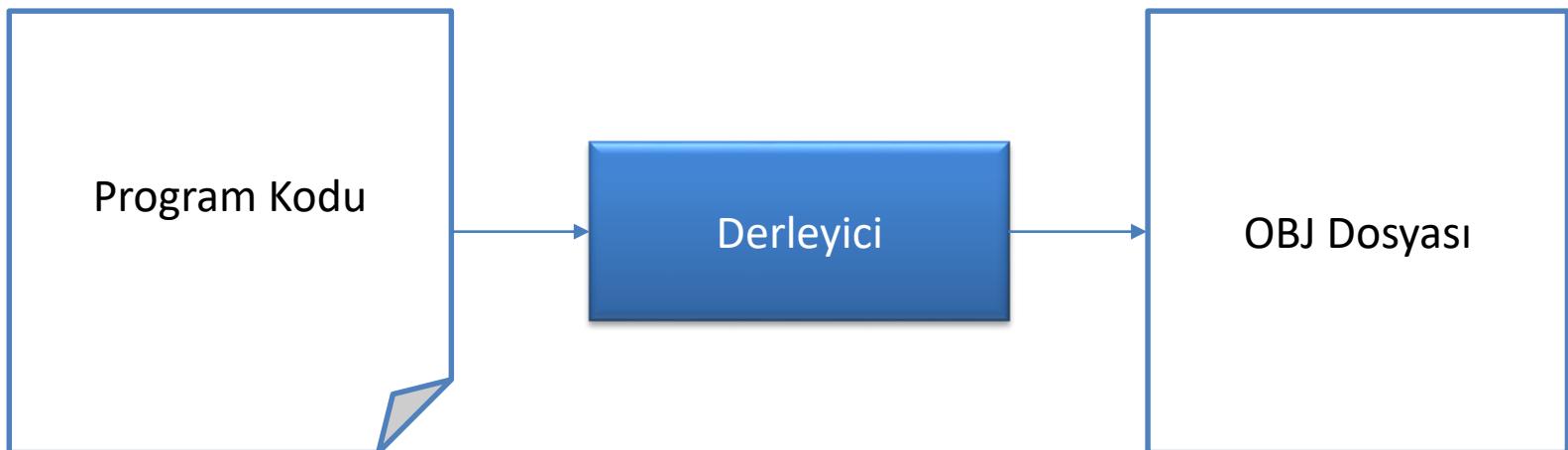
Programlama Dillerinin Seviyeleri

- Bir programlama dili konuştuğumuz doğal dile ne kadar yakın ise o kadar **yüksek seviyeli** dil, makine diline ne kadar yakın ise o kadar **düşük seviyeli** dil olarak sınıflandırılır.
- Düşük seviyeli: Makine Dili, Assembly
 - Bilgisayar donanımına direkt erişim, tam hakimiyet
- Orta seviyeli: C, C++
 - Belleğe tam erişim, kısa ve anlaşılır program kodu
- Yüksek seviyeli: Fortran, Visual Basic, Pascal, Cobol, C#, Python
 - Veritabanına kolay erişim, hazır modüller sayesinde kolay programlama
- Çok Yüksek seviyeli: Dbase, Clipper, Paradox, MS Access

Programlama Ortamı

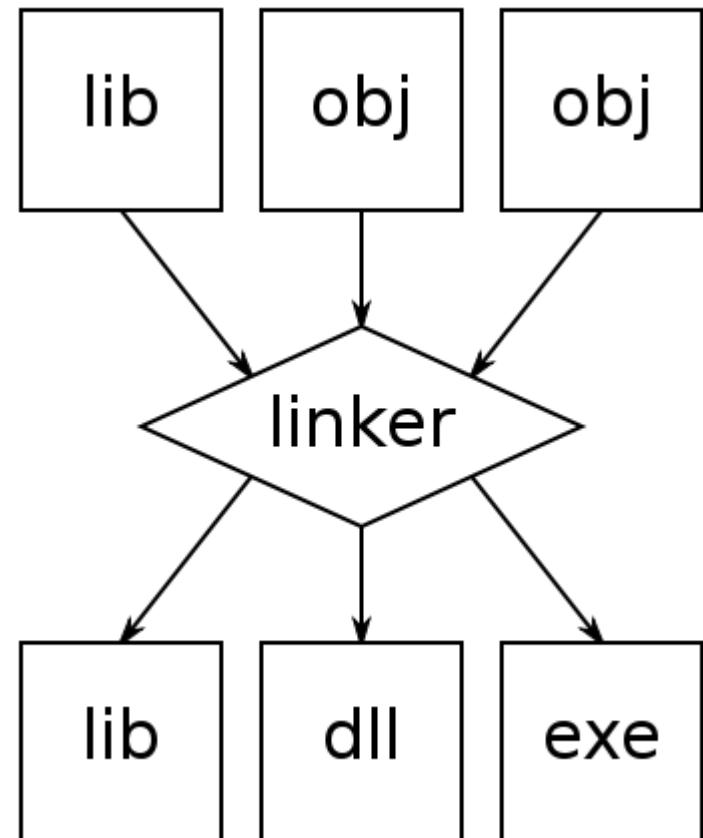
Derleyici (Compiler) Nedir?

- Bir programlama dili ile yazılmış program kodunu (kaynak kod: source code) bir ara koda çeviren yazılımdır. Bu ara kod genellikle OBJ dosyası (object file) olarak üretilir.



Bağlayıcı (Linker) Nedir?

- Derleyici tarafından üretilen ara kodları bağlayarak işletim sisteminin çalıştırabileceği makine kodlarına (exe: executable files, dll: dynamic link library) çeviren yazılımdır.



Yorumlayıcı (Interpreter) Nedir?

- Bir programın kaynak kodunu derlemek yerine doğrudan yürütten yazılımdır.
- Yorumlayıcı aracılığıyla çalışan dillerden bazıları:
 - Python
 - Matlab
 - R
 - SQL

Tümleşik Geliştirme Ortamı

- İngilizcesi: Integrated Development Environment (IDE)
- Kaynak kodu düzenleyicisi (source code editor), hata ayıklayıcı (debugger), derleyici (compiler) ve/veya yorumlayıcı (interpreter), bağlayıcı (linker) gibi uygulama geliştirme bileşenlerini tümleşik olarak sunan yazılımdır.
- Modern kaynak kodu düzenleyicileri, programı yazdığınız sırada arkaplanda çalışan; sözdizimi renklendirmesi, hata bulma ve otomatik tamamlama mekanizmalarına sahiptir.
- Microsoft tarafından geliştirilen Visual Studio ve açık kaynak kodlu Eclipse ve Netbeans yazılımları, faklı diller ile uygulama geliştirme imkanı sağlayan günümüzdeki en gelişmiş IDE'lerdir.
- DOS tabanlı eski IDE'ler: QBASIC, Turbo Pascal, Turbo C
- Windows tabanlı basit IDE'ler: Code-Blocks ve Dev-C++

Programlama Ortamının Temel Unsurları

- Editör (Editor):
 - Kaynak kodu oluşturmak ve gerekiğinde değişiklik yapmak için kullanılan araçtır. Editörde yazanlar seçilen dilin komutlarından oluşan metinlerdir.
- Derleyici (Compiler):
 - Editör tarafından bir bilgisayar dilinde yazılmış olan kaynak kodu makine koduna çeviren bir bilgisayar yazılımıdır. Yazılan kodun kullanılan dile uygunluğunu denetler.

Programlama Ortamının Temel Unsurları

- Kütüphane (Library):
 - Nesne dosyalarından oluşan kütüphanedir.
- Bağlayıcı (Linker):
 - Programın içerdiği tüm nesne dosyalarını birleştirerek tek bir yürütülebilir dosya haline getirir.
- Yükleyici (Loader):
 - Yürütülebilir dosyayı diskten belleğe kopyalar.

Programlama Ortamının Temel Unsurları

- Hata Ayıklayıcı (Debugger):
 - Programcının hatalarını anlayabilmesi için programın yürütülmesini adım adım kontrol edebilmesini sağlar.
- Yorumlayıcı (Interpreter):
 - Bir programın kaynak kodunu doğrudan satır satır yürüten bir programdır.

Programlama Dillerinin Elemanları

- **Söz Dizimi (Syntax)**
 - Sıradan dillerde olduğu gibi, programlama dillerinin de bir söz dizimi vardır.
 - Bir programın söz dizimi simgelerin geçerli olarak kabul edilebilmesi için hangi düzende yazılması gerektiğini belirleyen kurallar dizisidir.
 - Sözdizimsel hataların çoğu derleyici tarafından yakalanıp raporlanacaktır. Ancak bazı hatalar derleyicilerin yakalayamayacağı türden olup çalışma esnasında hatalara neden olabilir.

Programlama Dillerinin Elemanları

- Anlambilim (Semantics)
 - Bir programlama dilindeki bir ifadenin ne anlama geldiğidir.
- Veri (Data)
 - Verinin yapısı ve türü belirlenmelidir.
 - Tür program içinde kullanılacak verinin alabileceği değerleri belirler.

Programlama Dillerinin Elemanları

- Atama Deyimi (Assignment Statement)
 - Atama deyimi; Deyimin sağ tarafındaki ifadenin değerini hesaplayarak, bulduğu değeri sol taraftaki ifadeye aktarır.
- Kontrol Deyimleri (Control Statement)
 - Atama deyimleri normal olarak yazıldığı şekilde yürütülürken, kontrol deyimleri yürütmenin sırasını değiştirmek için kullanılır.
 - İyi yapılandırılmış bir kontrol deyimi iki şekilde olabilir.
 - İki yada daha fazla seçenekten birinin seçilmesi (if yada case)
 - Tekrarlı ifadelerin çevrim durumları (for –while-repeat)

Programlama Dillerinin Elemanları

- Alt Programlar (Subprograms)
 - Programın farklı yerlerinden defalarca çağrılabilen, içinde veri tanımlamalarının ve yürütülebilir deyimlerin bulunduğu program birimleridir.

Programlama Dillerinin Özellikleri

- **Giriş / Çıkış Komutları:** Bu komutlar, bilgisayara verileri almalarını ve sonuçlarını sergilemelerini belirtir. Verilen veri, çıktıların yönlendirildiği cihazlar bu komutlarda belirtilir.
- **Hesaplama Komutları:** Bu komutlar, temel matematik işlemlerini yapmaya yarayan komutlardır.

Programlama Dillerinin Özellikleri

- **Kontrolün Yönlendirilmesi İçin Kullanılan Komutlar:** Bu komutlar, programın normal komut akışından sapabilmek için kullanılır.
- **Verilerin Bilgisayar İçinde Taşınması, Saklanması ve Geri Çağrılmasına Yönerek Komutlar:** Bu komutlar, verileri çeşitli bellek adresleri arasında olduğu kadar diskle bellek arasında da hareket ettirmek amacıyla kullanılır.

Programlama Dillerinin Özellikleri

- Programlama dillerinin en önemli özelliklerinden birisi makineden bağımsız olabilmeleridir.
- Programların taşınabilirliği olarak da adlandırılan bu özellik programın geliştirildiği bilgisayardan başka bir bilgisayarda da çalışabilmesi anlamına gelir.

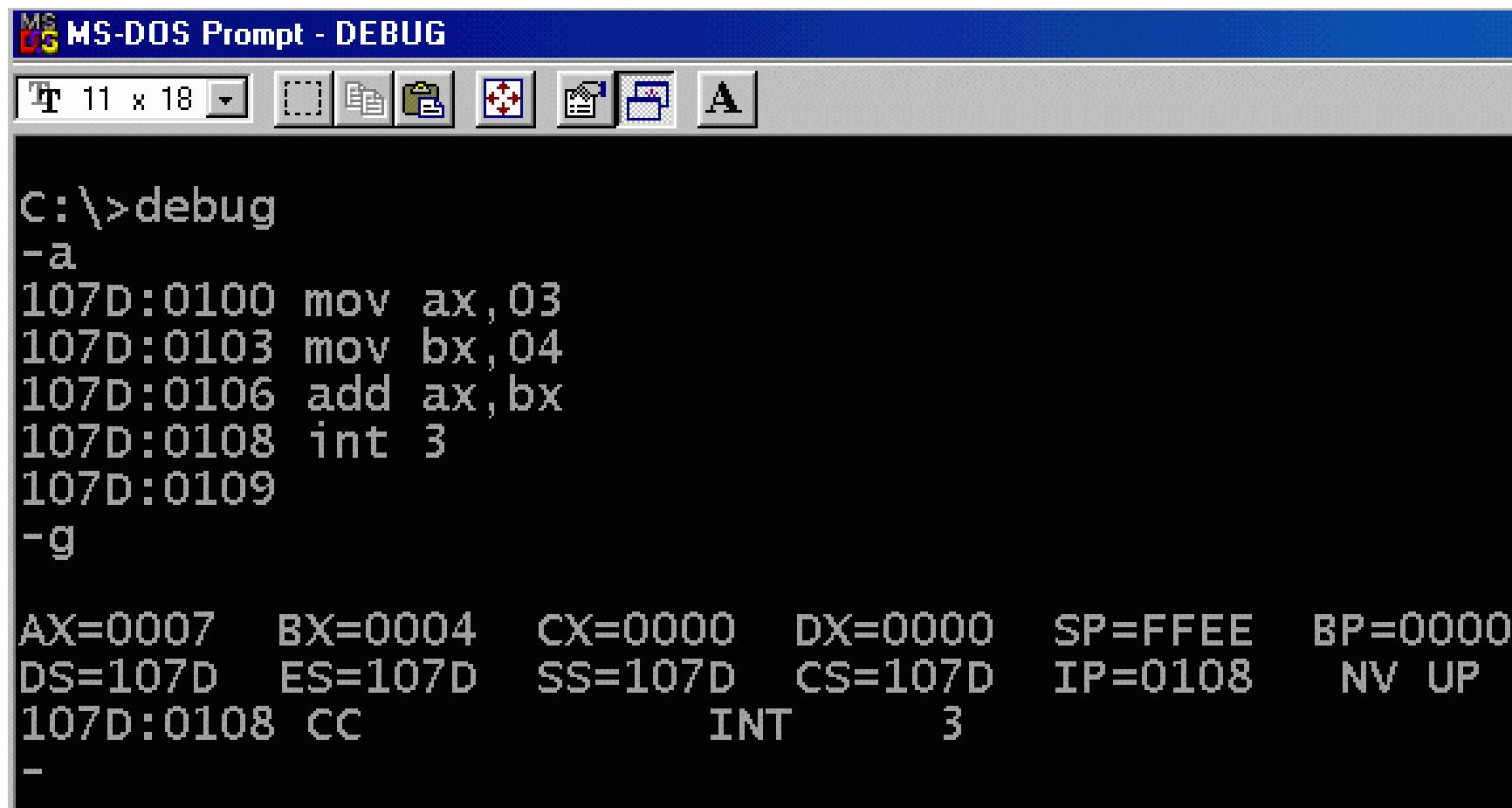
Makine Dili

- Makine dili, işlemcinin verilen komutlar doğrultusunda çalıştırılmasını sağlayan ve işlemci mimarisine göre değişen en alt seviyedeki programlama dilidir.
- Bu dil sadece 0 ve 1 binary ikililerinin anlamlı kombinasyonlarından meydana geldiği için, bu dili kullanarak program yazmak çok güçtür.
- Diğer programlama dillerin gerektirdiği derleyici ya da yorumlayıcı kullanımını gerektirmemesi ve donanımı doğrudan kontrol etme gücü pozitif yönleridir.

Assembly

- Assembly, makina dilinin sayısal ifadelerini, insanlar tarafından anlaşılır olarak programlanması daha kolay olan alfabetik ifadelerle değiştirerek düşük seviyede programlama için uygun bir ortam oluşturur.
- Bu dilde yazılan programlar orta ve yüksek seviyeli programlama dillerine göre çok daha hızlı çalışır. Ancak anlaşılmasıının güçlüğü ve kodlamanın çok uzun olması nedeniyle kullanımı zor bir dildir.
 - Örneğin C dili ile yazılan 8-10 satırlık kodun makinaya verdiği mesajı bu dilde ancak 80-90 satır kod ile verilebilir.

Assembly



The image shows a screenshot of the MS-DOS Prompt - DEBUG window. The title bar reads "MS-DOS Prompt - DEBUG". The menu bar includes "File", "Edit", "Format", "Run", "Help", and "Exit". The toolbar contains icons for "File", "Edit", "Format", "Run", and "Help". The command line area shows the following sequence:

```
C:\>debug  
-a  
107D:0100 mov ax,03  
107D:0103 mov bx,04  
107D:0106 add ax,bx  
107D:0108 int 3  
107D:0109  
-g  
  
AX=0007 BX=0004 CX=0000 DX=0000 SP=FFEE BP=0000  
DS=107D ES=107D SS=107D CS=107D IP=0108 NV UP  
107D:0108 CC INT 3  
-
```

FORTRAN

- FORTRAN, IBM tarafından IBM 704 bilgisayarı üzerinde fen ve mühendislik uygulamalarını programlamak amacıyla 1950'li yıllarda geliştirilmeye başlanmış, 1960'larda farklı uyarlamaları geliştirilmeye devam etmiştir (Fortran I: 1957, Fortran II: 1958, Fortran IV: 1962, Fortran 66: 1966, ...)
- Sadece IF, GOTO, DO ifadelerini içeren basit bir dil olduğu için Pascal ve C gibi programlama dilleri geliştirildikten sonra kullanımı azalmıştır.

FORTRAN'da Merhaba Dünya

- Bir dilin sözdizimine örnek olarak yaygın biçimde bir "Merhaba Dünya" programı gösterilir. Aşağıda Fortran ile yazılmış bir "Merhaba Dünya" programı verilmiştir:

```
PROGRAM HELLO  
PRINT*, 'Hello World!'  
END
```

COBOL

- Sayısal işlemlerin gerçekleştirilmesinde avantaj sağlayan FORTRAN dili giriş/çıkış (I/O) işlerinde yeterli değildir.
- Giriş/çıkış işlemlerinin oldukça önemli olduğu ticari uygulamalar için 1959 yılında COBOL dili geliştirilmiştir.
- Sayısal ve metin türü verilerin diziler ve kayıtlar gibi veri grupları haline organize edilmesini sağlamıştır.

BASIC

- BASIC (Beginner's All-purpose Symbolic Instruction Code) öğrenmesi ve yazması kolay olan bir dildir.
- Bu nedenle genellikle bilgisayar bilgisi az olanlara programlama dili öğretmek için kullanılır.
- Microsoft şirketi tarafından geliştirilen QBASIC (QuickBasic), MS-DOS işletim sisteminin belli sürümlerinin parçası olarak sunulmuştur

Örnek bir QBASIC programı

'Belirtilen İki Sayıyı Toplar

CLS

INPUT "Birinci Sayıyı Giriniz:";A%

INPUT "İkinci Sayıyı Giriniz:";B%

PRINT A% + B%

END

Yapısal Programlama Nedir?

- 1960'lı yılların sonunda ortaya çıkan yapısal programlama (structured programming) kavramı ile uzun ve karmaşık programların, bloklara ayırarak daha kolay biçimde yazılabilmesi mümkün olmaktadır.
- Yapısal programlamada programın belirli bir satırına gitmeye yarayan GOTO ifadesinin kullanılmaması, **yordamsal** (prosedürel) bir yapının tercih edilmesi esastır.

PASCAL

- PASCAL programlama dili pek çok öğrenciye bilgisayar programlamayı öreten ve çeşitli versiyonları bugün hala yaygın olarak kullanılmaya devam eden en önemli programlama dillerinden biridir.
- Bilgisayar bilimcisi Niklaus Wirth 1970'te yapısal programlamayı derleyiciler için daha kolay işlenir hale getirebilmek amacıyla PASCAL'ı geliştirmiştir. Adını matematikçi ve düşünür Blaise Pascal'dan almıştır.
- Wirth, PASCAL'dan başka Modula-2 ve Oberon programlama dillerini de geliştirmiştir. Bu diller Pascal'a benzerler ve ayrıca nesneye yönelik programmayı da desteklerler.

PASCAL

The screenshot shows the Turbo Pascal IDE interface. The menu bar includes File, Edit, Run, Compile, Options, Debug, and Break/watch. The title bar displays "Edit" and "Unindent * D:NONAME.PAS". The status bar at the bottom shows keyboard shortcuts: F1-Help, F5-Zoom, F6-Switch, F7-Trace, F8-Step, F9-Make, and F10-Menu.

```
Line 15    Col 39  Insert Indent          Unindent * D:NONAME.PAS
program KenLovesTurboPascal;
uses
  crt;
var
  age: Integer;
  name: String;
  message: String;
begin
  ClrScr;           ←
  name := 'Ken Egozi';
  age := 30;
  if age < 10 then
    message := ' loves Turbo Pascal'
  else
    message := ' loved Turbo Pascal';
  write (name);
  writeln (message);
end.
```

C

- AT&T Bell laboratuvarlarında, Ken Thompson ve Dennis M. Ritchie tarafından UNIX İşletim Sistemi'ni geliştirebilmek amacıyla B dilinden türetilmiş yapısal bir programlama dilidir.
- Geliştirilme tarihi 1972 olmasına rağmen yaygınlaşması Brian W. Kernighan ve Dennis M. Ritchie tarafından 1978'de yayımlanan "C Programlama Dili" kitabından sonra olmuştur.
- İşletim sistemleri, gömülü sistemler, sürücü yazılımı ve hız gereken her türlü işlemde kullanılan bir dildir.
- 70'li ve 80'li yıllarda birçok farklı uyarlaması geliştirildikten sonra, 1989'da ANSI tarafından standart bir C dili tanımlanmıştır (ANSI C).

Pascal & C (1)

- Pascal ve C dilleri yaklaşık aynı zamanlarda geliştirilmiştir ve aralarında önemli benzerlikler vardır.
- Orijinal Pascal ile C'nin ikisi de yapısal programlama fikrini gerçekleştiren küçük ve prosedürel dillerdir.
- İkisinde de dinamik bellek ayırma ve işaretçi (pointer) işleme mümkündür.
- Ancak, bu iki dil dışarıdan bakıldığında farklı görünüler (C programları genelde Pascal programlarından kısadır).

PASCAL ve C'de Merhaba Dünya

```
Program HelloWorld;  
uses crt;  
begin  
    writeln('Merhaba Dünya');  
    readln;  
end.
```

```
#include <stdio.h>  
main() {  
    printf("Merhaba Dünya");  
    getchar();  
}
```

Nesneye Yönerek Programlama

- NYP'nin altında yatan ana fikir; her bilgisayar programının, etkileşim içerisinde olan birimler veya nesneler kümesindenoluştuğu varsayımdır.
- Bu nesnelerin her biri, kendi içerisinde veri işleyebilir, ve diğer nesneler ile çift yönlü veri alışverişinde bulunabilir.
- NYP'dan önce var olan yaklaşımda (yordamsal programlama), programlar sadece bir komut dizisi veya birer işlev (fonksiyon) kümesi olarak görülmektediler.

Nesneye Yönelik Programlama

- NYP 3 temel yapı üzerine oturur:
 1. Veri soyutlama (data abstraction)
 2. Katılım (Inheritance)
 3. Çok biçimlilik (Polymorphism)

C++

- Simula dili ile ortaya çıkan nesneye yönelik yaklaşım Bjarne Stroustrup tarafından C diline uyarlanmış ve ortaya C++ çıkmıştır (1983).
- Günümüzde en çok kullanılan dillerden biri olan C++, C dilinin hız ve esneklik gibi tüm iyi özelliklerini korumuştur.
- 1998 yılında ANSI/ISO tarafından Standard C++ yayınlanmıştır.

JAVA

- Java, Sun Microsystems mühendislerinden James Gosling tarafından geliştirilmeye başlanmış açık kodlu, nesneye yönelik, ortamdan bağımsız, yüksek verimli, çok işlevli, yüksek seviye, adım adım işletilen (interpreted) bir dildir.
- Interaktif TV'ler ve küçük cihazlarda kullanılmak üzere tasarlanan JAVA, platform bağımsızlığı sayesinde geniş kullanım alanı buldu.
- JAVA'nın senktaksi C ve C++ dillerine benzemekle birlikte daha basit bir nesne modeli ve daha az alt seviye olanakları içerir (işaretçi kullanımı yoktur).

Java ile Merhaba Dünya

```
public class HelloWorld {  
    public static void main(String[] args) {  
        // Prints "Hello, World" to the terminal window.  
        System.out.println("Hello, World");  
    }  
}
```

C#

- C++ ve JAVA'nın iyi yönlerini bünyesinde barındıran yeni bir dildir (2001).
- Bu dilin tasarlanmasına Pascal, Delphi derleyicileri ve J++ programlama dilinin tasarımlarıyla bilinen Anders Hejlsberg liderlik etmiştir.
- Microsoft tarafından .NET çatısı üzerinde geliştirilen C#, JAVA'ya en önemli rakip olarak görülmektedir.
- C# görsel programlamaya da olak sağlar.

C# ile Merhaba Dünya

```
class MerhabaDunya
{
    // Programın ilk girdiği nokta
    static void Main(string[] args)
    {
        System.Console.WriteLine("Merhaba Dünya!");
    }
}
```

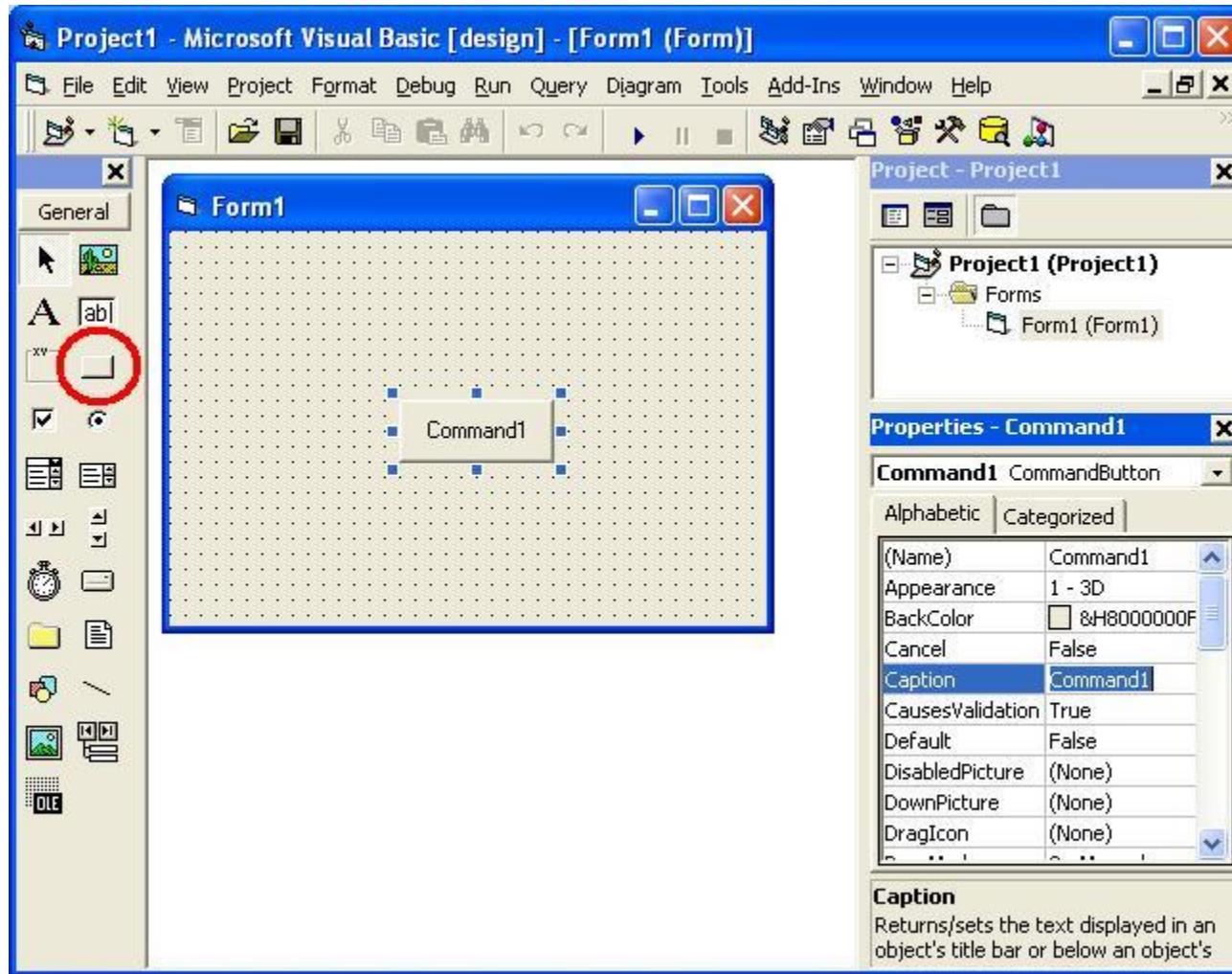
Görsel Programlama Dilleri

- Grafik Kullanıcı Arabirimi (GUI: Graphical User Interface) kullanan Windows, Linux gibi işletim sistemleri için uygulama geliştirmeye yarayan yeni nesil programlama dilleridir.
- Pencere içinde yer alan metin kutuları, butonlar, tablolar gibi elemanları program kodu yazarak değil, görsel olarak pencereye yerleştirmeye olanak sağlar.
- Görsel dillerin çoğu, sık kullanılan bazı yapıların kodunu programa kendisi ekleyerek programcıya kolaylık sağlar.

Visual Basic (VB)

- BASIC dilinden türetilen Windows tabanlı program yazmayı sağlayan bir dildir.
- BASIC'ten farklı olarak yapısal (yordamsal) bir programlama dildir.
- Kullanıcı sayısı az olan veya kısa sürede bitmesi gereken küçük ölçekli projelerde tercih edilir.
- VB.NET'ten önceki son sürümü 1998'de çıkan Visual Studio 6.0 paketinde yer almıştır.

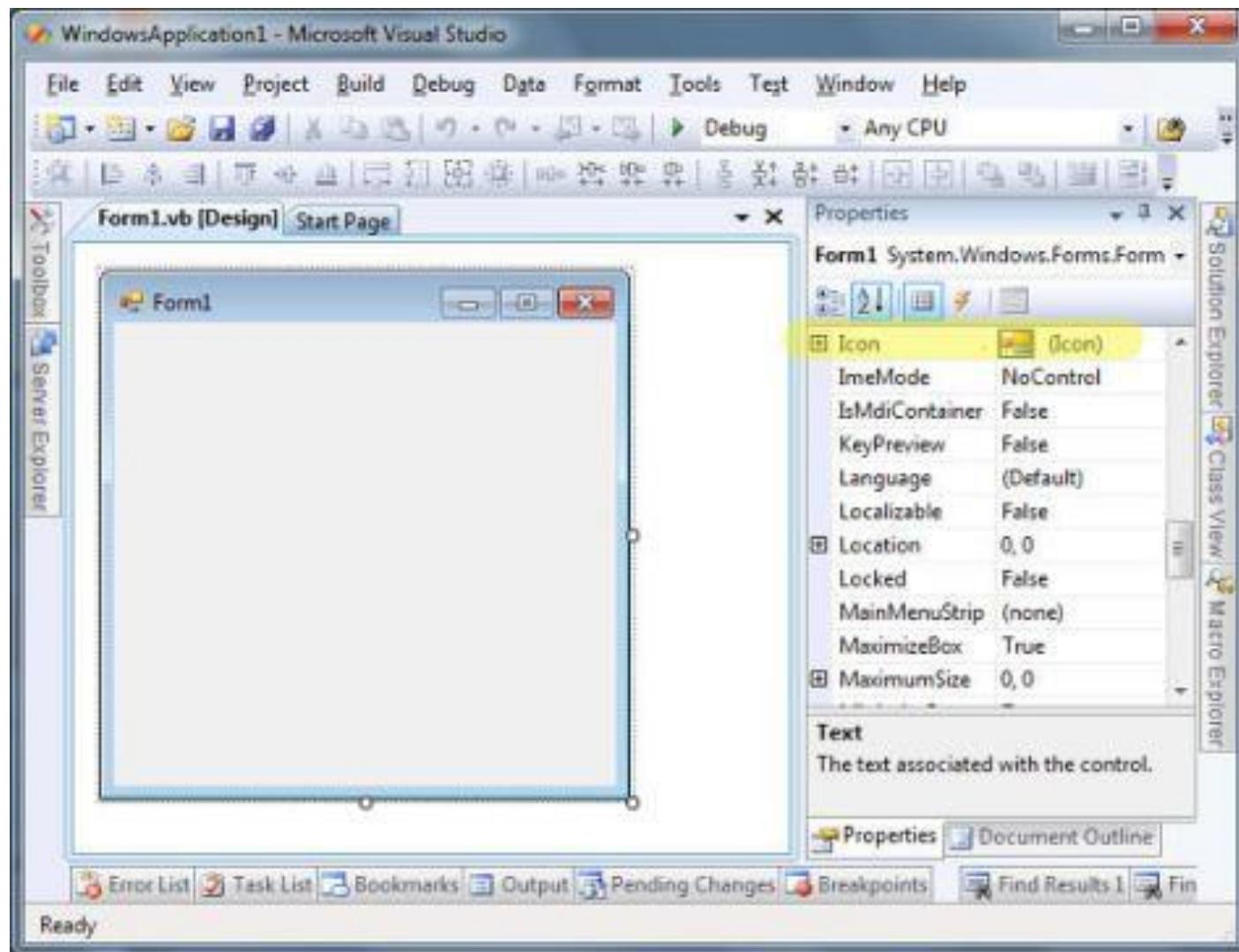
Visual Basic (VB)



VB.NET

- Microsoft 2001 yılında .NET adını verdiği uygulama geliştirme ortamını tanıtmış ve bu yapıyı (.NET Framework 1.0) kullanan VB.NET, C# gibi programlama dillerini Visual Studio .NET uygulama geliştirme paketiyle birlikte piyasaya sürmüştür.
- VB.NET, VB'den farklı olarak nesneye yönelik bir dildir.
 - Çok biçimlilik (polymorphism), kalıtım (inheritance), veri soyutlama (data abstraction) ve sarmalama (encapsulation) gibi tüm NYP özelliklerini içerir.

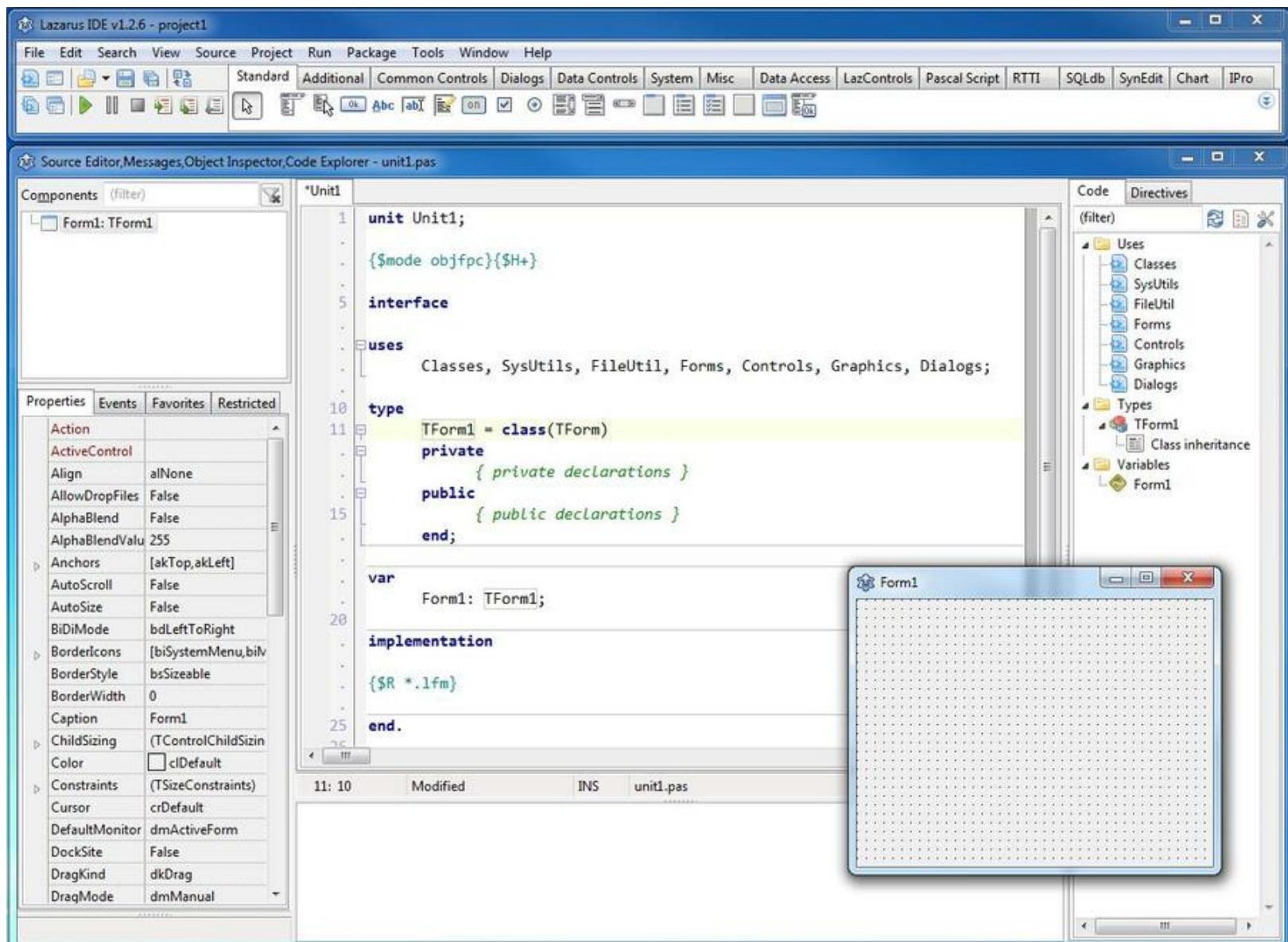
VB.NET



Delphi

- PASCAL dilinden türetilen görsel programlama dilidir.
- Borland firması tarafından geliştirilen Delphi'nin ilk sürümü 1995 yılında piyasaya sürüldü (VB'de olduğu gibi neredeyse her sene yeni bir sürüm yayınladı).
- 2003 yılından sonra Borland Firmasının kapanmasıyla popülerliği azalmıştır.
- Günümüzde LAZARUS adıyla açık kaynak kodlu bir DELPHI sürümü geliştirilmektedir.

Delphi (Lazarus)



Görsel Programlama Dilleri

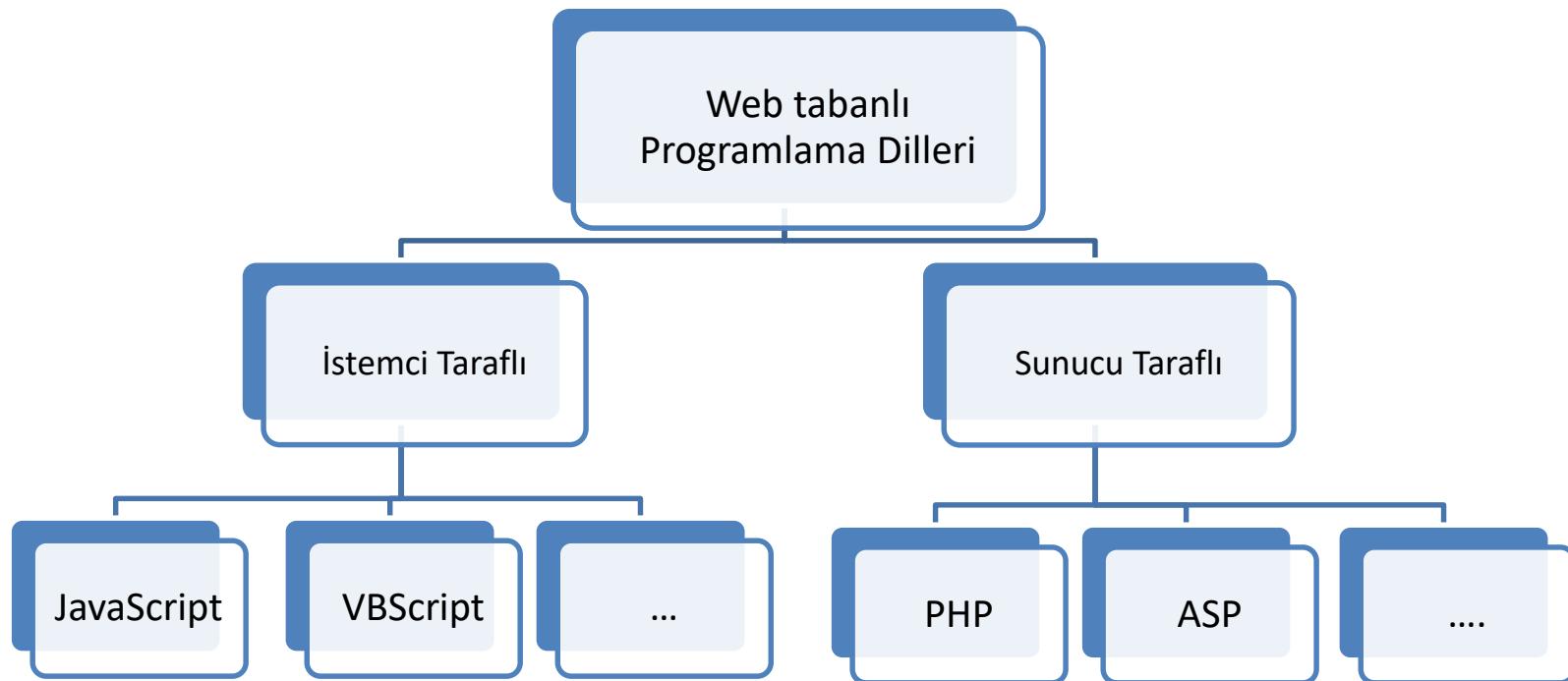
- C, C++, Java, C# dilleri ile hem komut satırı hem görsel programlar geliştirilebilir.
 - C için GTK+
 - C++ için QT Framework
 - C# için Windows Forms
 - Java için SWING

adlı görsel bileşen kütüphaneleri mevcuttur.

Web Tabanlı Programlama

- Sunucunun sadece durağan (statik) sayfaları istemciye (ziyaretçi) göndermesi yerine, ziyaretçiden veri kabul edilmesi (dinamik yaklaşım) gerekliliğinin sonucunda ortaya çıkan programlama yaklaşımıdır.
- Web tabanlı uygulamaların büyük ölçüde platform bağımsız olmaları ve kurulum gerektirmeyen yapıda olmaları (her yerden erişilebilir ve bakımı kolay) günümüzde bu tür uygulamalara olan ilgiyi arttırmıştır.

Web Tabanlı Programlama



Web Tabanlı Programlama

- İstemci Taraflı Programlama dilleri, HTML dilinin karşılayamadığı bazı ihtiyaçlara çözüm üretmek için kullanılır.
- İstemci Taraflı Programlama dilleri, kullanıcı ile veri alış verisi içerisindeidir.
- Her işlem istemci üzerinde gerçekleştirilir.
- =Script dilleri

Web Tabanlı Programlama

- Script dilleri sayfa ile kullanıcının etkileşimli olarak çalışmasını sağlar:
 - bir nesneye tıklamak,
 - bir nesnenin üzerine gelmek,
 - bir nesnenin üzerinde dolaşmak gibi

Web Tabanlı Programlama

- Sunucu Taraflı Programlama dilleri, sunucu ile veri alış verisi içerisindeidir.
- Her işlem sunucu üzerinde gerçekleştirilir.
- Örneğin bir dosya yüklersin, sunucudaki dosyayı düzenlersin.

Web Tabanlı Programlama

- ❖ Webde istemci –sunucu iletişimi
 - ❖ İstemci bilgisayarda internet explorer veya firefox gibi bir web tarayıcısına bir adres girilir,
 - ❖ Tarayıcı ilgili web sunucusunu bulur ve sayfayı ister,
 - ❖ Web sunucusu ilgili sayfayı komutlar topluluğu şeklinde istemci makineye gönderir,
 - ❖ İstemci makinedeki tarayıcı bu sayfaları alır, gelen komutları yorumlar ve web sayfasını anlaşılır bir şekilde gösterir

PHP (Personal Home Page)

- PHP ilk kez Rasmus Lerdorf tarafından, web sayfalarını ziyaret edenleri izlemek amacıyla bir dizi Perl Script (betik) kullanılarak geliştirilmiştir.
- Kısa süre sonra başka insanların ilgisini çekmeye başlayınca, Rasmus bir script motoru oluşturdu ve web formlarına da destek verdi ve böylece PHP/F1'i biçimlendirmiş oldu.
- Adını duyurdukça bir gurup yazılımcının dikkatini çekti ve ortak bir çalışma sonucu PHP3 ortaya çıktı.
- Daha sonra Zend motoru kullanılarak PHP4 yaratıldı.

ASP (Active Server Pages)

- ASP ilk olarak 1996 yılının sonunda Microsoft'un web sunucu uygulaması olan IIS (Internet Information Services) için bir eklenti olarak ortaya çıktı.
- ASP.NET 2002 yılında .NET Framework 1.0 ile birlikte piyasaya sürüldü.
- .NET çatısı sayesinde ASP.NET kodu CLR (Common Language Runtime) kullanan herhangi bir dil ile yazılmaktadır.

JSP (Java Server Pages)

- JSP 1.0 sürümü 1999 yılında Java'nın ASP ve PHP'ye cevabı olarak Sun Microsystems tarafından piyasaya sürüldü.
- 1.2 sürümünden sonra JSP **Java Community Process** tarafından geliştirilmeye başlandı.
- Mayıs 2006'da JSP 2.1 sürümü Java EE 5'in bir parçası olarak piyasaya sürüldü.

SON

İşletim Sistemleri

İşletim Sistemi

- Bilgisayar donanımının doğrudan denetimi ve yönetiminden, temel sistem işlemlerinden ve uygulama yazılımlarını çalışıtmaktan sorumlu olan sistem yazılımıdır.
- Bütün diğer yazılımların belleğe, girdi/çıkıtı aygıtlarına ve kütük sisteme erişimini sağlar.
- Birden çok yazılım aynı anda çalışıyorsa, her yazılıma yeterli sistem kaynağını ayırmaktan ve birbirleri ile çakışmamalarını sağlamaktan da sorumludur.

İşletim Sistemi

- İşletim sistemi temel fonksiyonları yerine getiren bir programlar kümesidir.
- İşletim sisteminin fonksiyonları:
 - Bilgisayarı başlatma
 - Programları yönetme
 - Hafızayı (bellek) ayarlama
 - Giriş ve çıkış birimleri arasındaki iletişimini sağlama
 - Kullanıcıyı bilgisayar kullanımı için imkanlı kılma

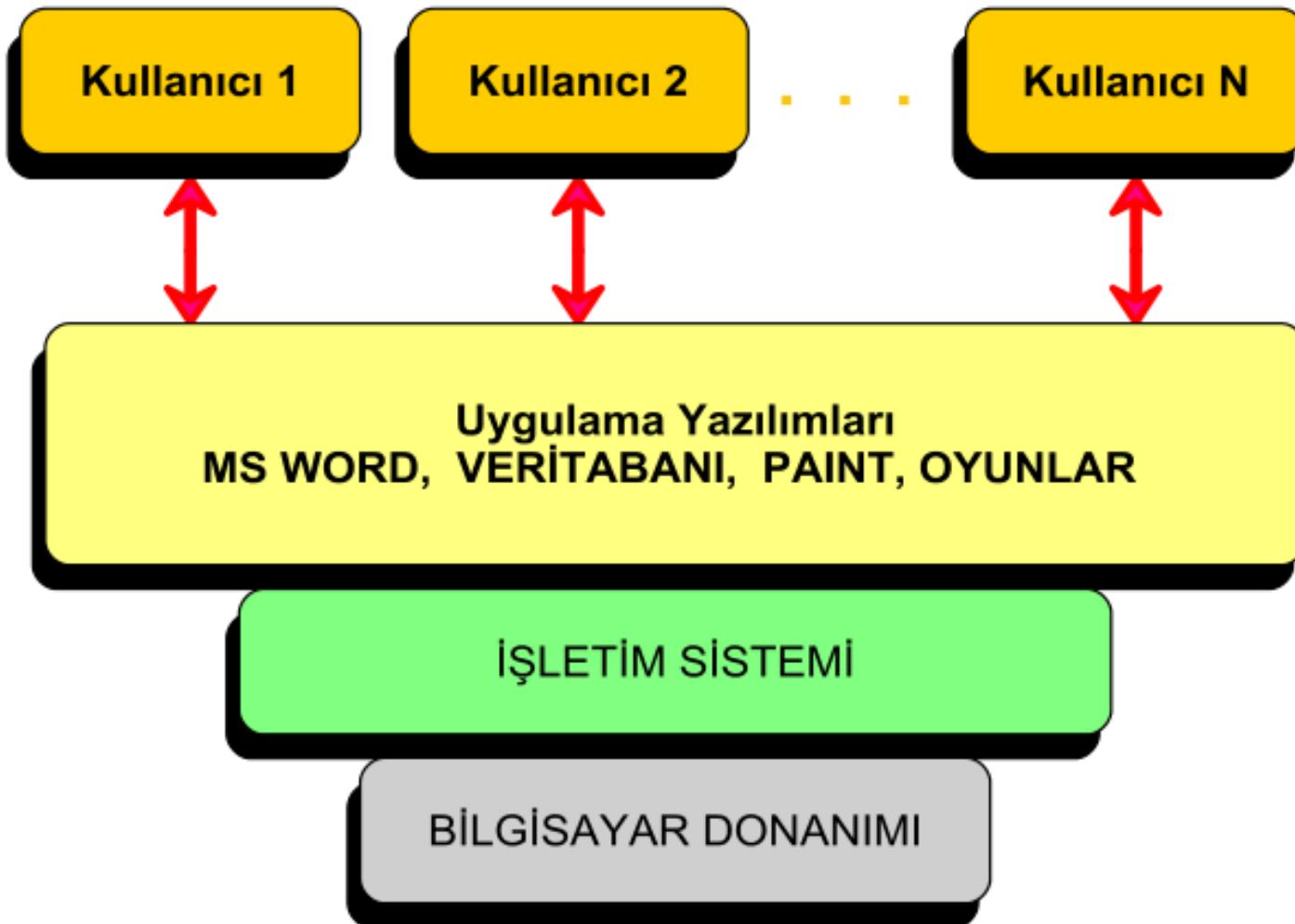
Bilgisayarın Açılışı

- Önyükleme (booting) – önyükleme işlemi işletim sisteminin bilgisayar belleğine kopyalanmasıyla gerçekleşir.
- Önyükleme iki şekilde incelenir:
 - soğuk Önyükleme – Bilgisayarın açılması esnasında işletim sistemin yüklenmesi.
 - sıcak Önyükleme – Bilgisayarın açık olduğu esnasında işletim sistemin tekrar yüklenmesi.

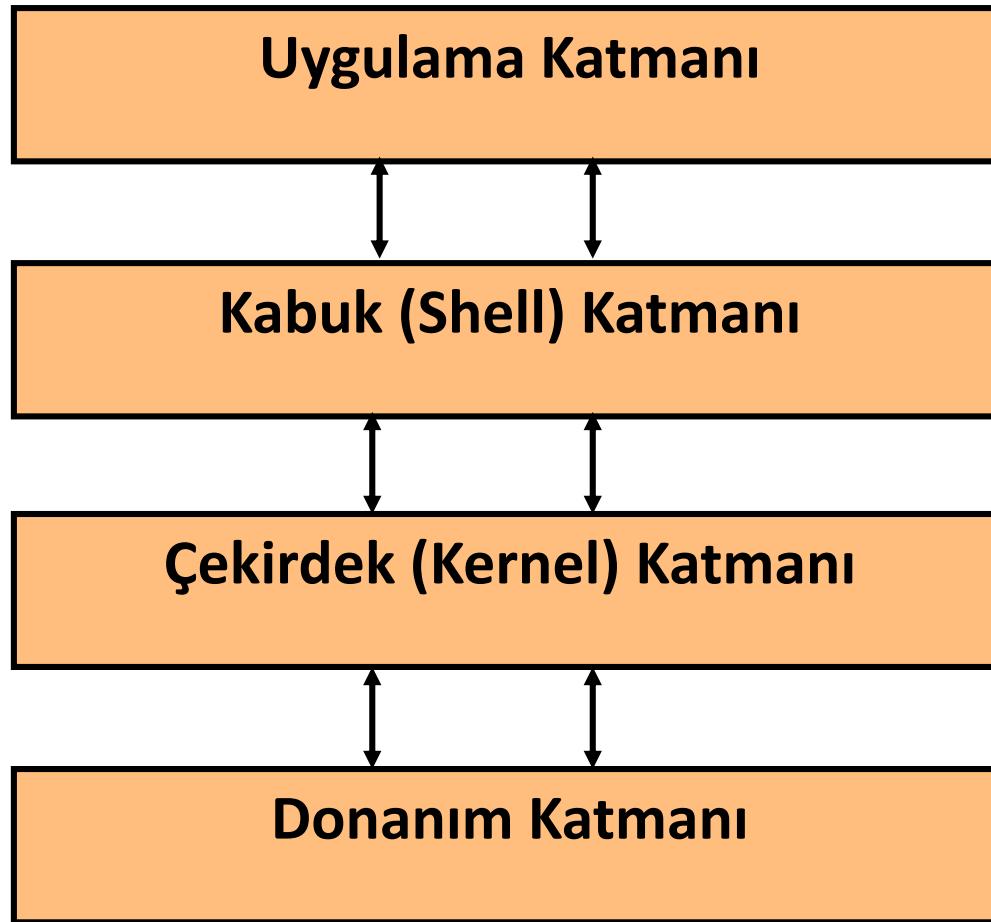
Bilgisayar Sisteminin 4 bileşeni

- **Donanım:** İşlemci,bellek,depolama, G/Ç birimleri
- **İşletim Sistemi:** Windows,Linux, Android, MacOS...
- **Uygulama Programları:** Web tarayıcı, ofis programları, oyunlar vs..
- **Kullanıcılar:** işletim sistemini kullanan kişiler

Bilgisayar Sisteminin 4 bileşeni



İşletim Sisteminin Katmanları



Çekirdek (Kernel)

- İşletim sisteminin ana bileşeni olan çekirdek, uygulama programları ile donanım bileşenleri arasında köprü vazifesi görür.
- İşlemcide çalışan proseslerin yönetimi, bellek yönetimi ve Giriş/Çıkış cihazlarının yönetimi gibi birçok temel işlev çekirdek tarafından gerçekleştirilir.

Çekirdek (Kernel)

- Çekirdek bilgilerin sabit sürücüden bilgisayar belleğine kopyalanmasını sağlar.
- Çekirdek (kernel):
 - İşletim sisteminin ana kısmını teşkil eder
 - Bütün uygulamaları başlatır
 - Birimleri ve belleği düzenler
 - Bellekte ayarlama yapar
 - Diğer gerekli fonksiyonları çalıştırır

Kabuk (Shell)

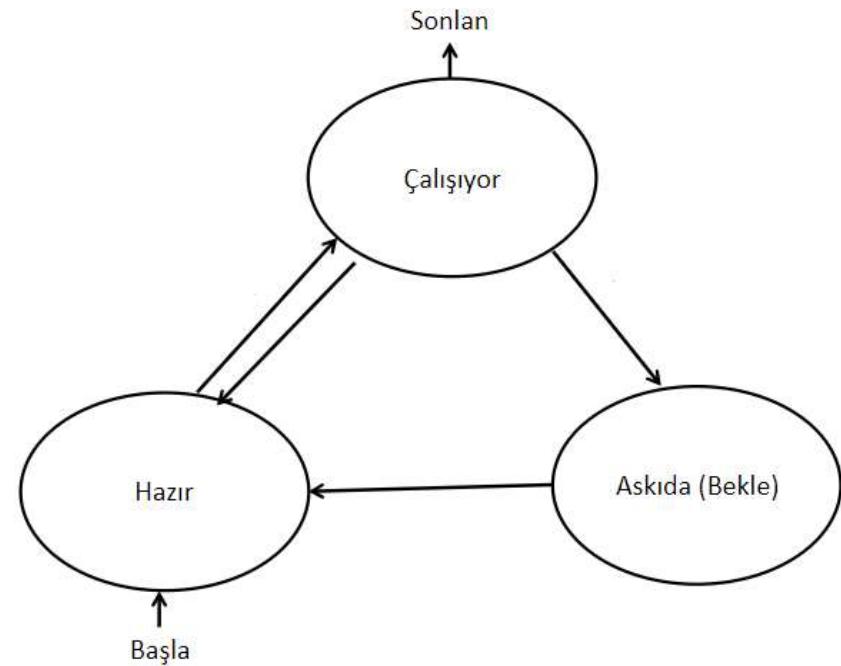
- Çekirdeğin sağladığı servislere erişim sağlayan bir ara katmandır.
- İki farklı kabuk çeşidi vardır:
 - Komut istemi (CLI: command-line interface) kullanan kabuk işlemlerin daha hızlı yapılmasını sağlar. (DOS)
 - Grafiksel arayüz (GUI: graphical user interface) kullanan kabuk ise kullanıcının yapmak istediği işlemleri daha hızlı tarif edebilmesini sağlar. (Windows)

Proses (Process)

- Bir işletim sisteminde anahtar kavamlardan biri olan **Proses** temel olarak “çalıştırılmakta olan bir program”dır.
- Bir **kaynak program** durgun bir komutlar dizisi şeklinde bulunurken, proses bu komutlar dizisinin işletilmesi anındaki durumuna verilen isimdir.
- Çok görevlilik (Multitasking) özelliğine sahip olan işletim sistemleri, aynı anda birçok prosesin çalıştırılmasına izin verir. Tek çekirdekli işlemciye sahip olan bilgisayarlarda bu iş **zaman paylaşımı** esasına göre yapılır.

Proseslerin durumu

- İşletim sisteminin amacı, verimi en yüksek tutacak şekilde işlemciyi proseslere paylaştırmaktır.
- Bir proses şu 3 durumdan birinde bulunabilir.
 1. Çalışıyor(Running)
 2. Askıda (Blocked)
 3. Hazır (ready)



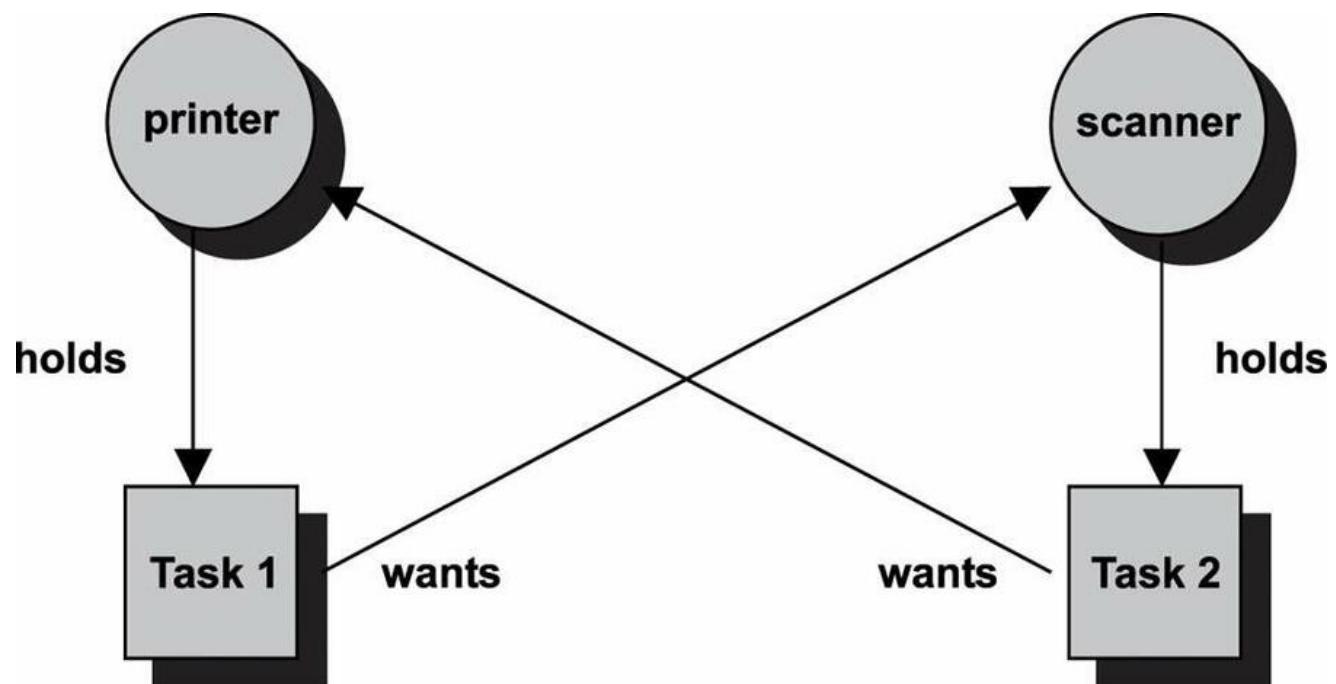
Zaman Çizelgeleyici (Schedular)

- Çizelgeleyici, işlemciye atanacak olan prosesi ve prosesin hangi koşullar altında işlemciyi kullanacağını belirler. İki ana görevi vardır:
 1. Proses kuyruğundan bir yürütülecek proses seçmek
 2. Prosese ait kullanım zamanı dilimini ayarlamak

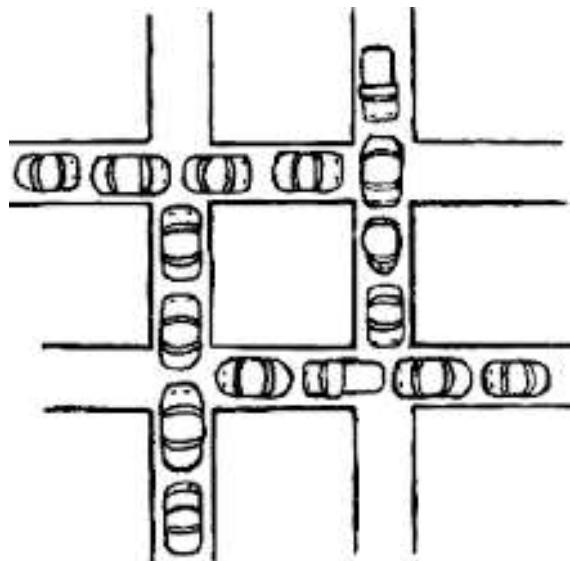
Ölümcul Kilitlenme (deadlock)

- Proseslerin hiçbir zaman ele geçiremeyecekleri bir birime yada kaynağa ihtiyaç duymaları halinde,
- İki proseten her birinin diğerinin işini bitirmesini beklemesi durumunda

Oluşabilir.



Ölümçül Kilitlenme (deadlock)



Toplu İşleme (Batch Processing)

- Kullanıcıların, bilgisayar sisteminde bağımsız bir bütün olarak ve belli bir sıra dahilinde işlenmesini istedikleri hizmetler kümesine toplu iş (batch job) denir.
- Bilgisayar sistemlerine gönderilen işler, bir veya birden fazla programın ayrı ayrı işletileceği alt adımlardan oluşabilir. İşler genellikle adımların art arda uygulanacağı biçimde düzenlenir. Her adım, bir öncekinin sonuçlanması üzerine işletime girer.
- Örneğin DOS işletim sisteminde **bat** uzantılı dosyalarda toplu iş adımları yer alır.

Dosya (File) & Dizin (Directory)

- Dosya, birbiriyle ilişkili veriler topluluğunu (bir bilgisayar programının kaynak kodu, programın derlenmiş olan çalıştırılabilir hali, metin-ses-görüntü verileri, vs.) bir saklama ünitesinde saklamak amacıyla kullanılan yapıdır.
- Dizin (Klasör) ise birbiriyle ilişkili dosyaların saklama birimlerinde hiyerarşik bir yapıda gruplanmasına olanak sağlayan yapıdır.
- Her işletim sistemi dosya/dizin yaratmak ve yok etmek, dosyadan veri okumak ve yazmak için komutlar (sistem çağrıları) içerir.

İşletim Sistemi Türleri

- Kontrol ettikleri bilgisayar türlerine ve destekledikleri uygulama türlerine göre işletim sistemleri genel olarak dört gruba ayrılır:
 - Gerçek zamanlı işletim sistemleri
 - Tek kullanıcı-tek görev işletim sistemi
 - Tek kullanıcı-çoklu görev işletim sistemi
 - Çoklu kullanıcı işletim sistemleri

Gerçek zamanlı işletim sistemleri

- Bu işletim sistemi türü, genel olarak endüstride ve büyük işletmelerde bilgisayarları ve bilgisayarlara bağlı sistemleri kontrol etmek amacıyla kullanılır.
- Bu tür bir işletim sisteminde kullanıcı arayüzü (user interface) kapasitesi oldukça sınırlıdır.
- İşlemlerin her defasında aynı biçimde ve standartta yapılmasını sağlar.
- Ürünlerin veya hizmetlerin aksamadan ve aynı kalitede üretilmesini sağlamak amacıyla, farklı ve karmaşık kaynaklar eşgüdümlü olarak yönetilerek bu işlemlerin devamlılığını sağlar.

Tek kullanıcı-tek görev işletim sistemleri

- Bu işletim türünü kullanan bilgisayar sistemi, tek bir kullanıcının her defasında tek bir işlemi gerçekleştirmesini olanaklı kılar.
- Diğer işletim sistemi türlerine göre daha az karmaşık işlemlerin yapılmasında kullanılan bilgisayarları kontrol eder.
- Avuç-içi bilgisayarlar bunun en güzel örneğidir.

Tek kullanıcı-çoklu görev işletim sistemleri

- En yaygın kullanılan işletim sistemi türüdür.
- Çoğumuzun kullandığı masaüstü veya dizüstü bilgisayarlarda bu işletim sistemi çalışır.
- Tek kullanıcı-tek görev türünden farkı, kullanıcının aynı anda birden fazla işlemi gerçekleştirebilmesidir.
- Bu işletim sistemi sayesinde bilgisayarlarda aynı anda bir çok işlemi yapabiliriz.
- Örneğin kelime işlemci programla bir rapor yazarken, aynı sırada İnternette bir veriyi tarayabilir veya Sunum programı ile bir sunuyu da hazırlıyor olabiliriz.

Çoklu kullanıcı işletim sistemleri

- Farklı bir çok kullanıcının, bilgisayar kaynaklarından aynı anda ve kesintisiz yararlanmasını sağlayan işletim sistemi türüdür.
- Bu sayede, farklı işlemleri yapan pek çok kullanıcı, aynı sistemin kaynaklarını, gereksinimlerine göre birbirlerinden etkilenmeden kullanır.
- Çoklu kullanıcı işletim sistemi, farklı kullanıcı taleplerini kontrol eder, sıraya koyar ve farklı işlemlerin dengeli bir şekilde yapılabilmesi için sistem kaynaklarını yönetir.

Sistem Kullanım Biçimlerine göre İşl. sistemleri

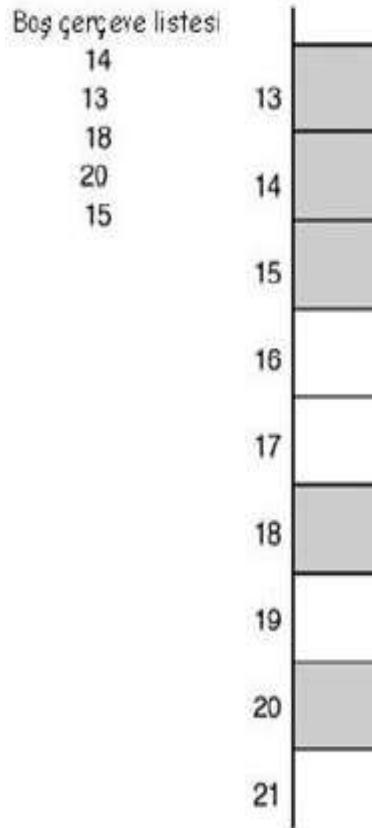
- Adanmış işlem (Dedicated processing)
- Toplu İşlem(Batch Processing)
- Etkileşimli işlem (interactive processing)

Bellek Yönetimi

- Bilgisayarın ana belleğini belirli bir düzen içinde yönetilmesine «bellek yönetimi» denir.
- Bellek yönetim biriminin:
 - İhtiyaç duyulan belleği vermek,
 - bununla ilgili bilgileri tutmak,
 - kullanılmayan bellek alanını serbest bırakmak, sanal bellek yönetimi
- gibi işlevleri vardır.

Bellek Yönetimi

- Bazı Bellek yönetim teknikleri
 - Tekli kesintisiz (single contiguous)
 - Bölümlemmiş (partitioned)
 - Sayfalı (paged)
 - Dilimli (segmented)



(a)

Yerleşimden önce

(b)

Yerleşimden sonra



Sanal Bellek

- Ana belleğin yetersiz kaldığı durumlarda, işletim sistemi sabit diskte yer açarak, bu alanı RAM gibi kullanır ve bu alana sanal bellek denir.
- Programların kullandığı iki tür bellek bulunur.
 - Ana bellek bilgisayarın fiziksel, yani gerçek belleğidir. Bu bellekte fiziksel adresler kullanılmaktadır.
 - Sanal bellek ise bilgisayarın sabit diskinde bulunan ve işletim sistemlerinin kullandığı özel bir bellek türüdür. Bu bellek türünde de mantıksal adresler kullanılmaktadır.

Sanal Bellek

- Sanal belleğin en büyük avantajı, programların boyutunun ana belleğin boyutundan bağımsız olmasıdır. Böylece, ana bellekten büyük programlar da çalıştırılabilir.
- Örneğin, ana belleği 32 MB olan bir bilgisayarda, 300 MB'lık bir program çalıştırılmak istenirse, bu programın tümünün RAM'e yerleşmesi mümkün değildir.
- Geri kalanı ise sanal bellekte bekler ve gerekli olduğunda RAM'e getirilir.

İşletim Sistemleri

- UNIX İşletim Sistemi
- DOS işletim sistemi
- Windows Ailesi
- Linux Ailesi
- Android
- MacOS
- iOS

UNIX

- 1971 yılında Bell laboratuarlarında çalışır hale getirilen ilk UNIX sürümü assemblerler ile yazılmıştı.
- 1973 yılında Ken Thompson, C derleyicisinin yaratıcısı Dennis Ritchie ile birlikte çekirdeği C ile tekrar kodladı.
- Böylece Unix, çeşitli hedef donanımlara uygun olarak tekrar derlenebilen kodlardan oluşan taşınabilir bir işletim sisteme dönüşmüştür.
- 1978 yılı Unix için çok önemli bir yıldı. Unix İşletim Sistemi 7. sürümüyle birlikte gelişimini artık iki farklı çizgide gerçekleştirecekti: BSD (Berkeley Software Distribution) ve System V.

Günümüzde UNIX

- BSD ve System V'in birçok komut ve özellikleri birbirine benzemektedir. Unix'in çoğu sürümünde her iki yaklaşımından da faydalanılmıştır.
- System V tabanlılar:
 - OpenSolaris
 - Oracle Solaris
- BSD Tabanlılar
 - FreeBSD
 - OpenBSD
 - NetBSD
 - PC-BSD



DOS (Disk Operating System)

- DOS ufak ve basit bir işletim sistemi olup, ana görevi disket ve sabit disk gibi saklama ortamlarının yönetimidir.
- Kişisel bilgisayar (PC) piyasasına giren IBM ürettiği Intel 8086/8088 işlemcili PC için bir işletim sistemi aramaya başlamıştır. O zaman hakim sistem CP/M olduğundan IBM de kullanmak istemiş fakat Digital Research ile anlaşamayınca o zaman küçük bir firma olan Microsoft'a yeni bir işletim sistemi siparişi vermiştir.
- Microsoft QDOS (Quick and Dirty OS) isimli sistemi alıp DOS 1.0 sürümü piyasaya çıkarmıştır.

DOS (Disk Operating System)

- IBM bir hata yaparak DOS'un satış iznini Microsoft firmasına vermiştir (bu hata dev bir şirketin doğusu olmuştur).
- DOS 3 sürümüne kadar iki DOS sürümü birbiri ile aynı iken, hızla büyüyen Microsoft'un, IBM OS/2 işletim sistemi çalışmasından ayrılop rakip olarak Windows 1.0 çalışmalarını başlatması şirketlerin arasını açmıştır.
- Bundan sonra da iki DOS sistemi altında bazı farklılıklar ortaya çıkmaya başlamıştır. MS-DOS son kullanıcıya yönelik özellikleri öne çıkarırken IBM-DOS ağ ve iş özelliklerini öne çıkarmıştır.



FreeDOS

- Microsoft 1981-2000 yılları arasında birçok farklı DOS sürümü geliştirmiştir. Son sürüm Windows ME içinde yer alan MS-DOS 8.0 olmuştur.
- Microsoft'un 2000'de MS-DOS'u geliştirmeyi durdurması üzerine DOS'u yaşatmak için açık kaynaklı bir işletim sistemi olan FreeDOS doğmuştur. (www.freedos.org)
- Amacı DOS ile %100 uyumlu olmaktır ve bu hedefe neredeyse tümüyle ulaşılmıştır.



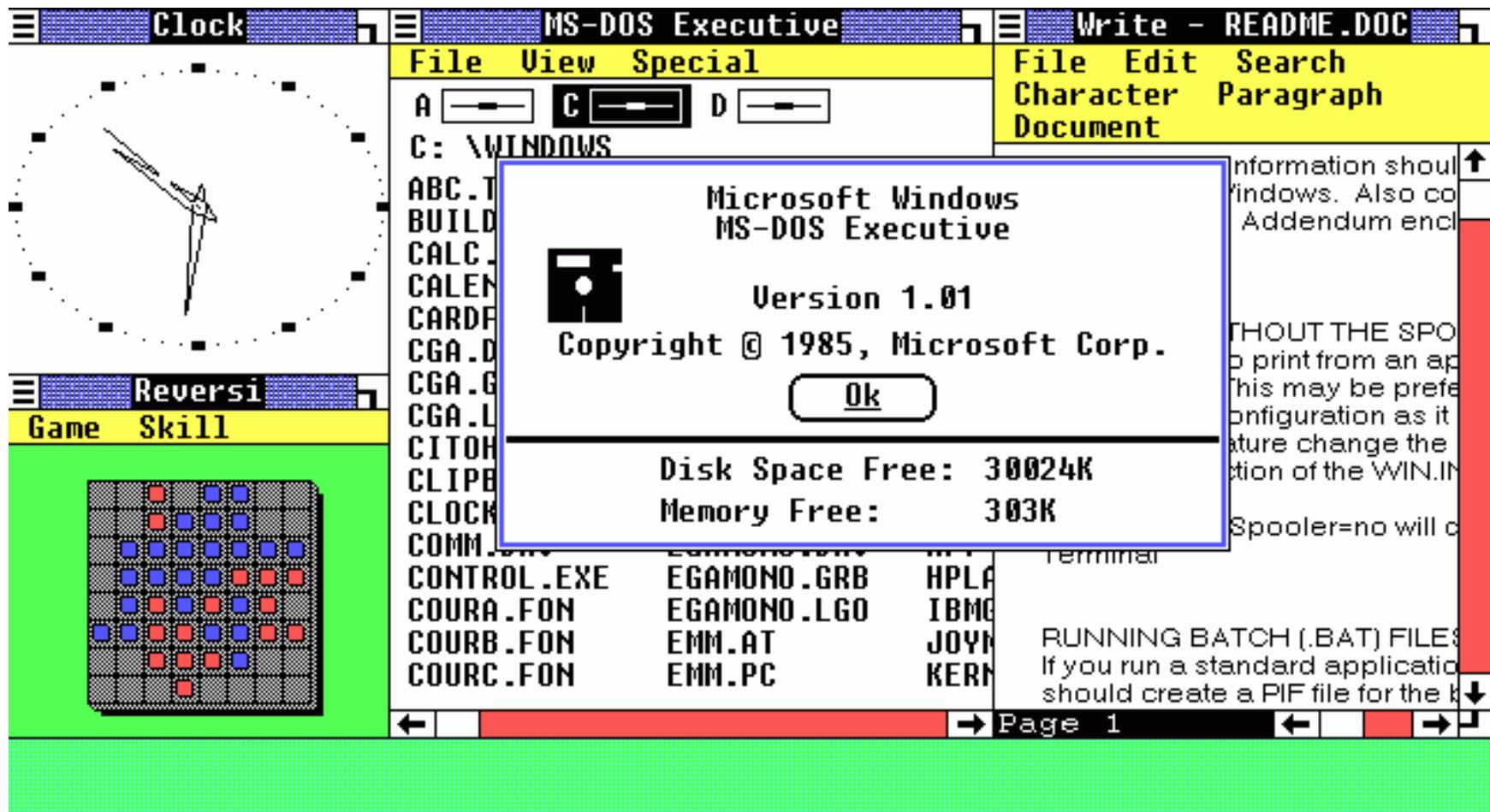


Microsoft Windows



- 1985 yılında piyasaya sürülen Windows 1.0 aslında DOS için bir grafiksel kullanıcı arayüzü idi.
- 1987'de 2.0 ve 1990'da 3.0 sürümleri satışa sunulan Windows'un 1992'de satışa sunulan 3.1 sürümü ile kullanım oranı iyice arttı.
- 1995 yılında 32-bit mimariye sahip (Fakat yine DOS tabanlı çalışan) Windows 95 piyasaya çıktı.
- Ev kullanıcılarını hedefleyen 1998'de satışa sunulan Windows 98 ve 2000 yılında satışa sunulan Windows ME yine DOS tabanlı idi.

Windows 1.0



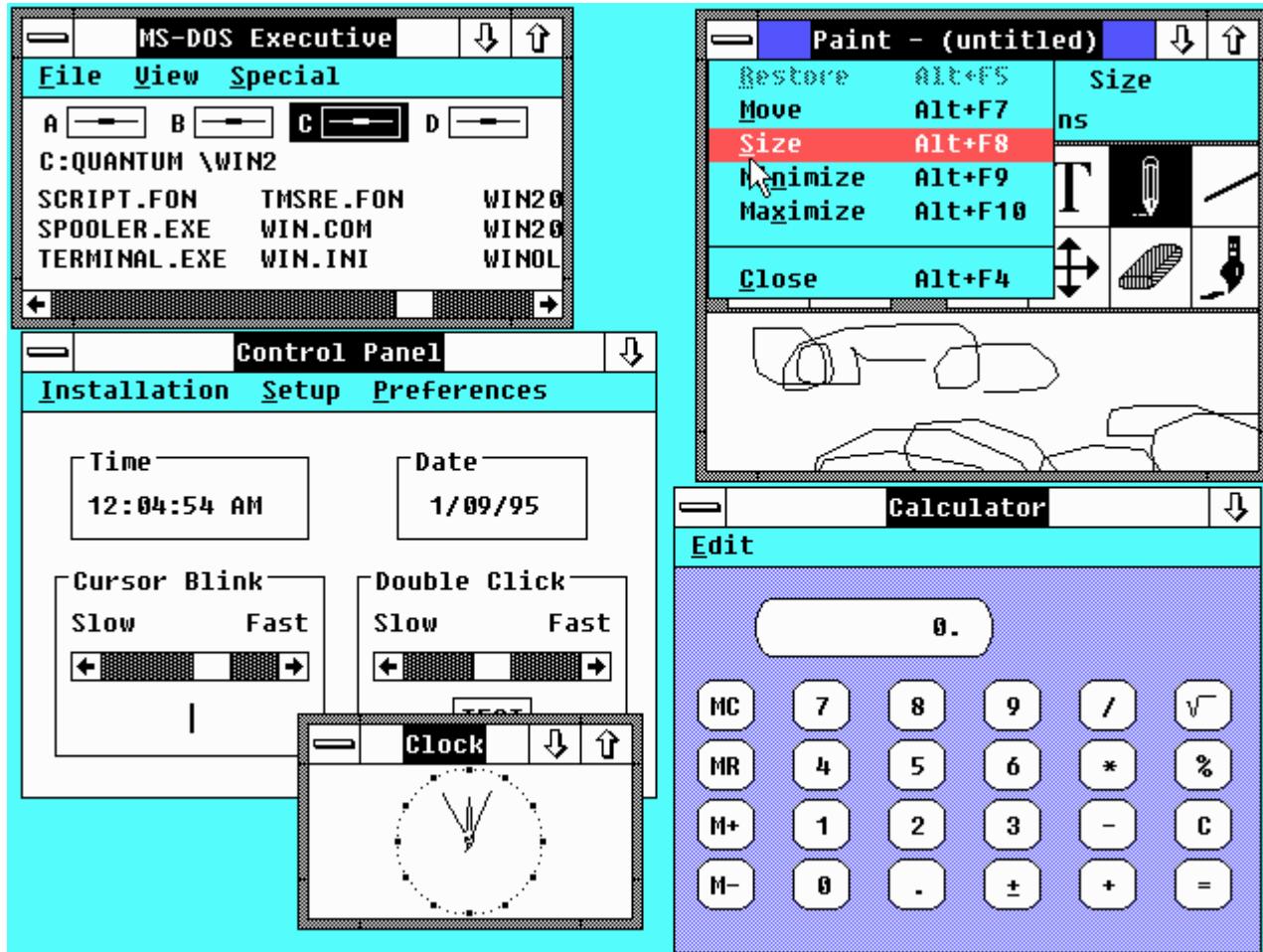


Microsoft Windows



- 1987'de 2.0 sürümü satışa çıkmıştır.
- Yeni grafiksel ofis uygulamaları olan Excel ve Word for Windows (Windows için Word) programlarının işletim sistemi içine dahil edilmesi ofis kullanıcılarının da bu işletim sistemini kullanmasını sağlamıştır.
- Bu sıralarda Aldus isimli bir yazılım firmasının Pagemaker ürününün Windows sürümü çıkarıldı.
- Önceki sürümleri yalnızca Macintosh'ta çalışıyordu.
- Bu olaydan sonra üreticiler programlarının Windows sürümlerini de hazırlamaya başladı ve Microsoft Windows'un yükselişi başlamış oldu.

Windows 2.0



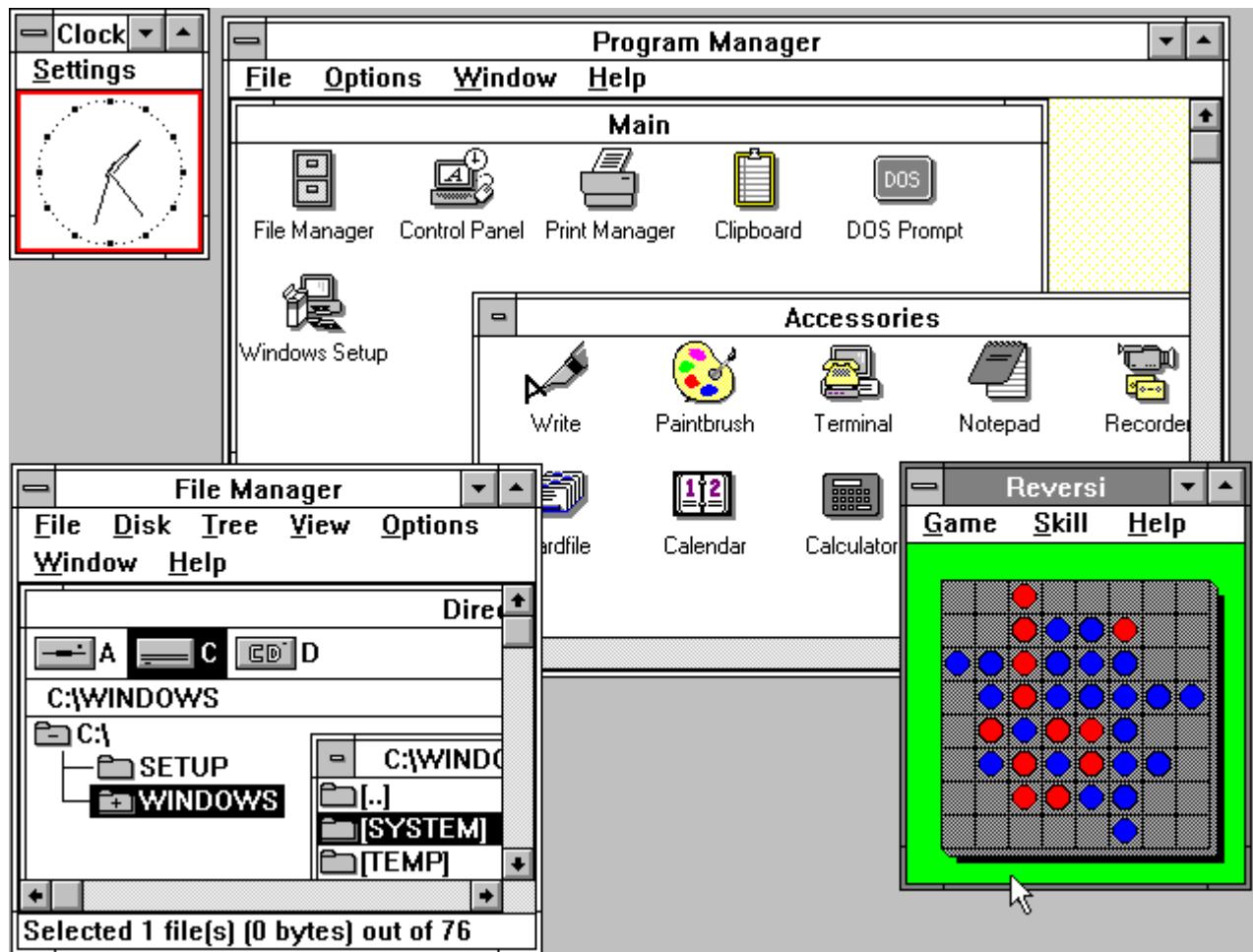


Microsoft Windows



- Microsoft Windows asıl başarısını 1990'da yapılan Windows 3.0 ile kazanmıştır.
- Yerel uygulamalarda geliştirilmiş kapasiteye ek olarak, Windows gerçek bellekler sayesinde kullanıcıya 386'daki MS-DOS tabanlı yazılımlara nazaran daha iyi çoklu görev yapma imkânı sunuyordu.
- Intel'in 8086/8088'den 80286 ve 80386'ya bütün işlemcilerinde çalışabilme başarısı da vardı.
- Ayrıca metin tarzı programları pencere içinde çalıştırabiliyor ve dosyaları liste şeklinde bir dosya yöneticisinde gösterebiliyordu.

Windows 3.0



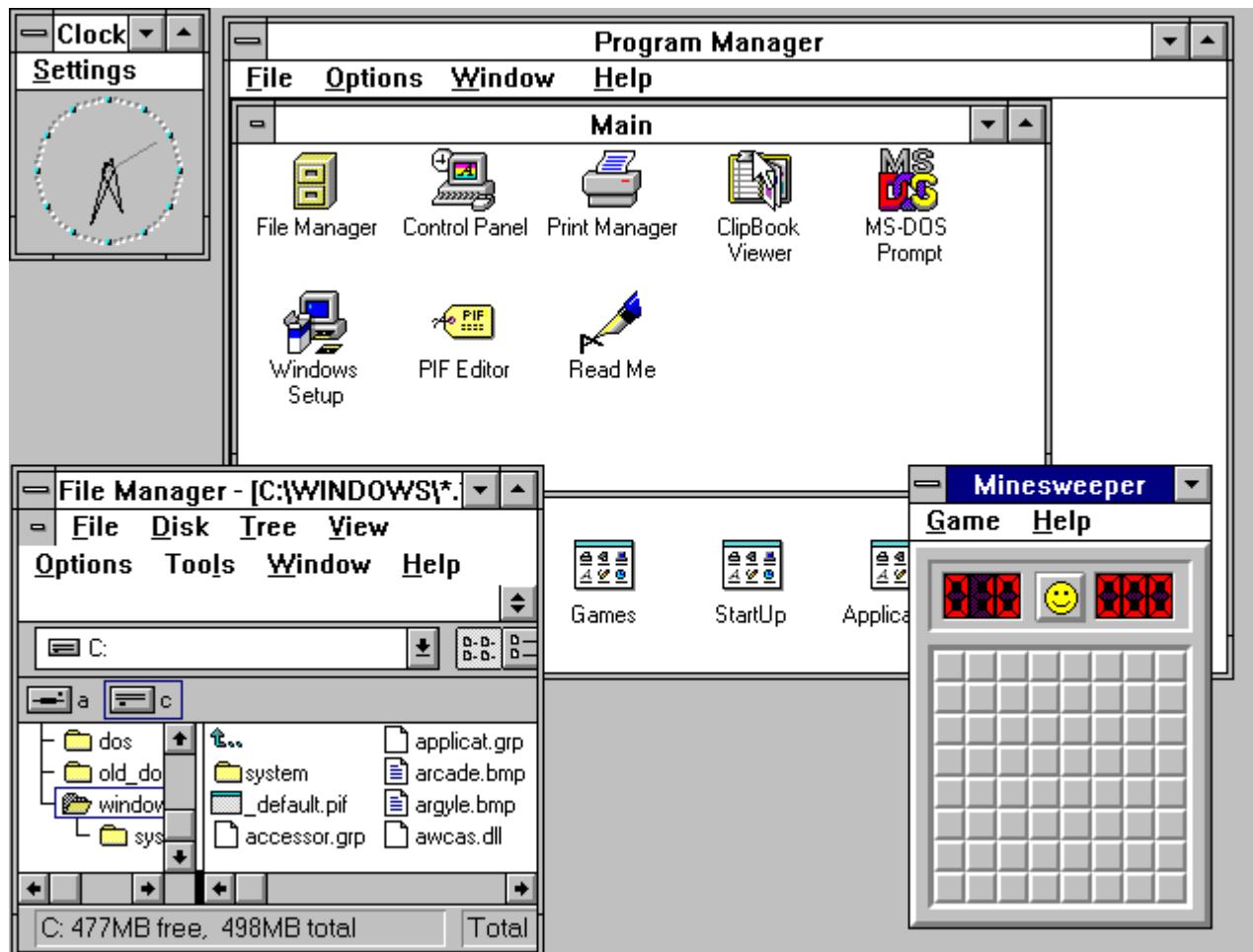


Microsoft Windows



- Microsoft, 1993 yılında Windows 3.1'i geliştirdi.
- Windows'un bu sürümü Windows 3.0'a bazı küçük yenilikler getiriyordu.
- Ölçeklendirilebilir TrueType yazı tiplerinin görüntülemesi sağlandı.
- Bazı hatalar düzeltildi.

Windows 3.1



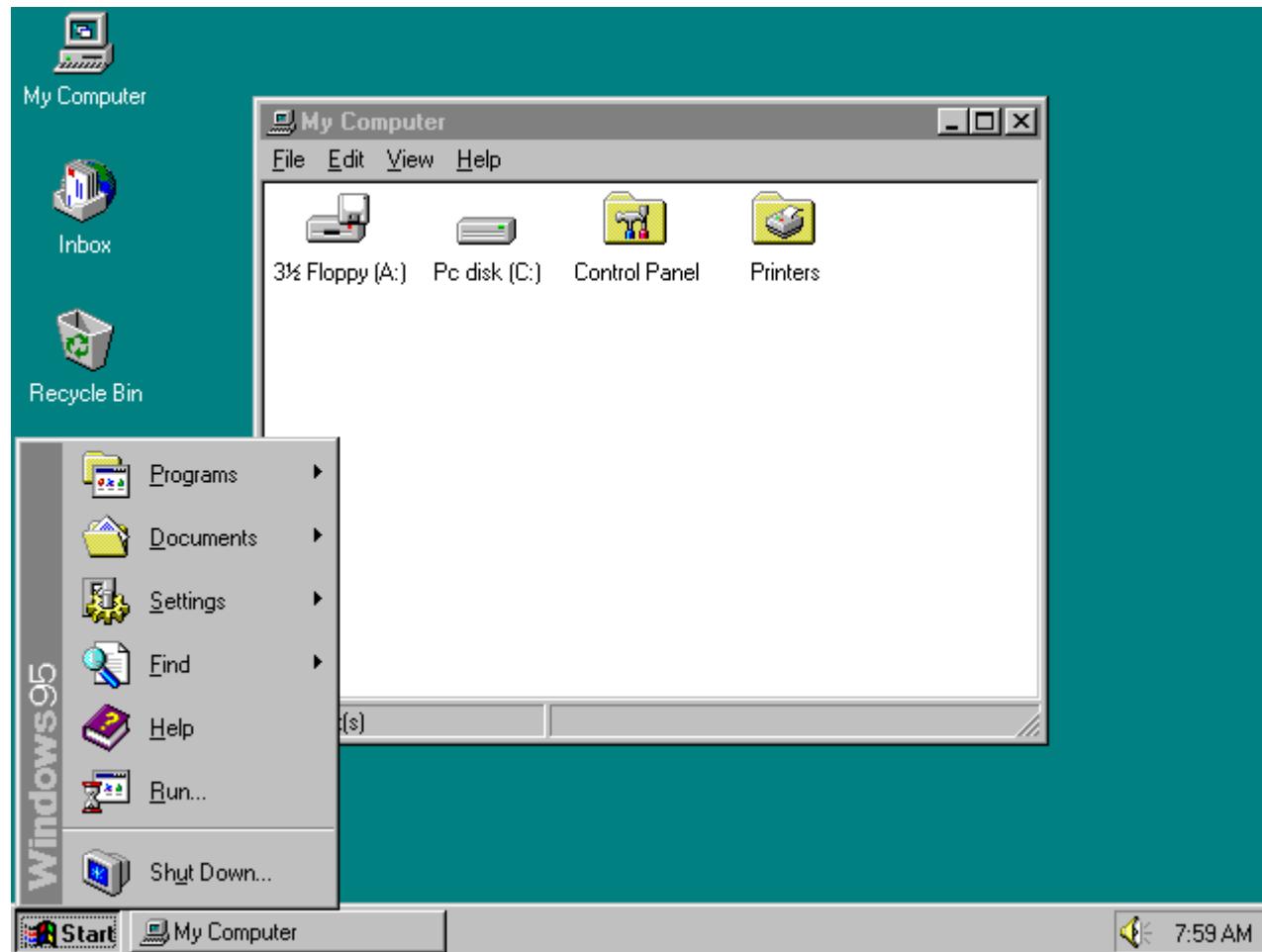


Microsoft Windows 95



- Arayüzde ciddi bir değişikliğe gidilen bu sürüm,
 - 255 karakter dosya isimleri
 - dahili **Internet Explorer**
 - *Masaüstü*nun işlevsellik kazanması,
 - *Başlat* düğmesigibi son kullanıcıya hitap eden özellikler de içeriyordu.
- Tamamen 32-bit olarak tasarlanmıştı.

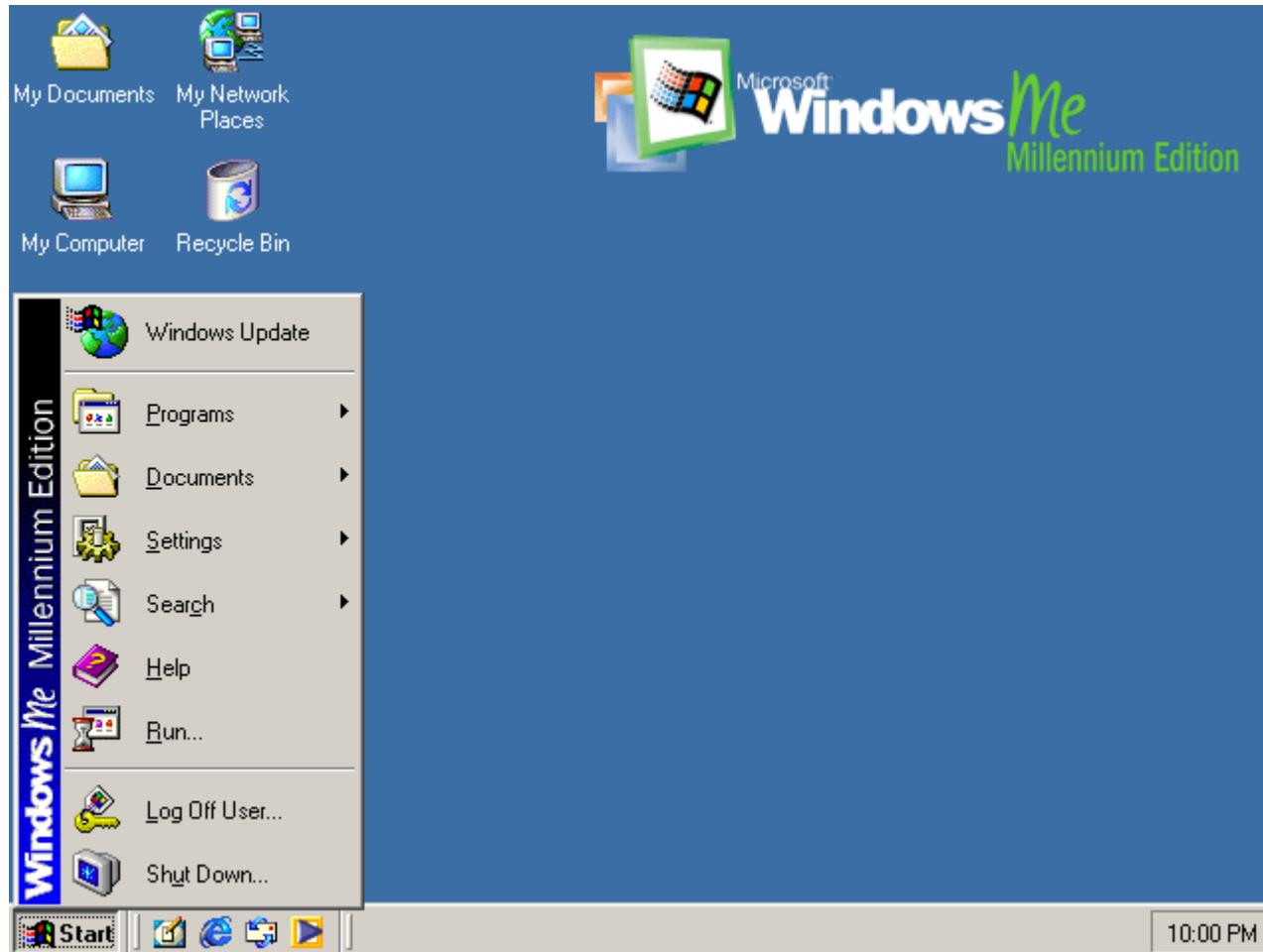
Windows 95



Windows 98



Windows ME





Windows NT Ailesi



- 1993 yılında sunucu pazarını hedefleyen Windows NT 3.1 piyasaya sürüldü. Grafiksel arayüzü Windows 3.1'e benzeyen bu işletim sistemi çok farklı bir altyapıya sahipti. Bu nedenle NT (New Technology) ön eki getirildi.
- 1994'te 3.5, 1995'te 3.51 ve 1996'da 4.0 sürümleri satışa sunuldu.
- 2000 yılında piyasaya sürülen Windows 2000 ve bu tarihten sonra geliştirilen tüm Windows işletim sistemleri NT çekirdeğine sahiptir.



Windows NT Ailesi



• 2000	Windows 2000	NT 5.0
• 2001	Windows XP	NT 5.1
• 2003	Windows Server 2003	NT 5.2
• 2006	Windows Vista	NT 6.0
• 2008	Windows Server 2008	NT 6.0
• 2009	Windows 7	NT 6.1
• 2009	Windows Server 2008 R2	NT 6.1
• 2012	Windows 8	NT 6.2
• 2013	Windows 8.1	NT 6.3
• 2015	Windows 10	NT 10.0



LINUX



- Linux, Andy Tannenbaum tarafından geliştirilmiş olan Minix işletim sistemine dayanmaktadır.
- Helsinki Üniversitesi'nde bilgisayar mühendisliği öğrencisi olan Linus Torvalds'ın 1991 yılında Intel'in yeni işlemcisi 80386'nın korumalı kip mimarisini üzere geliştirmeye başladığı Unix tabanlı bir işletim sistemi çekirdeğidir.
- Linus Torvalds boş zamanlarında Minix'den daha iyi bir Minix işletim sistemi yaratmak düşüncesiyle 1991 Ağustos sonlarında ilk çalışan Linux çekirdeğini oluşturdu.



LINUX



- 5 Ekim 1991 tarihinde 0.02 sürümü Linux ilk defa tanıtıldı.
- Linus, haber grubuna gönderdiği yazında yeni bir işletim sistemi geliştirmekte olduğunu ve ilgilenen herkesin yardımını beklediğini yazmıştır.
- İşletim sisteminin çekirdeği için verilen numaralar kısa sürede bir standart kazandı.
- a.x.y şeklinde belirtilen çekirdek türevlerinde y bulunulan seviyeyi, x gelişim aşamasını göstermektedir. a ise değişik Linux sürümlerini belirtir.

Özgür Yazılım Nedir?

- Özgür yazılım, 4 farklı özgürlüğü temel alır:
 - Hangi amaç için olursa olsun, programı çalıştırma özgürlüğü
 - Kendi ihtiyaçlarınıza uyarlamak için programın nasıl çalıştığını öğrenme özgürlüğü (Kaynak koduna erişim bunun için bir önkoşuldur)
 - İhtiyacı olanlara yardım edebilmeniz için programın kopyalarını dağıtma özgürlüğü
 - Bütün toplumun faydalananabilmesi için programı iyileştirme ve yaptıklarınızı halka açma özgürlüğü (Kaynak koduna erişim bunun için bir önkoşuldur)

Kaynak: www.gnu.org

Linux Dağıtımları

- Farklı çalışma grupları tarafından Linux çekirdeği üzerinde geliştirilmiş olan işletim sistemlerine Linux Dağıtımları denir.
- Bilgisayara kurulmadan CD, DVD veya USB Bellek üzerinden doğrudan çalıştırılabilen dağıtımlar da vardır.
- Linux dağıtımları daha çok sunucularda kullanılmasına rağmen masaüstü ve dizüstü bilgisayarlarda da kullanım oranı artmaktadır.
- Gömülü sistemlerden, süper bilgisayarlara kadar birçok donanım türünde kullanılan Linux dağıtımları da vardır.

Linux Dağıtımları: redhat

- 1993 yılında ABD'de kurulmuş Red Hat şirketinin geliştirdiği ticari Linux dağıtımasıdır.
- İyi test edilmiş, nispeten kararlı bir Linux dağıtımı olması nedeniyle, dünya çapındaki sunucuların bir çoğunda tercih edilmektedir
- Hata rapor etme ve beta programlar kullanıcılarına açıktır ve çok sayıda e-posta listesi bulunmaktadır.

Linux Dağıtımları: fedora

- Red Hat Linux'un devamı olarak adlandırılacak Fedora Core hiçbir kar amacı gütmeyen bir organizasyon tarafından üretilmektedir.
- Her ne kadar bir zamanlar efsane olan Red Hat kadar büyük bir popülerliği yoksa da Linux dünyasının en sevilen ve en başarılı dağıtımlarından biridir.
- Mandriva veya Suse kadar olmasa da oldukça güncel paketlere sahip Fedora onlardan daha sağlam ve kararlı bir dağıtım olması ile ünlüdür.

Linux Dağıtımları:



- Almanya'da hazırlanmakta olan bir Linux dağıtımasıdır.
- İsmi "**S**oftware **u**nd **S**ystem **E**ntwicklung" ("Software and system development") sözcüklerinin baş harflerinin bir araya gelmesiyle oluşmuştur.
- SuSE dağıtımı 2003'ten beri Novell şirketine aittir.

Linux Dağıtımları:



- En yaygın Linux dağıtımlarından biridir.
- Mepis, Ubuntu, Yoper, Knoppix, Libranet, Linspire, Xandros ve Adamantix gibi birçok Linux dağıtımına da kaynak teşkil etmekte ve Google başta olmak üzere iyi tanınan birçok Web sitesinde de tercih edilmektedir.
- Farklı işletim sistemi çekirdekleriyle birlikte i386, AMD64, PowerPC, SPARC, DEC Alpha, ARM, MIPS, HPPA, S390, IA-64 gibi çok sayıda donanım platformunda da çalışabilmektedir.

Linux Dağıtımları: ubuntu

- Debian'ı temel alan Ubuntu'nun hedefi ortalama bilgisayar kullanıcılarına kullanımı ve kurulumu oldukça basit, güncel ve güvenli bir işletim sistemi sunmaktır.
- Güney Afrikalı bir girişimci olan Matthew Shuttleworth'ün sponsor olduğu Ubuntu, bir şirket içinde değil, bunun aksine dünyanın çeşitli yerlerindeki programcılar sayesinde geliştirildi.
- 2004 yılında ortaya çıkan Ubuntu aldığı %30'luk oyla, 2007 yılında masaüstü kullanımında en gözde Linux dağıtımı seçilmiştir.

Linux Dağıtımları:



- Türkiye'de TUBİTAK bünyesinde Pardus isimli bir dağıtım geliştirilmektedir.
- Pardus her seviyede bilgisayar kullanıcısının kolayca kurup kullanabilmesi amaçlanarak geliştirilmektedir. Bu nedenle kurulumundan yönetimine kadar pek çok araç ve teknoloji Pardus geliştiricileri tarafından özel bir proje olarak geliştirilmektedir.

Linux (sunucu-süper bilgisayar)

- Linux ürünleri sunucu işletim sistemi olarak uzun zamandır oldukça yaygın bir şekilde kullanılmaktadır, 2008 Eylül ayında Microsoft CEO'su Steve Ballmer, dünya genelinde web sunucularının %60'ında Linux'un, %40'ında Windows'un kullanıldığını itiraf etmiştir.
- Linux dağıtımları, süper bilgisayarlarda işletim sistemi olarak da yaygın şekilde kullanılmaktadır: Kasım 2010 tarihinden beri en iyi 500 sistem arasında 459'u (%91.8) Linux dağıtımı kullanmaktadır.
- Ayrıca dünyadaki güçlü süper bilgisayarlardan olan ve 2011'de kullanılmaya başlanan IBM Sequoia için de işletim sistemi olarak seçilmiştir.

Android

- Google ve Open Handset Allience (OHA) tarafından geliştirilen açık kaynak kodlu bir işletim sistemidir.
- Linux çekirdeğine sahiptir ancak alıştılageldik Linux çekirdekli işletim sistemlerinde bulunan bazı temel özellikleri ve parçaları içermediği için genel anlamda bir Linux işletim sistemi olarak kabul edilmez.
- İşletim sistemi kodlanması yaklaşık 12 milyon (3 milyon XML , 2.8 milyon C , 2.1 milyon Java, 1.75 milyon C++ ve diğer kodlar) satır kod yazılmıştır.
- Eski Yunancada insanımsı makine anlamına gelen Android işletim sistemi için yeşil robotumsu bir logo kullanılmaktadır.
- Android içerisinde bir Java Sanal Makinası (DALVIK) bulunur, bu sebeple Java dili ile yazılım geliştirilir.

Mac OS

- Macintosh İşletim Sistemi, kısaca Mac OS, Apple firması tarafından piyasaya sürülen ticari bir işletim sistemi ailesidir. İlk kez 1984 yılında ortaya çıkan Mac OS, grafik arayüz tabanlı bir işletim sistemidir.
- MAC OS, UNIX türevi kapalı kaynak kodlu bir işletim sistemidir. Yasal olarak sadece Apple marka bilgisayarlar ile uyumludur.

iOS

- Apple tarafından geliştirilen Mac OS X (Unix türevli) işletim sistemi ailesinden gelmiştir. Apple marka mobil cihazlar için özel tasarlanmıştır.
- Sadece parmak etkileşimi ile çalışacak biçimde tasarlanmıştır.
- Çoklu dokunma özelliğini desteklemektedir.
- Kapsamlı uygulama geliştirme ortamı (iPhone SDK) sayesinde uygulama açısından en zengin platformdur.
- Uygulama geliştirmek için Mac OS yüklü bir bilgisayara ihtiyaç vardır.
- Java ve Flash desteği yoktur.

SON

Mikroişlemciler ve Mikrodenetleyiciler

GİRİŞ

- ❖ Mikroişlemci Nedir?
- ❖ Mikroişlemcileri Birbirinden Ayıran Özellikler
- ❖ Mikroişlemciyi Oluşturan Birimler ve Görevleri
- ❖ Bellekler
- ❖ Mikrodenetleyiciler
- ❖ Mikroişlemci ve Mikrodenetleyiciler Arasındaki
- ❖ Mikrodenetleyici Seçimi
- ❖ ARDUINO nedir?
- ❖ Neden ARDIUNO?
- ❖ DONANIM
- ❖ YAZILIM
- ❖ Örnek Uygulamalar

Mikroişlemci Nedir?

- Mikroişlemciler, bilgisayar sisteminin kalbidir.
- Bilgisayar operasyonlarını kontrol ederek veri işleme işlevlerini yerine getirir.
- CPU (Central Process Unit-Merkezi İşlem Birimi), kullanıcı ya da programcı tarafından yazılan programları meydana getiren komutları veya bilgileri yorumlamak ve yerine getirmek için gerekli olan tüm mantıksal devreleri kapsar.
- İlk mikroişlemci 1971 yılında hesap makinası amacıyla üretilen Intel firmasının 4004 adlı ürünüdür. Bir defada işleyebileceği verinin 4 bit olmasından dolayı 4 bitlik işlemci denilmektedir.

Mikroişlemcileri Birbirinden Ayıran Özellikler

- 1) **Kelime Uzunluğu:** Mikro işlemcinin **her saat darbesinde işlem yapabileceği bit sayısına** kelime uzunluğu denir. İşlenen veriler işlemcinin özelliğine göre 4-bit, 8-bit, 16-bit, 32-bit ve 64-bit uzunlığında olabilir. Kelime uzunluğu **veri yolu uzunluğuna** eşittir.
- 2) **Komut İşleme Hızı:** Mikro işlemcilerin çalışması için **saat sinyallerine** ihtiyaç vardır. İşlemci (CPU) her saat sinyalinde bir sonraki işlem basamağına geçer. Saat frekansı mikro işlemciye dışardan uygulanan ya da işlemcinin içinde bulunan osilatörün frekansıdır. Komut çevrim süresi ise herhangi **bir komutun görevini tamamlayabilmesi için geçen süredir.**

Mikroişlemcileri Birbirinden Ayıran Özellikler

3)Adresleme Kapasitesi: Bir işlemcinin adresleme kapasitesi, adresleyebileceği veya **doğrudan erişebileceği bellek alanının büyüklüğüdür**. Bu büyülüük işlemcinin adres hattı sayısına bağlıdır. Bu hattın sayısı tasarlanacak sistemde kullanılabilecek bellek miktarını da belirlemektedir.

4)Kaydedici Sayısı: Mikro işlemcilerde kaydediciler, genel amaçlı kaydediciler ve özel amaçlı kaydediciler olmak üzere iki grupta toplanır. Bu kaydediciler 8, 16, 32 ve 64- bitlik olabilir. Kaydedicilerin sayısının programcının işinin kolaylaştırmasının yanında programın daha sade ve anlaşılır olmasını da sağlar. Her mikro işlemcinin kendine has yapısı ve kaydedici isimleri vardır. Herhangi bir mikro işlemciyi programlamaya başlamadan önce mutlaka bu kaydedicilerin isimlerinin ve ne tür işlevlere sahip olduklarının iyi bilinmesi gereklidir.

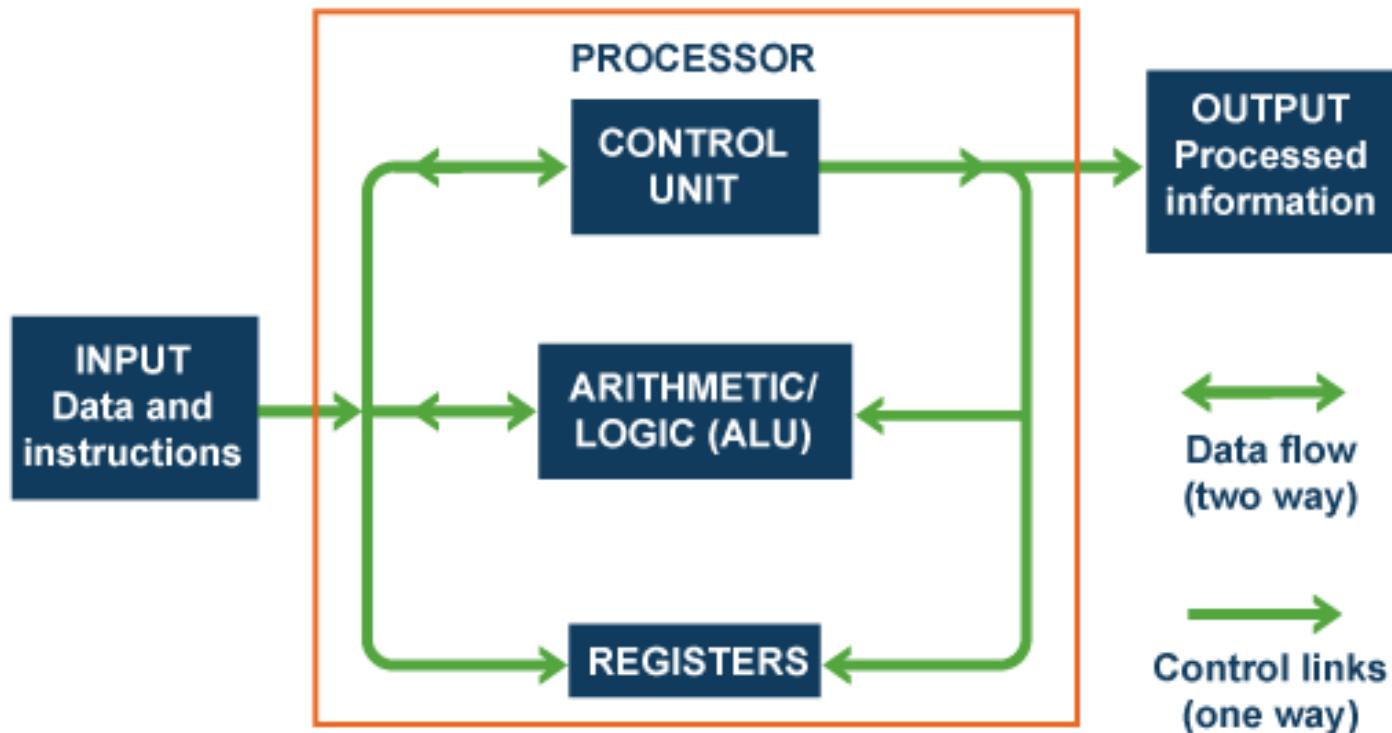
Mikroişlemcileri Birbirinden Ayıran Özellikler

5)Farklı Adresleme Modları: Bir komutun işlenmesi için gerekli verilerin bir bellek bölgesinden alınması veya bir bellek bölgесine konulması ya da bellek-kaydedici veya kaydedici-kaydedici arasında değiştirilmesi için farklı erişim yöntemleri kullanılır. Mikro işlemcinin işleyeceği bilgiye farklı erişim şekilleri, "adresleme yöntemleri" olarak ifade edilir. Kısaca **adresi tarif yollarıdır.**

Adresleme türleri;

- ❖ Doğrudan adresleme
- ❖ Dolaylı adresleme
- ❖ Veri tanımlı adresleme
- ❖ Kaydedici adresleme
- ❖ Mutlak adresleme
- ❖ Göreceli adresleme
- ❖ İndisli adresleme
- ❖ Akümülatör ve imalı adresleme

Mikroişlemciyi Oluşturan Birimler ve Görevleri



Mikroişlemciyi Oluşturan Birimler ve Görevleri

1) Kaydediciler (Registers)

Kaydediciler; genel ve özel amaçlı olmak üzere iki gruba ayrılır. Bunlardan başka programcıya gözükmeyen (ilgilendirmeyen) kaydediciler de vardır. Genel amaçlılara akümülatör, indis kaydedicileri girmektedir. Özel amaçlılar ise program sayıcı, yoğun işaretçisi, bayraklar gibi kaydediciler girmektedir.

A) Akümülatör: Akümülatörler, işlemcinin aritmetik ve mantık işlemleri sırasında depo görevi yapan önemli bir kaydedicidir. **Ara değerlerin üzerinde tutulması**, sisteme gelen verinin ilk alındığı yer, belleğe veya dış dünyaya gönderilecek verilerin tutulduğu yer olarak görev yapar. Bazı işlemcilerde B kaydedicisi de yardımcı akümülatör olarak kullanılır.

B) İndis Kaydedicileri: İndis kaydedicilerinin temelde üç görevi vardır. Hesaplamlarda **ara değerlerin geçici tutulmasında**, **program döngülerinde** ve zamanlama uygulamalarında **bir sayıcı olarak** ve bellekte depolanmış bir dizi verinin üzerinde bir **indisçi** olarak kullanılmaktadır.

Mikroişlemciyi Oluşturan Birimler ve Görevleri

1) Kaydediciler

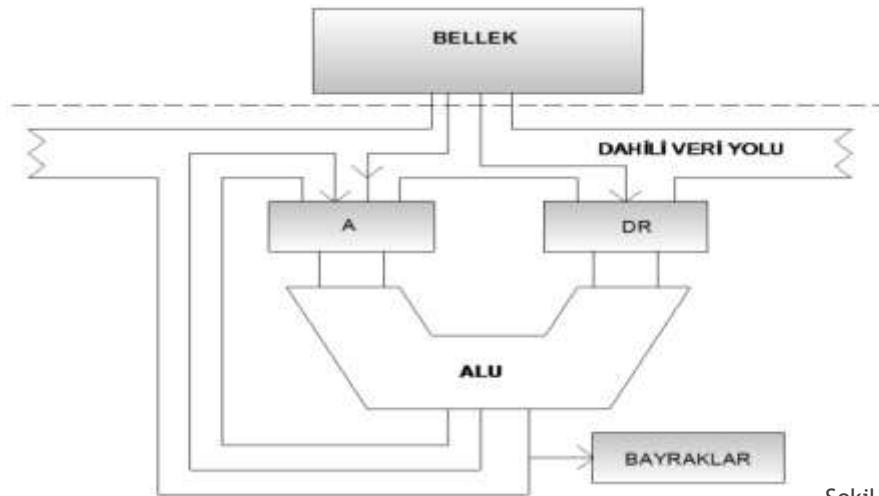
C) Program Sayıcı (PC, Program Counter) : Mikro işlemcinin yürütme sırasında olduğu program komutlarının adres bilgisini tuttuğu özel amaçlı bir kaydedicidir. Mikroişlemcinin çalışması sırasında hangi komutun hangi sırada kullanılacağıının bilinmesi gereklidir. Bu görevi program sayacı (PC) yerine getirir. Bellekten alınan her komut kodundan sonra alınacak yeni komut kodunun adresi program sayıcıya otomatik olarak işlemci tarafından yüklenir. Komut çevrimi, PC'nin yeni adresi adres yoluna koyması ile başlar. Bunun ardından da ilgili kontrol sinyali gönderilir. Bellekten gelen her bilgiden sonra PC, kontrol devresinden aldığı işaretle uyarak adres satırını 1 arttırır. Böylece bilgilerin bellekten işlemciye düzenli bir şekilde gelmesi sağlanır.

D) Durum Kaydedicileri (Bayraklar): Mikro işlemci içinde veya dışardan yapılan herhangi aritmetiksel, mantıksal veya kesmelerle ilgili işlemlerin sonucuna göre bitleri değer değiştirir. Bir işlem sonucunda bu bitlerin aldığı değere göre program yön bulur. Programcı bu bitlerde oluşacak değerlere göre programa yön verebilir. Elde Bayrağı-C, Sıfır Bayrağı-Z, Ondalık Bayrağı-D, Taşma Bayrağı-V, Negatif Bayrağı-N gibi çeşitleri vardır.

Mikroişlemciyi Oluşturan Birimler ve Görevleri

2) Aritmetik ve Mantık Birimi (ALU)

Mikroişlemcinin en önemli kısmını aritmetik ve lojik birimi (ALU) oluşturur. Bu ünite kaydediciler üzerinde toplama, çıkarma, karşılaştırma, kaydırma ve döndürme işlemleri yapar. Yapılan işlemin sonucu kaydediciler üzerinde saklanır. Bazen de yalnızca durum kodu kaydedicisini etkiler. ALU'daki bir işlem sonucunda durum kodu kaydedicisindeki bayrakların birkaçı etkilenebilir veya hiçbir etkilenmez. Programcı için çoğu zaman ALU'da yapılan işlemin sonucunda etkilenen bayrakların durumu daha önemlidir. Gelişmiş mikroişlemcilerin içindeki ALU'lar çarpma ve bölme işlemlerini yapabilmektedir. ALU'nun işlem yapabileceği en büyük veri, mikro işlemcideki kaydedicilerin veri büyüklüğü ile sınırlıdır.



Şekil-2

Mikroişlemciyi Oluşturan Birimler ve Görevleri

2) Aritmetik ve Mantık Birimi (ALU)

A) Aritmetik İşlemler : ALU'da yapılan aritmetiksel işlemler mikro işlemcinin yapısına göre çeşitlilik gösterebilir. Toplama, Çıkarma, Çarpma, Bölme, Sağa veya sola kaydırma, içerik artırma veya azaltma ve döndürme işlemleri gibi işlemleri kapsar. Gelişmiş işlemcilerde büyük ondalıklı sayılarla işlem yapmak için ayrıca matematik işlemci mevcuttur.

B) Mantıksal İşlemler : VE, VEYA, Özel VEYA (XOR), Değil (NOT), Karşılaştırma ($=$, $=<$, $=>$, $<>$ gibi) gibi işlemleri kapsar.

Bütün bu işlemler teknolojik yapısı değişik kapı ve flip-flop'lardan oluşan bir sistem tarafından yürütülmektedir.

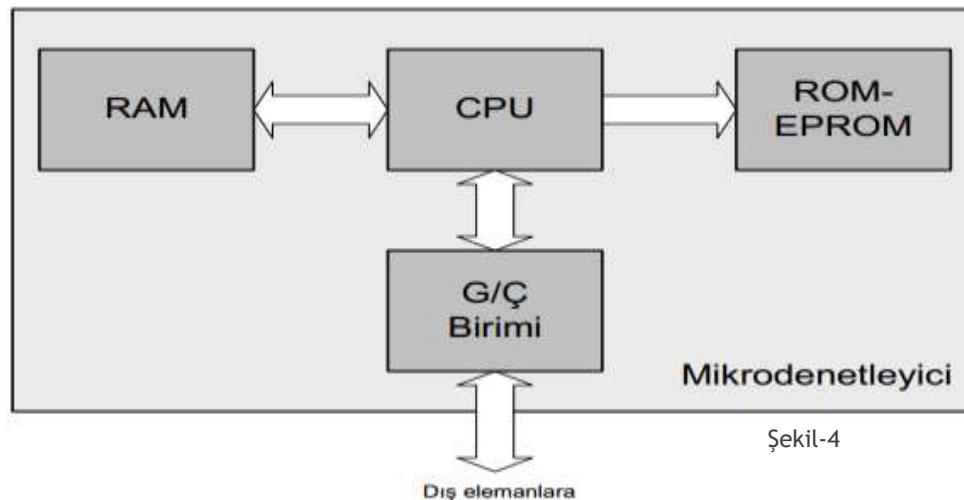
3) Kontrol Birimi

Kontrol birimi, sistemin tüm işleyişinden ve işlemin zamanında yapılmasından sorumludur. Kontrol birimi, bellekte program bölümünde bulunan komut kodunun alınıp getirilmesi, kodunun çözülmesi, ALU tarafından işlenmesi ve sonucun geri belleğe konulması için **gerekli olan kontrol sinyalleri** üretir.

Mikrodenetleyiciler

Bir mikroişlemci çekirdeğine ilave olarak, ortak bellek alanlarını kullanan, özelleştirilmiş görevler ile donatılmış çevrebirimlerin eklenmesi ile ortaya çıkan yapıya mikrodenetleyici denir. Denetim teknolojisi gerektiren uygulamalarda kullanılmak üzere tasarlanmış olan mikrodenetleyiciler, mikro işlemcilere göre çok **daha basit ve ucuzdur**.

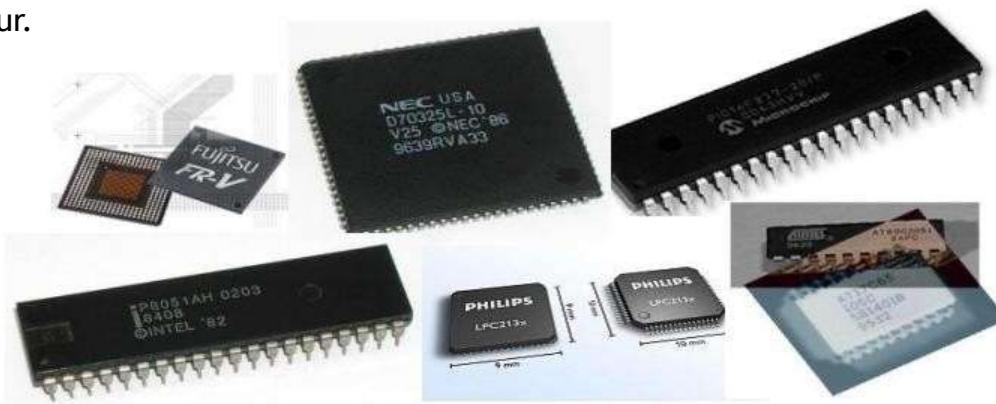
- ❖ Mikroişlemcili sistemin tasarımını ve kullanımı mikrodenetleyicili sisteme göre daha karmaşık ve masraflıdır.
- ❖ Mikrodenetleyicili bir sistemin çalışması için elemanın kendisi ve bir osilatör kaynağının olması yeterlidir.
- ❖ Mikrodenetleyicilerin küçük ve ucuz olmaları, bunların tüm elektronik kontrol devrelerinde kullanılmasını sağlamaktadır.



Şekil-4

Mikroişlemci ve Mikrodenetleyiciler Arasındaki Farklar

Bir mikroişlemci görevini yerine getirebilmesi için mutlaka, verilerin saklanacağı bellek birimine, dış dünyadan veri alışverişinin düzenli yapılmasını sağlayan giriş/çıkış birimine ihtiyaç duyar. Bunlar bir mikroişlemcili sistemde ayrı ayrı birimler (entegreler) şeklinde yerini alır. Bundan dolayı mikroişlemcili sistemlere çok entegreli sistemler denilir. Bilgisayar gibi mikroişlemcili sistemlere verilen bir örnekte, bir bilgisayarın bir çamaşır makinesinde veya cep telefonunda kullanılması elbette mümkün olmayacağıdır. Bilgisayar aynı anda milyonlarca işi yapabildiğinden ve çok yer kapladığından böyle yerlerde kullanılması mantıklı olmaz ve maliyetli olur. Bundan dolayı, sistemi meydana getiren elemanların birçok özelliklerinden feragat edilerek ve bir entegrede birleştirilerek mikroişlemcilerin yeni türevleri (mikrodenetleyiciler) oluşturulmuştur. Bir saydırma sinyali üreteceğimizi düşünürsek mikroişlemci ile bunu yazılımsal olarak yapmamız gerekecektir. Ancak mikrodenetleyicinin özelleştirilmiş modülleri sayesinde bu işlemleri programa paralel olarak modüllerle yapabiliriz. Böylece ana programdaki yoğunluk azalır ve işlemcimiz hızlanmış olur.

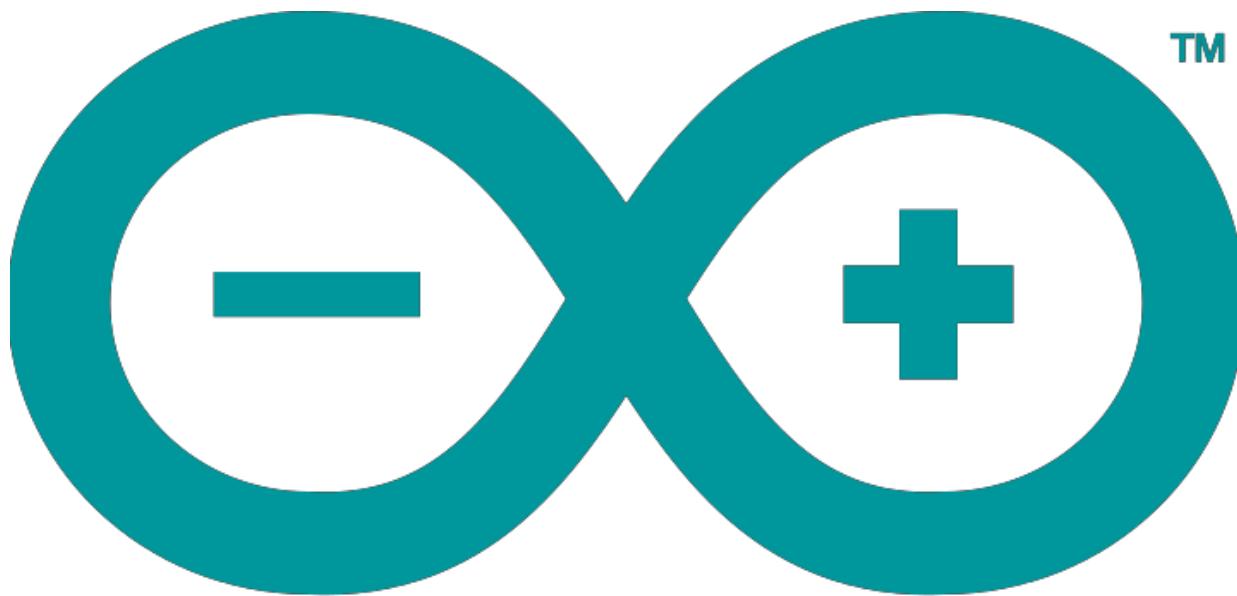


Şekil-5

Mikrodenetleyici Seçimi

Mikrodenetleyiciler ile tasarım yapmadan önce tasarlanan sisteme uygun bir denetleyici seçmek için o denetleyicinin taşıdığı özelliklerin bilinmesi gereklidir. Mikrodenetleyicinin hangi özelliklere sahip olduğu kataloglarından anlaşılabilir. Aşağıda sıralanan özellikler bunlardan bazlılarıdır;

- ❖ Programlanabilir dijital paralel giriş/çıkış.
- ❖ Programlanabilir analog giriş/çıkış.
- ❖ Seri giriş/çıkış (senkron, asenkron ve cihaz yönetimi).
- ❖ Motor veya servo kontrol için puls sinyali çıkışı.
- ❖ Harici giriş vasıtasyyla kesme.
- ❖ Harici bellek arabirimleri.
- ❖ Harici veri yolu arabirimleri.
- ❖ Dahili bellek tipi seçenekleri (ROM, EPROM, PROM, EEPROM).
- ❖ Dâhilî RAM seçeneği.
- ❖ Kayan nokta hesaplaması.



ARDUINO

ARDUINO Nedir?

Arduino, açık kaynak kodlu yazılım ve donanıma sahip bir **mikrodenetleyici platformudur**. Açık kelimesi ile gerçek anlamda açık tasarımını ifade edilmektedir. Baskılı devresi, şematik tasarımları, PC üzerinde çalışan derleyicisi, kütüphaneleri ve **tüm detayları ile internet ortamında paylaşılmaktadır**.



Şekil-7

Neden ARDUINO?

- ❖ Hem donanımsal hem de yazılımsal olarak açık kaynaklı.
- ❖ Ucuz.
- ❖ Alt seviye mikroişlemci bilgisi gerektirmez.
- ❖ Çok zengin kütüphaneleri vardır.
- ❖ Ek bir programlama devresi gerektirmez.
- ❖ USB üzerinden haberleşebilir.
- ❖ Shield kartları sayesinde ileri teknolojiler kolayca entegre edilebilir.
- ❖ Temel elektronluğun bilinmesi uygulama geliştirmek açısından yeterlidir.

DONANIM



Arduino Uno



Arduino Leonardo



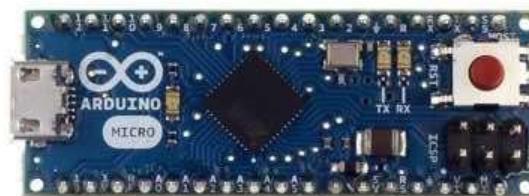
Arduino Due



Arduino Tre



Arduino Yun



Arduino Micro



Arduino Robot



Arduino Esplora

DONANIM



Arduino Mega ADK



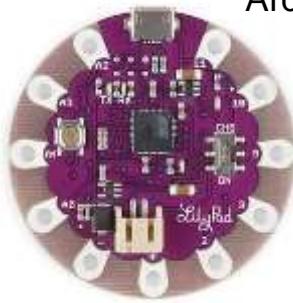
Arduino Ethernet



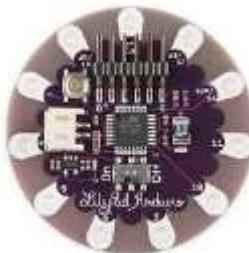
Arduino Mega 2560



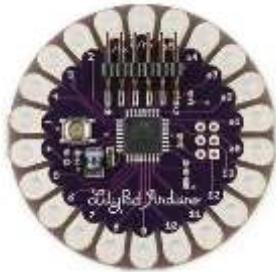
Arduino Mini



LilyPad Arduino USB



LilyPad Arduino Simple



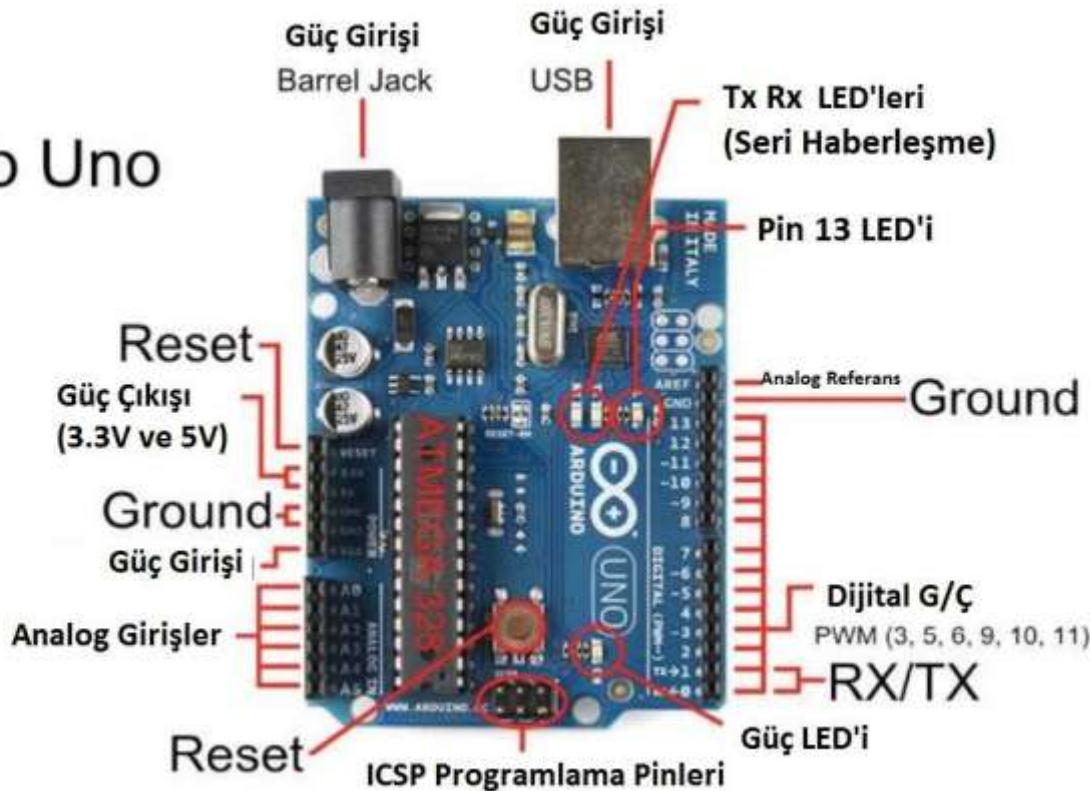
LilyPad Arduino



Arduino Nano

DONANIM

Arduino Uno



Şekil-8

YAZILIM

Derleme

İşlemciye
Yükleme

Serial
Monitör

Kod Editörü

Program
Çıktısı

```
void setup()
{
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
```

Arduino Uno on COM1

Şekil-9

YAZILIM

setup() : Programın **başlangıç** ayarları içindir sadece başlangıçta tek seferlik çalışır. Bu fonksiyon değişkenler, pin modları, seri iletişim, kütüphaneler vb. için kullanılır.

Loop() : Loop Türkçesi **döngü**dür. Adından anlaşılacağı gibi setup() fonksiyonundan sonra **döngü** şeklinde sürekli çalışır. Ana program kodları bu fonksiyon içine yazılır

```
//Tanımlamalar
void setup() {
// Kurulum kodları buraya yazılır. (Bir defa çalışır)
}
void loop() {
// Ana program kodları buraya yazılır. (Sürekli çalışır)
}
```

YAZILIM

- ❖ Program yazımı belirli kalıpta, bloklar halinde olur.
- ❖ Bloklar, { } parantezleri ile oluşturulur.
- ❖ Komutlar aynı veya alt alta satırlara yazılabilirler.
- ❖ Tüm komutlar noktalı virgül (;) ile biter. Yalnız blok başlatan ifadelerden sonra noktalı virgül kullanılmaz.
- ❖ Programda kullanılan tüm değişkenler ve bilgi tipleri bildirilir.
- ❖ Programın başında kütüphaneler aktifleştirilir/çağrılır.
- ❖ Açıklamalar “//” ve “/* */ ” (Birden fazla satır için) ile yazılır.
- ❖ #define ile eşdeğer ifade atanır.
- ❖ #include ile kütüphane çağrılır.

YAZILIM

pinMode(pin no, pin türü);

Pinleri giriş veya çıkış olarak yapılandırma işlemi yapar.

digitalWrite(pin no, lojik değer);

Çıkış olarak ayarlanan pinlerin değerlerini, HIGH veya LOW olarak ayarlar.

digitalRead(Pin no);

Belirtilen digital pin değerini okur.

analogReference();

Analog giriş için referans gerilimini ayarlar.

analogRead(Pin no);

Belirtilen analog pin değerini okur.

analogWrite(pin no, dijital değer);

Ayarlanan pinden analog çıkış almayı sağlar. LED parlaklıği, motor hızı ayarlama gibi işlemlerde kullanılır.

delay(Milisaniye);

Milisaniye biriminde zaman gecikmesi verir.

Örnek Uygulamalar

LED Uygulaması

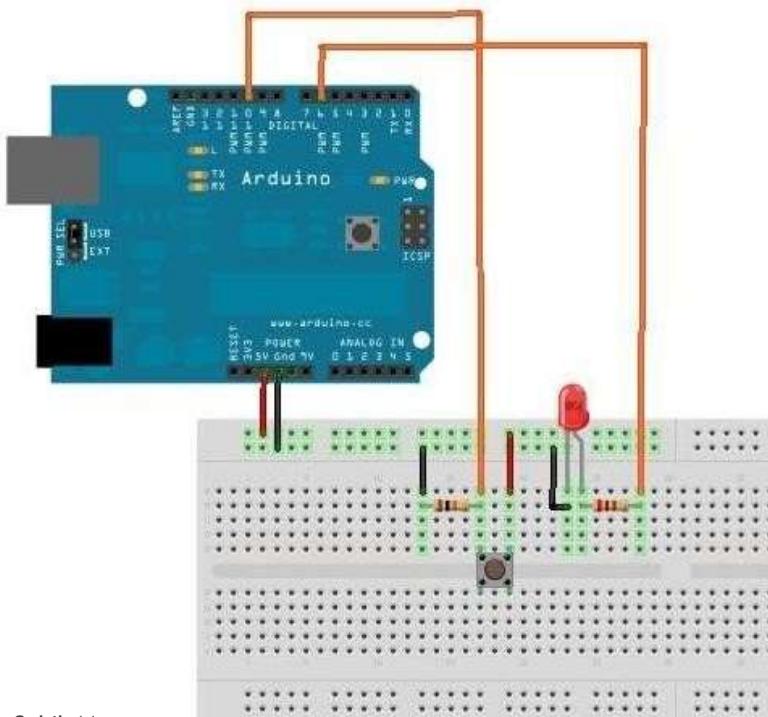


Şekil-10

```
1. #define LED 13
2. void setup()
3. {
4.   pinMode(LED, OUTPUT);
5. }
6. void loop() {
7.   digitalWrite(LED, HIGH);
8.   delay(1000);
9.   digitalWrite(LED, LOW);
10.  delay(1000);
11. }
```

Örnek Uygulamalar

Dijital Okuma-Seri İletişim-LED uygulaması

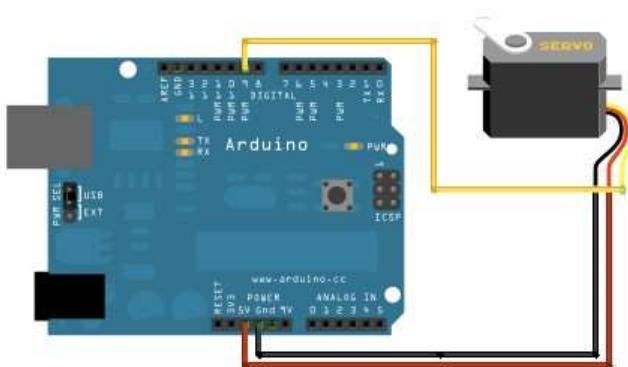


Şekil-11

```
1. int Buton = 10;
2. int LED = 6;
3. void setup()
4. {
5.   Serial.begin(9600);
6.   pinMode(Buton, INPUT);
7.   pinMode(LED, OUTPUT);
8. }
9. void loop()
10.{ 
11.   int butonDurumu = digitalRead(Buton);
12.   digitalWrite(LED, butonDurumu);
13.   Serial.println(butonDurumu);
14.   delay(1);
15. }
```

Örnek Uygulamalar

Servo Motor Uygulaması



Şekil-14

```
1. #include <Servo.h>
2. Servo SERVO1;
3. int pozisyon = 0;
4. void setup()
5. {
6.   SERVO1.attach(9);
7. }
8. void loop()
9. {
10.  for(pozisyon = 0; pozisyon <= 180; pozisyon++)
11.  {
12.    SERVO1.write(pozisyon);
13.    delay(15);
14.  }
15.  for(pozisyon = 180; pozisyon>=0; pozisyon--)
16.  {
17.    SERVO1.write(pozisyon);
18.    delay(15);
19.  }
20:}
```

Bilgisayar Ağları ve Internet Teknolojisi

Bilgisayar Ağı Nedir?

- İki veya daha fazla bilgisayarın bir araya gelerek **belirli bir protokol altında** iletişimde bulundukları yapıya **bilgisayar ağı** denir.

Protokol

- İki bilgisayar arasındaki iletişimini sağlamak amacıyla fiziksel düzeyden uygulama düzeyine kadar birçok protokol tanımlanmıştır.
- Popüler iletişim protokolleri şunlardır:
 - ADSL, ISDN, Ethernet
 - 802.11 WiFi
 - TCP/IP, UDP
 - DNS,
 - HTTP, FTP, DHCP

Bilgisayar Ağlarının Sınıflandırılması

Bilgisayar ağları

1. Kapsadıkları alana,
2. İletişim teknolojilerine,
3. Ağ yapılarına,

Göre sınıflandırılmaktadır.

1- Kapsadıkları Alana Göre Bilgisayarların Sınıflandırılması

- Kapsadıkları alana göre bilgisayar ağlarını genel olarak 3 sınıfa ayırmak mümkündür:
 - a) Yerel Alan Ağları(Local Area Networks-LAN)
 - b) Metropolitan Alan Ağları(Metropolitan Area Networks-MAN)
 - c) GenişAlan Ağları(Wide Area Networks-WAN)

1-a) Yerel Alan Ağları (LAN)

- Yüksek hızlı ve genelde tek bir bina yada yerleske içerisinde (1-1000m arasındaki) kurulan ağları tanımlar.
- Yerel ağ içinde bilgisayarlar, workstation, yazıcılar, çiziciler, CDROM sürücüleri ve diğer çevre birimleri yer alabilir.
- Ağ bağlantısı kablolu veya kablosuz olarak kurulabilir.
- LAN'lar bilgisayar kullanıcılarına uygulamalara ve cihazlara ulaşım, bağlı kullanıcılar arasında dosya değişimi, elektronik posta ve diğer uygulamalar yoluyla haberleşme gibi çeşitli avantajlar sağlarlar.

1-a) Yerel Alan Ağları (LAN)

- LAN'lar bilgisayar kullanıcılarına uygulamalara ve cihazlara ulaşım, bağlı kullanıcılar arasında dosya değişimi, elektronik posta ve diğer uygulamalar yoluyla haberleşme gibi çeşitli avantajlar sağlarlar.
- Bir oda yada bir ev içindeki çeşitli cihazların birbirleri ile yada bir bilgisayar ile haberleşmesini sağlayan küçük çaplı LAN'lara PAN (Personal Area Network) adı verilir.

1-b) Metropol (Kampüs) Alan Ağları (MAN)

- Genellikle şehrin bir kısmını (1-10km) kapsayan yerleşkelerarası veri alışverişini sağlayan ağlardır.
- Mesafeye ve coğrafyaya göre kablolulu yada kablosuz veri transferi seçilebilir.
- Mesafenin etkin olarak kapsamması gereği ve ağa bağlı her bölge arasında tam erişim gerekliliğinden değişik donanım ve aktarım ortamları kullanılır.

1-c) Geniş Alan Ağları (WAN)

- Bir ülke ya da dünya çapında yüzlerce veya binlerce kilometre mesafeler arasında iletişimini sağlayan ağlardır.
- Coğrafi olarak birbirinden uzak yerlerdeki (şehirlerarası/ülkelerarası) bilgisayar sistemlerinin veya yerel bilgisayar ağlarının (LAN) birbirleri ile bağlanmasıyla oluşturulur.
- Genellikle kablo ya da uydular aracılığı ile uzak yerleşimlerle iletişimimin kurulduğu bu ağlarda çok sayıda iş istasyonu kullanılır.

LAN-MAN-WAN Karşılaştırılması

	WAN	MAN	LAN
Coğrafi Büyüklük	1.000'lerce km	1-100 km arası	0-5 km
Düğüm Sayısı	10.000'lerce	1-500	1-200
Bit Hızı	0.1-100 kbits/s	1-100 Mbits/s	1-100 Mbits/s
Gecikme	> 0.5 sn	100-1000 ms	1-100 ms
Yönlendirme	Karmaşık	Basit	Yok
Bağlantı Aygıtı	Yönlendiriciler	Köprüler	Köprüler

2- İletim Teknolojilerine Göre Ağların Sınıflandırılması

- Veri iletimi sırasında kullandığı teknolojiye göre değerlendirildiğinde bilgisayar ağları
 - a) Yayın ağları ve
 - b) Anahtarlamalı ağlarolarak iki grupta toplanmaktadır.

2-a) Yayın ağları (Broadcast Networks)

- Yayın ağlarında tek bir iletişim ortamı ağa bağlı tüm bilgisayarlar tarafından paylaşılır.
- Bir bilgisayarın yaptığı yayın, diğer tüm bilgisayarlar tarafından dinlenir.
- Yayın yapacak olan, çoğu zaman önce ortamı dinler.
- Başka yayın yapan yoksa göndermek istediği bilgiyi paketler halinde iletişim ortamına aktarır.

Yayın ağları...

- Her bir pakette, gönderilmesi hedeflenen bilgisayar ya da bilgisayarlar adresleri vardır.
- İlgili bilgisayarlar, iletişim ortamından kendilerine gelen paketi alırlarken, diğer bilgisayarlar paketin adres kısmında olmadıklarını gördükten sonra paket için başka bir işlem yapmazlar.

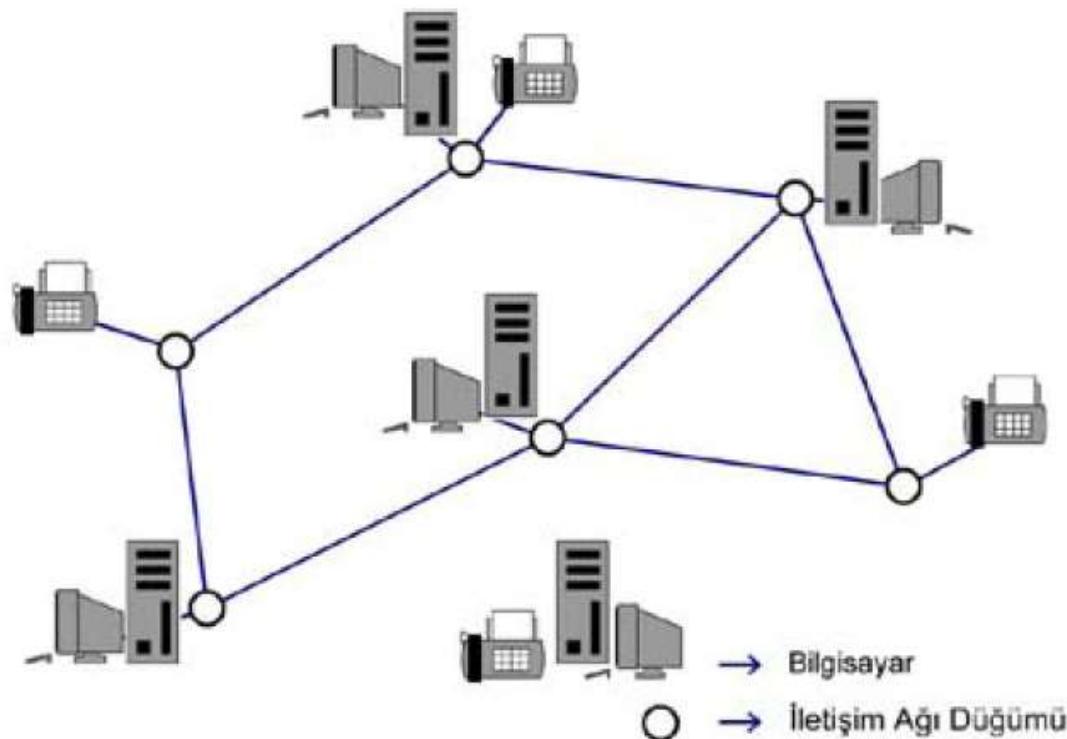
Yayın ağları...

- Bir paket sadece bir bilgisayara gönderiliyorsa **tekli yayın** (unicasting)
- Birden fazla bilgisayara gönderiliyorsa **çoklu yayın** (multicasting)
- Tüm bilgisayarlara gönderiliyorsa da **genel yayın** (broadcasting)

Olarak adlandırılır.

2-b) Anahtarlamalı Ağlar (Switched Networks)

- Anahtarlamalı ağlar, yayın ağlarından farklı bir prensipte çalışmaktadır.



Anahtarlamalı Ağlar...

- Alıcı ve verici dışındaki diğer düğümler verinin içeriği ile ilgilenmez.
- Amaç, iki nokta arasında veriyi hedefe varana kadar bir düğümden diğerine aktararak taşımaktır.
- Bu ağlarda, kullanılan düğümler arasında bir bağlantı kurulur.

3- Ağ Yapılarına (Topoloji) göre Ağların Sınıflandırılması

- Topoloji,
 - bir ağdaki bilgisayarların nasıl yerleşeceğini,
 - nasıl bağlanacağını,
 - veri iletiminin nasıl olacağını

belirleyen genel yapıdır.

Temel Topoloji Türleri

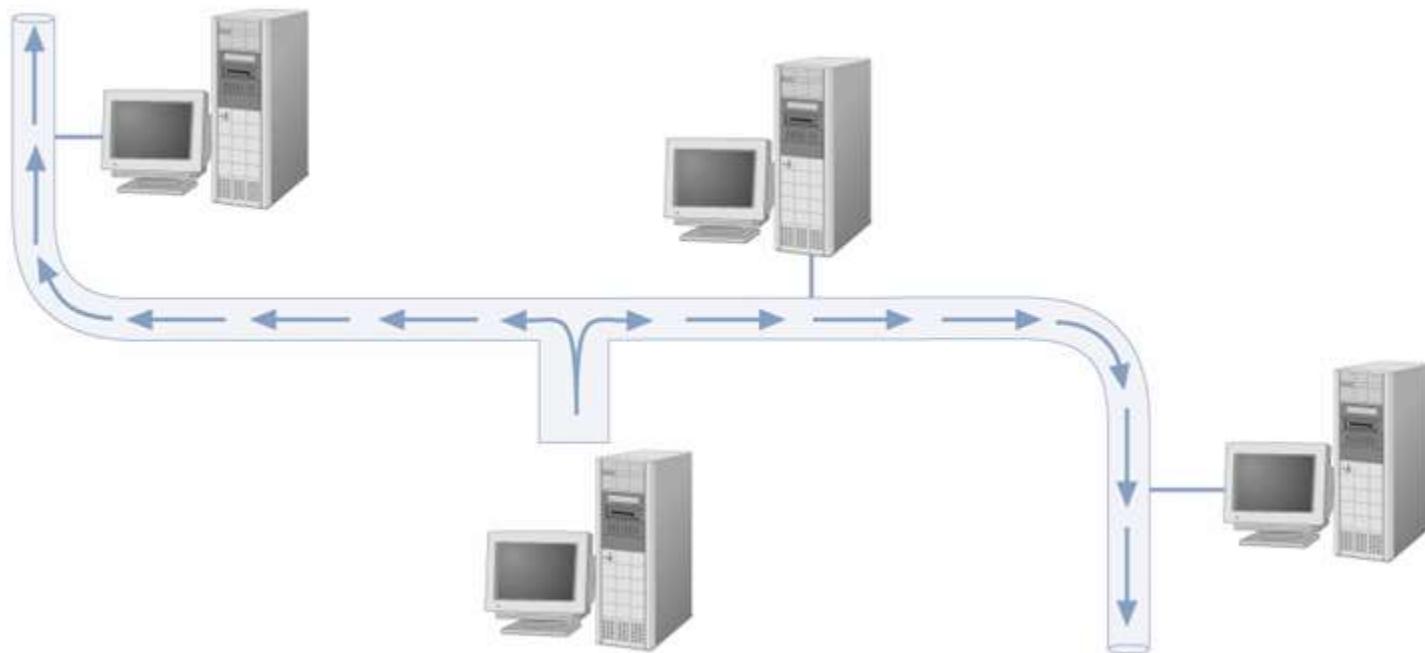
- Doğrusal (Bus Topology)
 - Halka (Ring Topology)
 - Yıldız (Star Topology)
 - Ağaç (Tree Topology)
 - Örgü (Mesh Topology)
- 
- LAN
- WAN

Doğrusal (Bus) Topoloji

- Bus topolojisinde tek bir iletişim ortamı (örneğin bir kablo), düğümlerin birbirleriyle iletişimini sağlar.
- Bus topolojide genellikle Ethernet kartları kullanılmaktadır.
- İstasyonlar (düğümler) veri yoluna musluk (tap) adı verilen bir bağ ile bağlanmışlardır.
- İstasyonlar ve arabirimler ile iletişim ortamı arasında çift-yönlü bir iletişim (full-duplex) vardır.

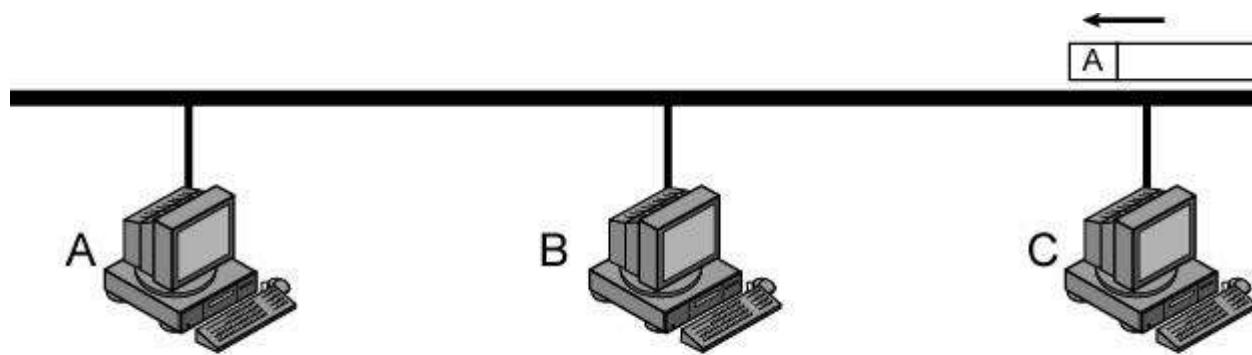
Doğrusal (Bus) Topoloji

- Bir kablo yol olarak düşünülürse, bu yol üzerindeki her bir durak ağda bir düğümü (node-terminali/cihazı) temsil etmektedir.
- Bu tek kabloya; bölüm (segment), omurga (backbone), trunk denilebilir.

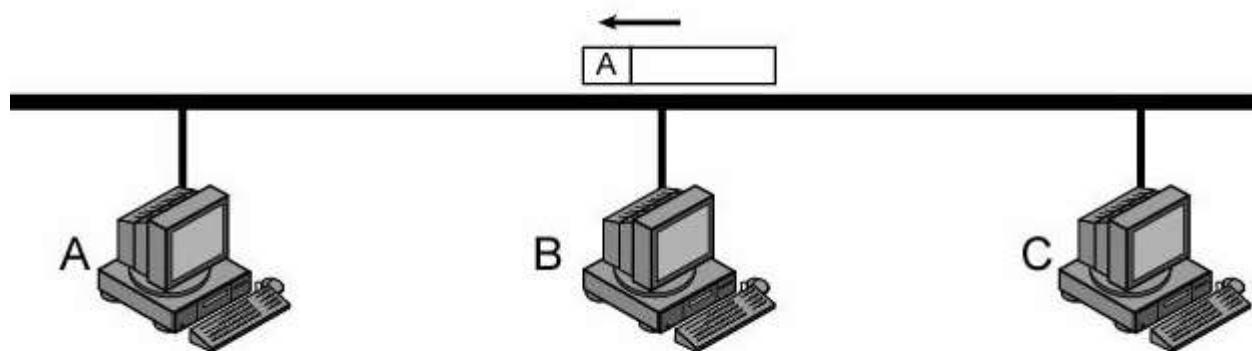


Doğrusal (Bus) Topoloji

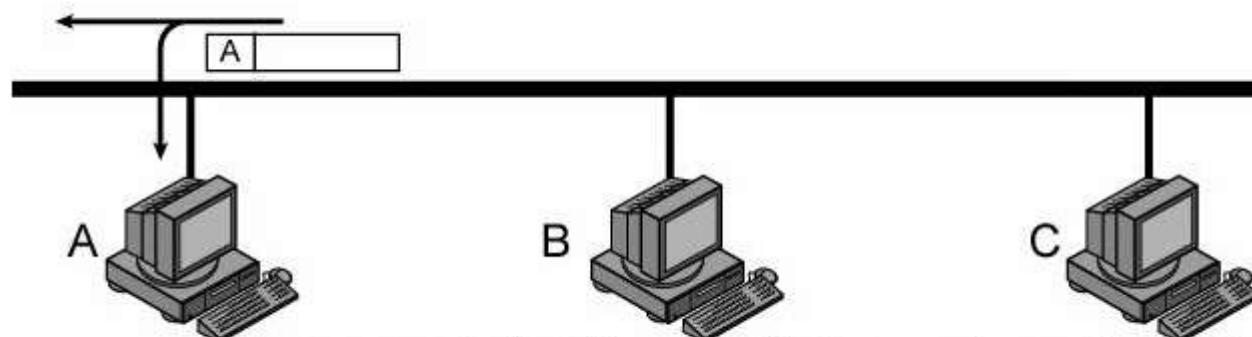
- Veri yolunun başlangıç ve bitisi birbirine bağlı değildir.
- Bu topolojide her düğüme bir adres verilir ve bu yapıdaki bir ağda veri herhangi iki düğüm arasında iletilebilir.
- İletişimde bulunan düğümler veri yolunu iletim süresince işgal eder.
- Bundan dolayı her istasyon mesaj göndermeden önce veri yolunu kontrol ederek herhangi bir mesaj olup olmadığını bakar.
- Aynı iletişim ortamı tüm düğümlerce paylaşıldığı için, mesajlar gönderildiği düğümün adresiyle iletılır.



C bir çerçeveyi A bilgisayarına gönderir



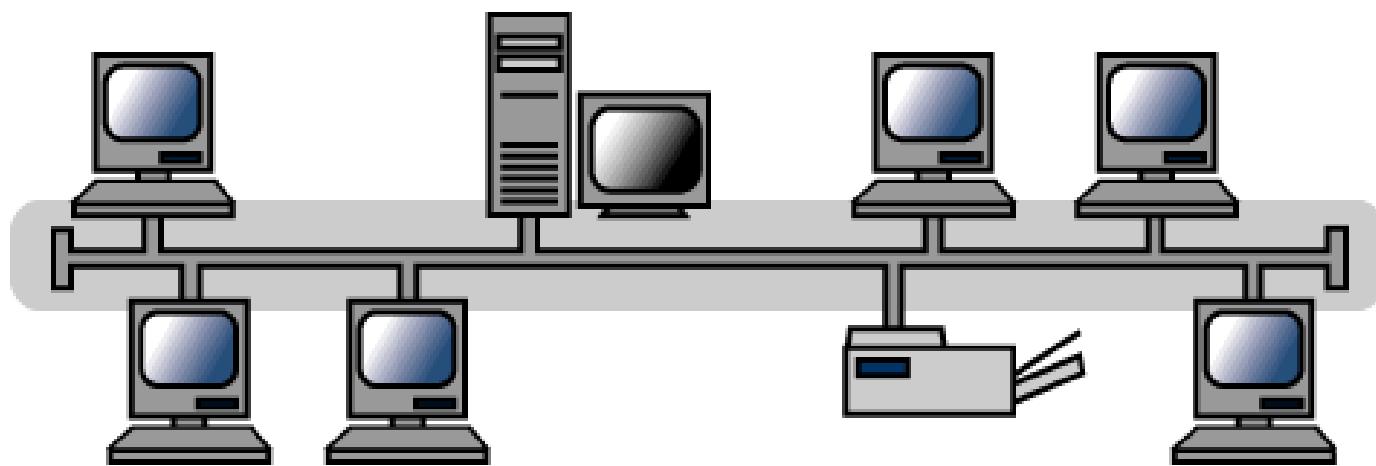
Çerçevenin adresi B olmadığından, B bunu dikkate almaz



A, çerçevenin kendine ait olduğunu anlar ve bunu alır,
çerçeve kablo boyunca yoluna devam eder.

Doğrusal Topolojinin Avantajları

- Ağa bir bilgisayarı bağlamak oldukça kolaydır
- Daha az uzunlukta kablo gerektirir.
- Bir bilgisayarda oluşacak hata tüm ağı etkilemez.
- Merkez birime ihtiyaç duyulmaz.



Doğrusal Topolojinin Dezavantajları

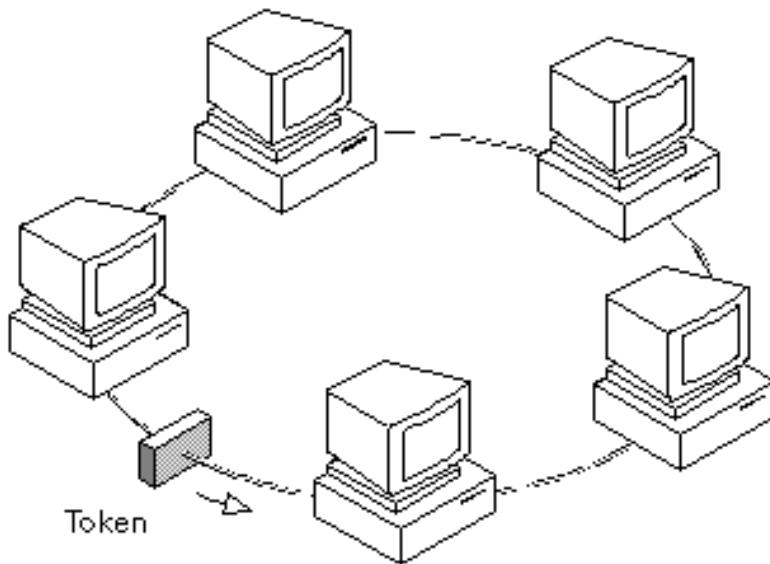
- Omurga kabloda bir bozulma veya kesilme olursa tüm ağ bağlantısı kesilir.
- Kablonun sonunda sonlandırıcı (Terminator) olmalıdır.
- Ağda sorun olduğunda sorunun nerden kaynaklandığını bulmak zaman alıcı olabilir.
- Çarpışma olma olasılığı yüksektir.
- Maksimum 30 istasyon bağlanabilir.
- Ağın uzunluğu 500 metreden fazla olmaz.

Halka (Ring Topology)

- İletişim bağlantısının başlangıç ve bitişleri birbirlerine bağlanmıştır.
- Veriler paket halinde gönderilir ve halka boyunca tek yönde iletilir.
- Halkayı birçok bilgisayar paylaştığından hangi bilgisayarın paketinin halka üzerinde iletileceğinin kontrol etmek amacıyla mekanizmalar geliştirilmiştir.
- Buna örnek olarak jetonlu halka (token ring) verilebilir.

Jetonlu Halka (Token Ring Topology)

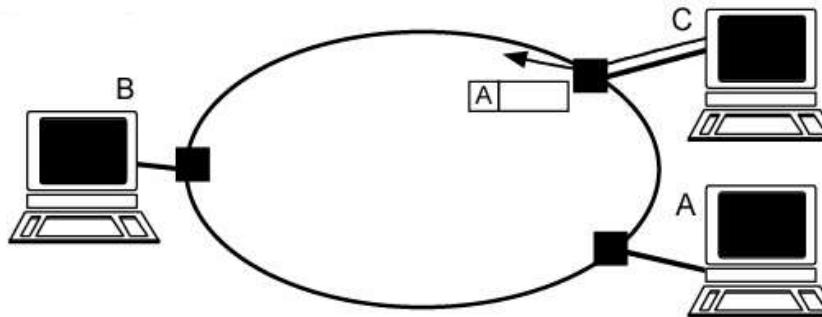
- Bu tip topolojide iletişim ağ içerisinde sürekli dönen jeton (token) yardımıyla yapılır.
- Jeton özel iletişim kodu ile iletişimi düzenler.
- Token (Jeton) düğümler arasında dolaşan bilgidir.



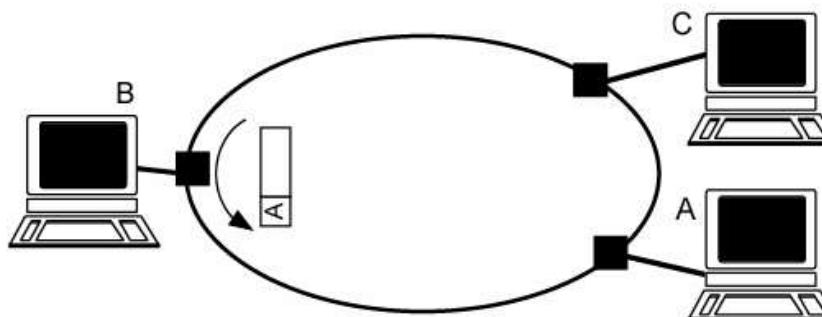
Jetonlu Halka (Token Ring Topology)

- İletişime başlamak isteyen düğüm öncelikle jeton'un kendisine ulaşmasını bekler ve ulaştığında jeton'u alır.
- Artık jeton serbest dolaşımdan kullanıma geçmiş olur. Bilgi gönderildikten sonra alıcıya gelene kadar halka etrafında dolaşır.

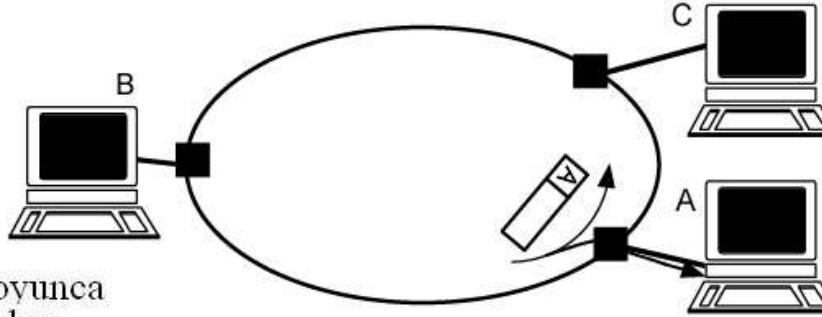
(a) C bir çerçeveyi
A bilgisayarına
gönderir



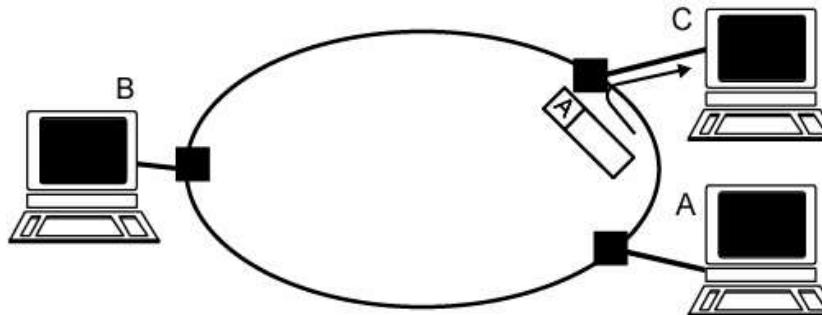
(b) Çerçevenin adresi
B olmadığından,
B bunu
dikkate almaz



(c) A, çerçeveyin
kendine ait
olduğunu anlar
ve bunu alır,
çerçeve kablo boyunca
yoluna devam eder.



(d) C geri dönen
çerçeveyi alır.



Halka (Ring Topology)

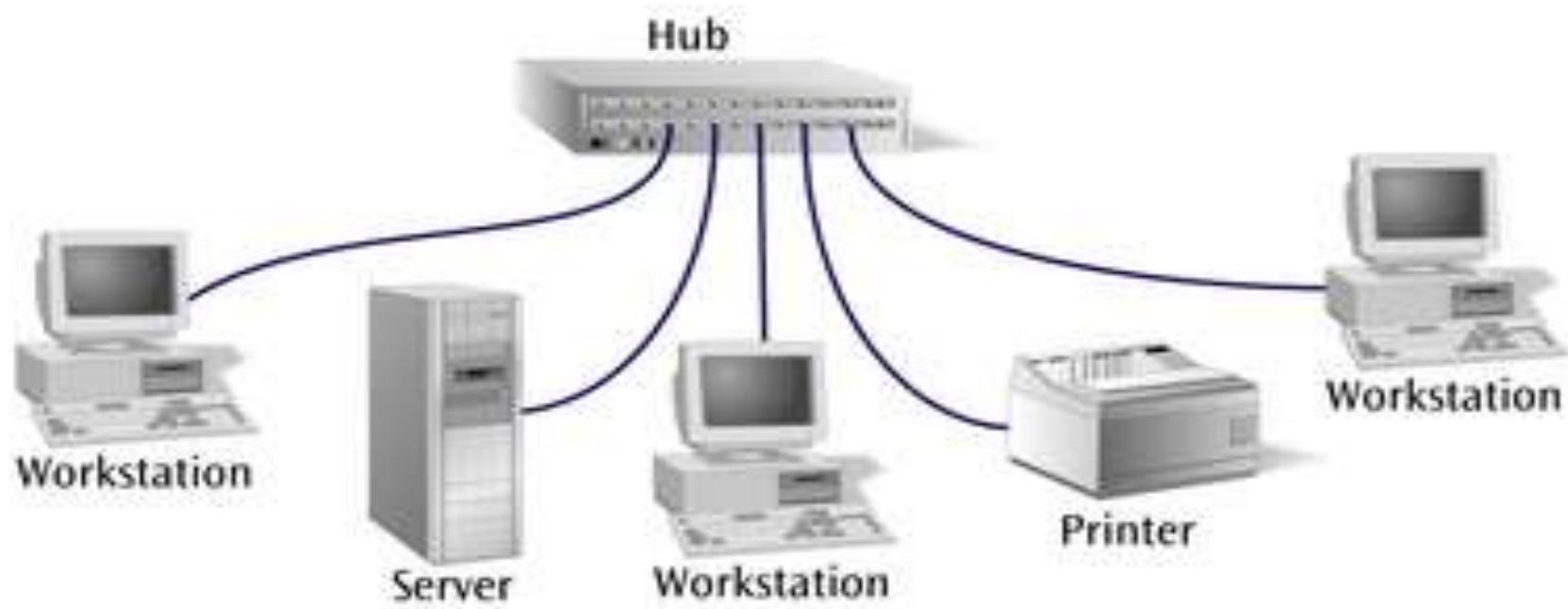
- Bu tür topolojilerin tek yönlü ve çift yönlü olmak üzere iki çeşidi vardır.
- Tek yönlü ağlarda mesaj, düğümler arasında tek yönde gönderilir.
- Bu hatta doğacak herhangi bir problem iletişim durmasına yol açar.
- Çift yönlü halka ağlarda, halka çift yönlü iletişimini destekler.
- Bir yönde hata olması durumunda, diğer yön kullanılabilir.
- Her ağ düğümünün her mesajı iletmesi bu topolojinin potansiyel zayıflığı olmakla beraber, kavram ve gerçekleştirim açısından oldukça basittir.
- Halka topolojisi optik lifli Yerel Alan ağlarında sıkça kullanılır.

Halka Topolojinin Avantaj ve Dezavantajları

- Bekleme süresi devredeki eleman sayısına bağlıdır.
- Sistemin hızı devreye eklenen her elemanla biraz daha azalır.
- Her PC bilgisayar ağının bir elemanı olduğu için PC’de bir aksaklı olması halinde network durur. Paralel bir ikinci hattın (by pass hattı) çekilmesiyle bu soruna çözüm bulmak mümkündür.
- Maliyet bakımından diğer ağlardan biraz daha pahallıdır.
- İletişim hızları kablolama sisteminize bağlıdır.

Yıldız (Star Topology)

- Tüm düğümlerin ortak bir merkeze (örneğin, hub, switch) bağlanmasıdır.



Yıldız (Star Topology)

- Ağı oluşturan bilgisayarlar ana makinaya noktadan noktaya bir bağlantı sağlarlar.
- Merkezi bilgisayar ağ düğümleri arasındaki veri iletişimini koordine eder.
- Tüm iletişim önce merkezi bilgisayara gider, merkezi bilgisayar işlemleri ve bilgi paylaşımını kontrol eder.

Yıldız Topolojinin Avantajları

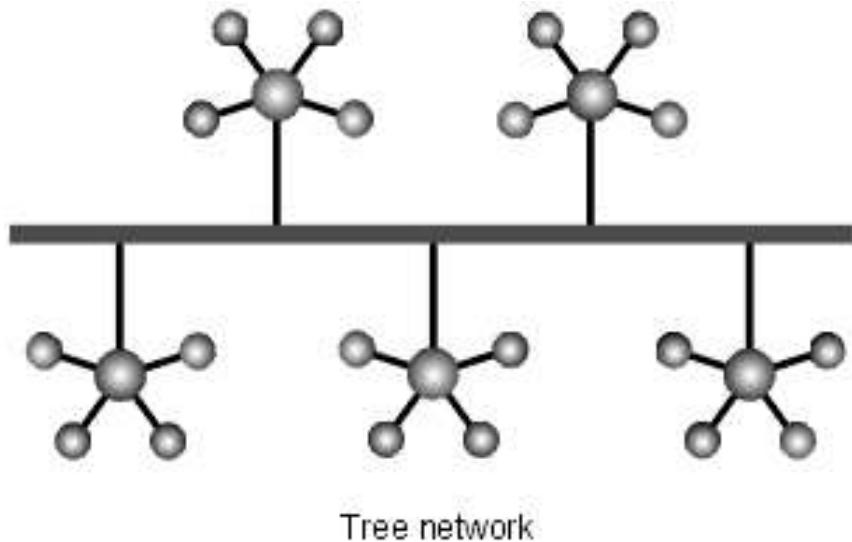
- Ağ kurmak kolaydır
- Bir bilgisayara bağlı kablo bozulduğunda ağın çalışması etkilenmez.
- Ağdaki sorunları tespit etmek kolaydır.
- Kurulum sırasında maliyet yüksek, daha sonra genişletilmesi daha ekonomiktir.
- İletişim ortamı olarak telefon hatlarından yararlanılabilir.

Yıldız Topolojinin Dezavantajları

- Ağa bağlanan her cihaz için bir kablo çekilmesi gereklidir.
- Server'a yada hub'a bir şey olduğunda tüm ağ çalışmaz hale gelir.
- Ağın genişletilmesi server'ın yada hub'ın kapasitesine bağlıdır.
- Sistem performansı da ana makine yada hub'ın veri yolu kapasitesine bağlıdır.

Ağaç (Tree Topology)

- Ağaç topolojisi Bus ve yıldız topolojilerinin karakteristiklerini birleştirir.
- Bus omurga üzerinde yıldız topolojide bilgisayarlardan oluşur.
- Böylece ağlar büyütülebilir.
- Bir ağacın dalları farklı topolojilerdeki ağları temsil eder, ağacın gövdesi ile de bunlar birbirine bağlanabilir.



Ağaç Topolojisinin Avantaj ve Dezavantajları

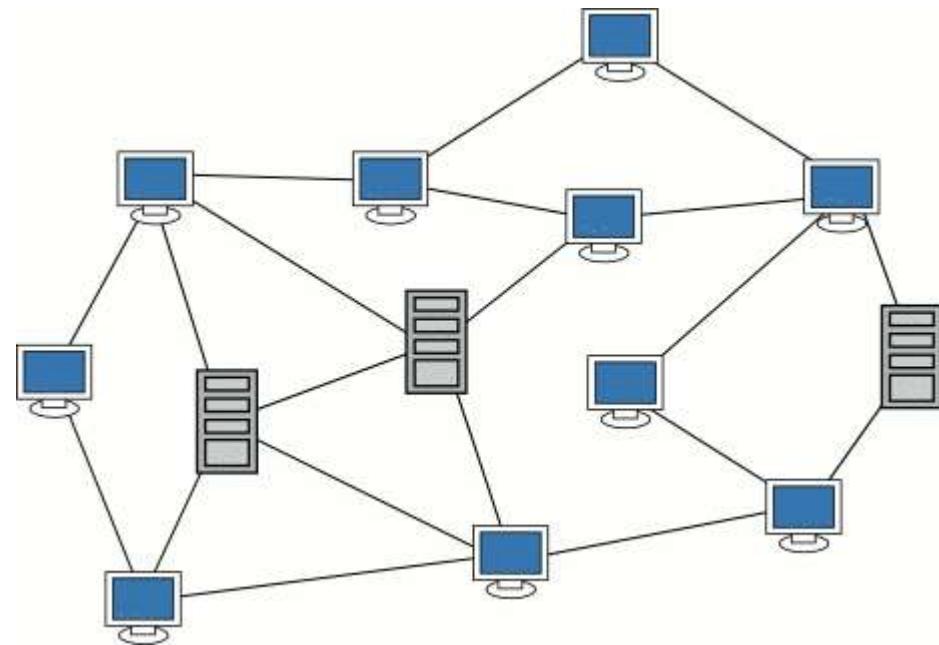
- Avantajları:
 - Her bir bölüme (segment) ulaşmak kolaydır
 - Bir çok çalışma grubu bir araya getirilebilir.
- Dezavantajları:
 - Her bir bölümün uzunluğu kullanılan kablo ile sınırlıdır.
 - Omurga kablosu bozulduğunda bölgelerdeki ağ trafiği etkilenir.
 - Kurulumu ve düzenlenmesi daha zordur.

Örgü (Mesh Topology)

- Fiziksel mesh topolojisi ağdaki tüm birimler arasında uçtan uca bağlantı içerir.
- Ağdaki her birim diğer tüm birimler için birer bağlantı gerektirdiğinden, genellikle pratik bulunmaz.
- Daha çok WAN'da kullanılır.

Örgü (Mesh Topology)

- Tipik olarak mesh topolojisi en geniş ya da en önemli yerlerin bağlandığı hibrid ağlarda kullanılır.
 - Örneğin bir kuruluşun 4 veya 5 ana merkezi ile çok sayıda uzak ofisi olduğunu varsayıalım.



İstemci/Sunucu Mimarisi

- Sunucu (iş istasyonu yada bilgisayar):
 - Pasif durumdadır
 - İstekleri (request) bekler
 - İstek olduğunda bilgiyi hazırlar ve cevap yollar
- İstemci (bilgisayar yada mobil cihaz):
 - Aktif durumdadır
 - İstekleri gönderir
 - Cevap dönene kadar bekler

Bant Genişliği (İletim Hızı)

- 1 saniyede transfer edilen veri miktarını gösteren ölçü birimidir.
- $1 \text{ Kbit/s (Kbps)} = 1.000 \text{ bit/s} (\neq 1.024 \text{ bit/s})$
- $1 \text{ Mbit/s} = 1.000.000 \text{ bit/s} = 125.000 \text{ byte/s}$
- ‘B’ ile ‘b’ aynı değildir:
 - Kbps : Kilobit per second
 - KBps : KiloByte per second
- 1000 – 1024 karmaşıklığını gidermek için Kibi, Mebi, Gibi, ... kullanımları tavsiye edilmektedir. (Bak. http://en.wikipedia.org/wiki/Data_rate_units)

Ağ İletişimi Referans Modeli (Başvuru Modelleri)

- Ağlarda donanım bileşenlerinin birbirleriyle haberleşebilmeleri için birbirleri üzerine inşa edilmiş, değişik seviyelerde çalışan (bilgisayara en yakından, kullanıcıya en yakınına olmak üzere) protokoller kullanılır.
- Bir ağdaki haberleşme standartlarını belirleyen bu protokol katmanları ile ağır katmansal görüntüsü ('layered view') elde edilir.
- Bu katmansal mimariler, ağlardaki standart farklılaşmalarını engellemek temel amacıyla geliştirilmiştir.

Referans Model...

- Katmansal yapının iki temel avantajı bulunmaktadır.
 - Farklı katmanlardaki protokoller nerdeyse birbirlerinden bağımsız olarak geliştirilebileceklerinden ağ tasarım işi büyük ölçüde kolaylaşmaktadır.
 - Ayrıca katmanların birbirlerinden bağımsız olması sonucunda uyumsuzluk problemi ortadan kalkmaktadır. Bir kurum tarafından hazırlanmış katmanlar başka kurumlarda da bir değişiklik gerekmeksiz kullanılabilecektir.

OSI (Open Systems Interconnection) Referans Modeli

- Bilgisayar ağlarının ilk günlerinde farklı firmalar kendilerine özel teknolojilerle ağ sistemleri geliştiriyorlar ve satıyorlardı.
- Kendi başlarına düzgün çalışan bu ağlar ortak çalışma yeteneğine sahip değildi.
- Her birinin kendine özel yazılım ve donanımları vardı.
- Farklı isimlendirme sistemleri ve sürücüler kullanan bu ağları birbirleriyle iletişime geçirmek imkânsızdı.
- Ağ sistemlerinin bu özel yapısı diğer donanım ve yazılım üreticilerinin bu ağlar için ürün geliştirmesini de imkânsız hale getiriyordu.

OSI Referans Modeli

- Ağ sistemlerine olan talebin artması ile ağ sistemlerinin işlevlerini tanımlayan ortak bir model oluşturulması gerektiği anlaşıldı.
- Bunu gerekli kıلان bir diğer unsur ise ağ sistemlerini açıklamakta kullanılan terimlerin üreticiden üreticiye değişiklik göstermesi, ağ üzerinde işlem gören yazılım ve donanım bileşenlerinin ne görev üstlendiklerinin standart halinde olmamasıydı.

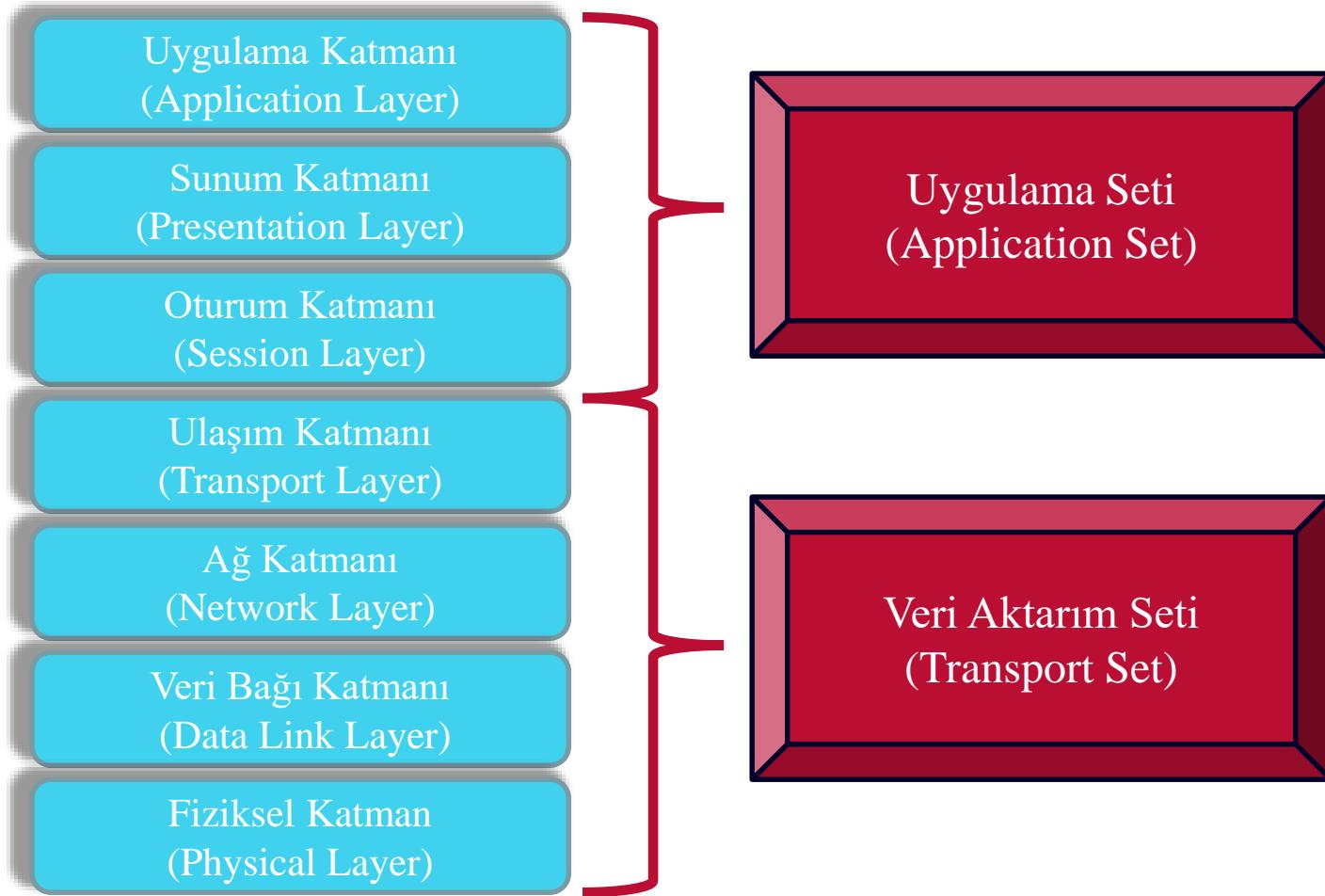
OSI Referans Modeli

- 1980'li yılların başında Uluslararası Standartlar Organizasyonu (International Standards Organization-ISO) bilgisayar sistemlerinin birbirleri ile olan iletişiminde ortak bir yapıya ulaşmak yönünde çabaları sonuca bağlamak için bir çalışma başlatmıştır.
- Bu çalışmalar sonucunda 1984 yılında Open Systems Interconnection - OSI referans modeli ortaya çıkmıştır.

OSI Referans Modeli

- OSI kavramsal bir modeldir.
- OSI modeli verinin bir bilgisayar üzerinde bir program'dan, ağ ortamından geçerek diğer bir bilgisayar üzerindeki diğer bir programa nasıl ulaşacağını tanımlar.
- OSI referans modelinde, iki bilgisayar sistemi arasında yapılacak olan iletişim problemini çözmek için 7 katmanlı bir ağ sistemi önerilmiştir.
- OSI Modelinde her katman çözülmesi gereken problemleri tanımlar.
- Bu katmanda çalışan aygıt ve protokoller ise bu problemlere çözüm getirir.

OSI Referans Modeli



OSI Referans Modeli

- 7 katmanlı OSI modeli 2 bölümde incelenebilir.
 - **Application Set (uygulama seti):**
 - Uygulamalar yani programlarla ilgili konuları içerir.
 - Genellikle sadece yazılımsaldır.
 - Modelin en üstündeki uygulama katmanı kullanıcıya en yakın katmandır.
 - **Transport Set (veri aktarım seti):**
 - Veri iletişimini ile ilgili meseleleri tanımlar.
 - Fiziksel ve veri aktarım katmanları hem yazılımsal hem de donanım olarak görevini yerine getirebilir.
 - Fiziksel katman (en alt katman) fiziksel ağ ortamına (ağ kablosuna mesela) en yakın katmandır. Ve esas olarak bilgiyi kablodan aktarmakla görevlidir.

1-) Fiziksel Katman

- Verinin fiziksel (bakır tel, optic lif, hava...) ortamda taşınması için gerekli yapıyı, kodlamayı oluşturur.
- Diğer katmanlar 1 ve 0 değerleriyle çalışırken, fiziksel katman 1 ve 0'ların nasıl elektrik, ışık veya radyo sinyallerine çevrileceğini ve aktarılacağını tanımlar.
- Gönderen tarafta fiziksel katman bir ve sıfırları elektrik sinyallerine çevirip kabloya yerleştirirken, alıcı tarafta fiziksel katman kablodan okuduğu bu sinyalleri tekrar bir ve sıfır haline getirir.

2-) Veri Bağı (Veri İletimi) Katmanı

- Ağ katmanından aldığı veri paketlerine hata kontrol bitlerini ekleyerek çerçeve (frame) halinde fiziksel katmana iletme işinden sorumludur.
- Bu katmanda veri çerçeve adı verilen bloklara bölünür.
- Hat üzerinden aktarım sırasında oluşan hataların sezilmesi bu katmanın görevidir. Bir hata bulduğunda düzeltir ya da verinin tekrar gönderilmesini ister.
- Hattın iki ucundaki birimin aynı hızlarda çalışmasını ayarlamak da bu katmanın görevidir.

3-) Ağ Katmanı

- Bu katmanda taşınan veri paket adını alır.
- Verinin kaynaktan varışa ulaşması için takip edeceği yolun bulunması bu katmanın görevidir.
- Veri aktarımı sırasında bazı düğümler (yönlendiriciler) üzerinde tıkanıklıklar olabilir.
- Bunların sezilmesi ve gerekli önlemlerin alınması da ağ katmanının görevidir.
- Bunları servis kalitesini (Quality of Service, QoS) arttıracı görevler olarak da adlandırabiliriz.

4-) Ulaşım (Taşıma) Katmanı

- Bu katman, kaynak tarafında, oturum katmanından aldığı veriyi segmentlere böler.
- Varış tarafında ise gelen paketleri birleştirerek oturum katmanına iletir.
- Bölünen verinin numaralandırılması ve varış noktasında karışmış paketlerin tekrar sıralanması, yolda veri üzerinde oluşmuş hatalarla ilgili işlemlerin yapılması bu katmanın görevidir.

5-) Oturum Katmanı

- Bu katman yardımı ile farklı bilgisayarlardaki kullanıcılar arasında oturumlar kurulması sağlanır.
- Bu işlem oturumların kurulmasını, yönetilmesini ve bitirilmesini içerir.
- Bu katman ayrıca iletişim kurallarını belirler. (half-duplex, full-duplex vb.)
- İletişimin kopması durumunda oturumun devam etmesi için eşzamanlama (synchronization) bilgileri tutulur.

6-)Sunum Katmanı

- Bu katmanda iletilecek verinin yapısı belirtilir.
- Uygulamaya bağlı olarak verinin sıkıştırılması/açılması, şifrelenmesi/çözülmESİ yine bu katmanın görevleri arasındadır.

7-) Uygulama Katmanı

- Kullanıcıya en yakın ve en üstteki katmandır.
- Uygulama katmanı programların ağı kullanabilmesi için araçlar sunar.
- Kullanıcının etkileşimde bulunduğu uygulama programlarını destekleyen protokoller bu katmanda yer alır.
- Telnet, SMTP, FTP, SNMP, NCP, SMB, HTTP, Browserlar..

Katmanlar ve Görevleri

Katman	Görevi
7.) Uygulama (Application)	Kullanıcının uygulamaları
6.) Sunum (Presentation)	Aynı dilin konuşulması; veri formatlama, şifreleme
5.) Oturum (Session)	Bağlantının kurulması ve yönetilmesi
4.) Taşıma – Ulaşım (Transport)	Verinin böümlere ayrılarak karşı tarafa gitmesinin kontrol edilmesi
3.) Ağ (Network)	Veri böümlerinin paketlere ayrılması, ağ adreslerinin fiziksel adreslere çevrimi
2.) Veri İletim (Data Link)	Ağ paketlerinin çerçevelere ayrılması
1.) Fiziksel (Physical)	Fiziksel veri aktarımı

TCP/IP Referans Modeli

- ISO/OSI modelinin yaygın olarak tanınmasına karşın, internetin teknik açıdan standartı TCP/IP modelidir (Transmission Control Protocol / Internet Protocol).
- Bu modelin temelini ABD Savunma Bölümü tarafından desteklenerek geliştirilen ARPANET oluşturur.
- TCP/IP referans modeli OSI'den daha önce uygulanmaya başlanmış bir modeldir.
- TCP/IP modeli de tipki OSI modelinde olduğu gibi katmanlı bir yapıya sahiptir. Modelin 4 katmanı vardır.

TCP/IP Referans Modeli

Uygulama Katmanı
(Application Layer)

Taşıma Katmanı
(Transport Layer)

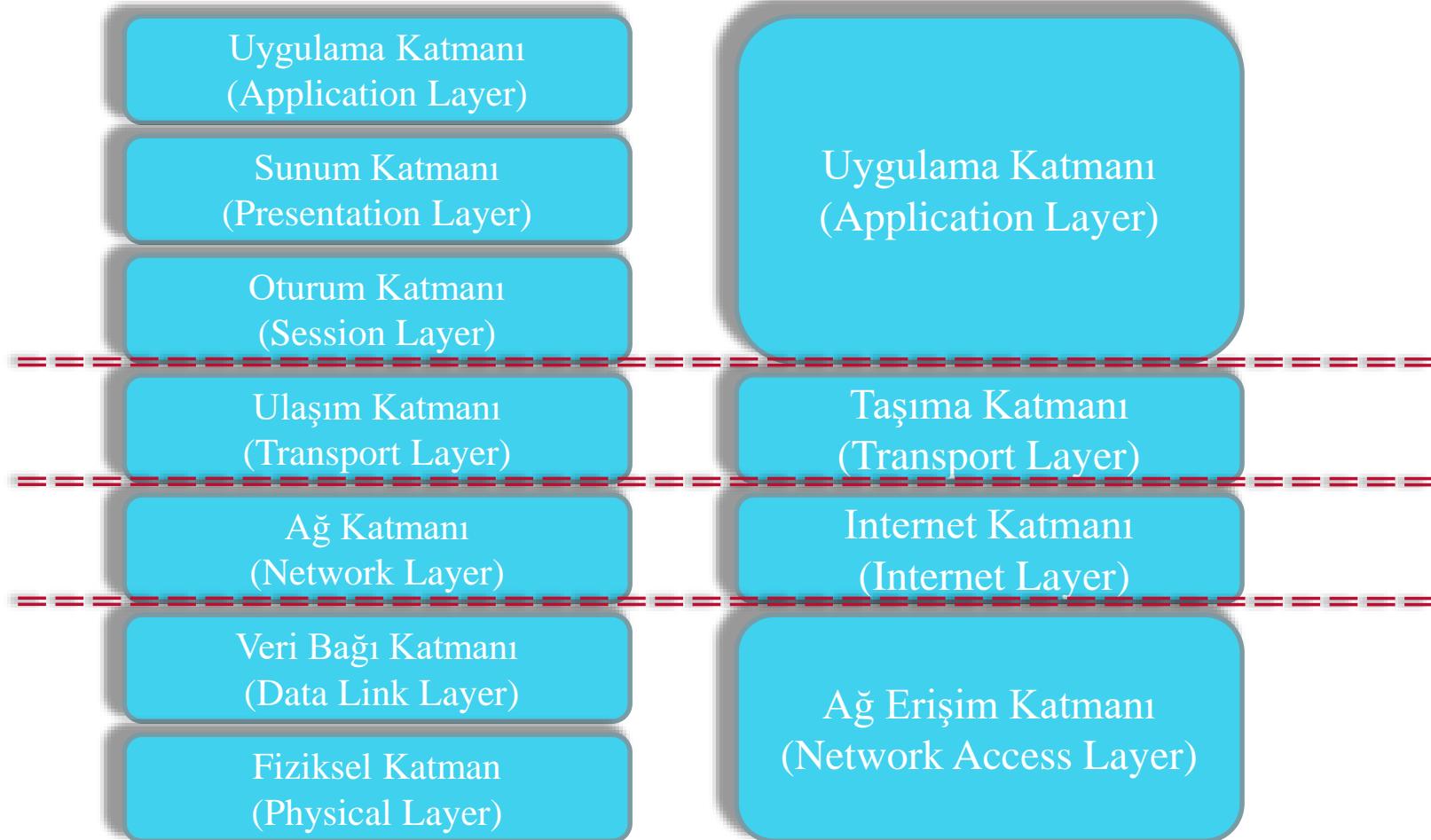
Internet Katmanı
(Internet Layer)

Ağ Erişim Katmanı
(Network Access Layer)

TCP/IP Referans Modeli – Uygulama Katmanı

- ISO/OSI modelinin en üst 3 katmanı ve tüm uygulama-iliskili görevleri TCP/IP modelinde tek bir katmanda birleştirilmiştir.
- Böylece sunum, kodlama ve dialog kontrolü işlerinin yürütüldüğü tek bir uygulama katmanıaratılmıştır.

OSI ve TCP/IP Katmanları



AĞ CİHAZLARI

- Bilgisayar veya benzeri sayısal sistemlerin birbirleriyle karşılıklı çalışmaları, iletişim yapmalarını sağlayan ara cihazlardır.
- Ağ cihazlarının en basiti ağ kartı ve HUB'dır. Bu iki cihaz OSI modelinde ilk iki katmandaki işleri yerine getirir.

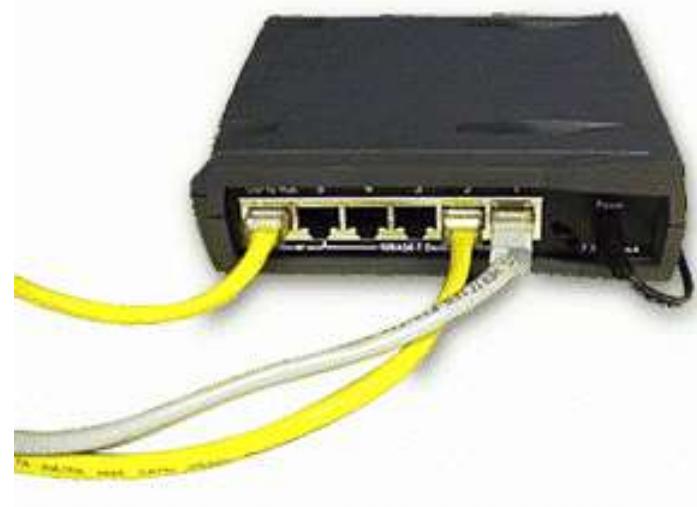
Ağ kartı

- Bir bilgisayarı başka bir bilgisayara yada bir ağ cihazına bağlamak için kullanılan ağ kartları genellikle ethernet protokolünü kullandıkları için ethernet kartı olarak da bilinirler.



Hub

- Hub veriyi sadece alıcıya göndermez, kendisine bağlı olan bütün bilgisayarlara gönderir. Bilgisayarlar verinin kendilerine gönderilip gönderilmединi tespit eder.
- Hub aynı anda sadece 1 iletim yapabilir. Bağlı olan diğer bilgisayarlar iletim için beklemek zorundadır.



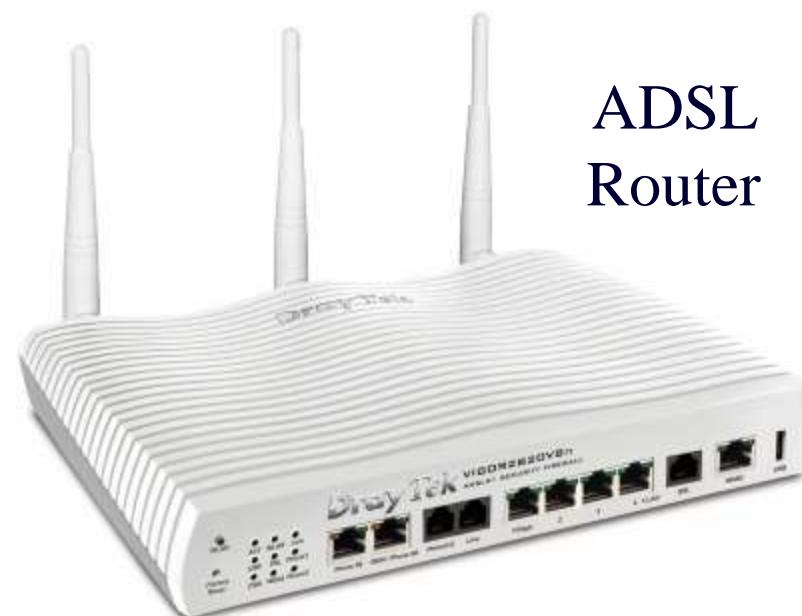
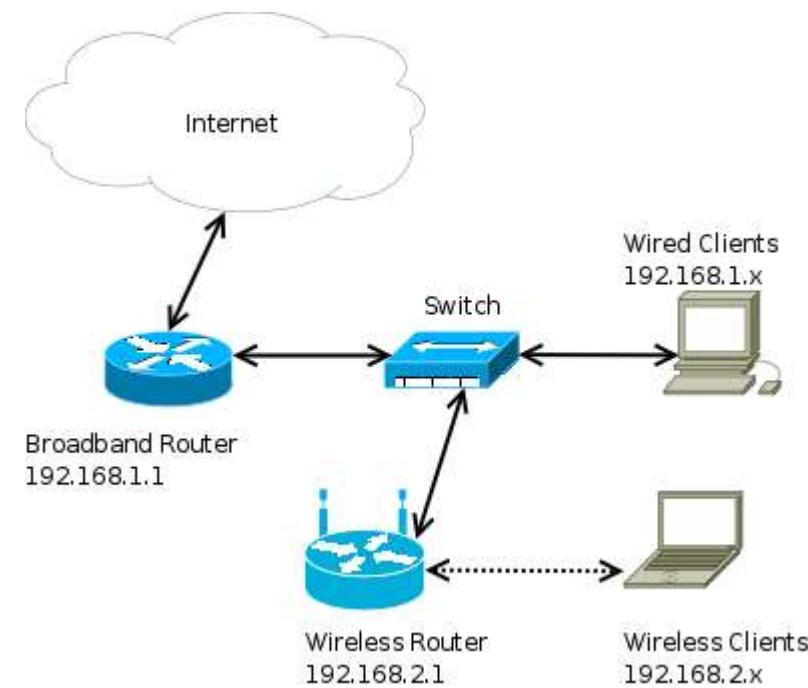
Switch (Anahtar)

- Hub cihazının gelişmişidir.
- Kendisine bağlı cihazlara ortak bir yol değil anahtarlamalı bir yol sunar.
- Dolayısıyla aynı anda birden çok iletişim yapılması olanağı vardır.
Bilgisayar sayısı arttıkça ağ trafiği de artar ve hub yerine switch kullanımı tercih edilir.



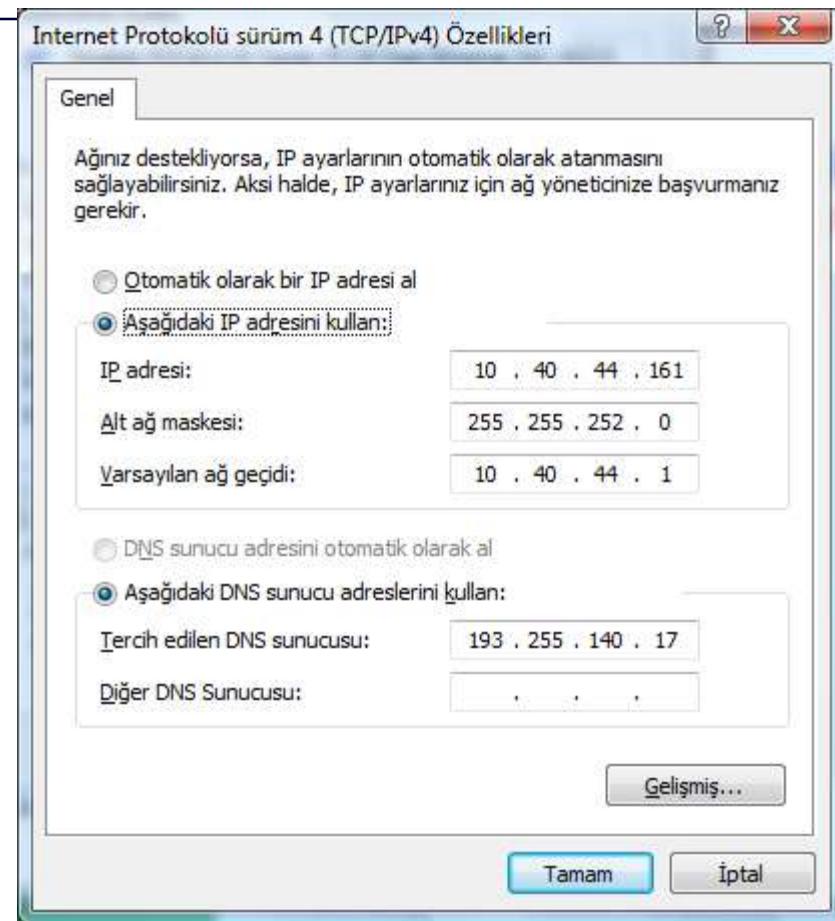
Router (Yönlendirici)

- Genel olarak LAN-WAN ve LAN-LAN bağlantılarında kullanılır. Üzerinde LAN ve WAN için ayrı portlar bulunur.



Ağ Geçidi (Gateway)

- **Ağ Geçitleri** farklı protokollerini kullanan ağların birbiri ile iletişimini sağlar.
- Genellikle bir LAN üzerinden Internet'e çıkmak için router cihazının IP adresi geçit olarak belirlenir.



Modem (MOdulator/DEModulator)

- Analog hat (telefon hattı gibi) üzerinden sayısal veri gönderimini sağlar.
- Dial-up modemler en fazla 56 Kbit/s hızında indirme (download) yapabilirken, ADSL2 modemler ise 25 Mbit/s hızına kadar çıkabilmektedir.



İletim Ortamı (Transmission Media)

- Verilerin bir noktadan diğer bir noktaya iletilmeleri için kullanılan ortamdır.
- Kablo kullanan iletişim ortamları **kılavuzlu** iletişim ortamı, kablosuz iletişim ortamları ise **kılavuzsuz** olarak adlandırılır.



Çift bükümlü (twisted-pair) Kablolar

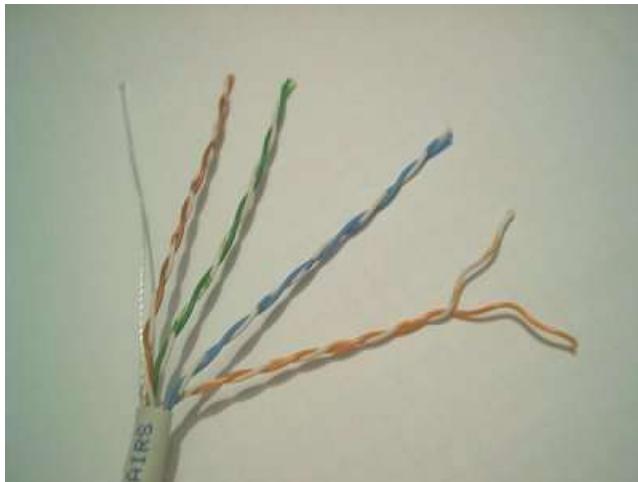
- Bir bakır kablo çifti sinyal taşımada kullanıldığı zaman bir anten gibi etrafında manyetik alan oluşturmaktadır ve komşu kablolarla etkileşime neden olmaktadır.
- Bakır kablolar ikili ikili bükülürse bu manyetik etki en aza indirgenmektedir.
- Yüksek hızlarda veri taşımak için yalıtılmış bakır kabloların çifter çiftter birbirleriyle bükülerek çift bükümlü kablolar elde edilir.
- Kablolar ne kadar sıkı bükülürse manyetik alanının etkisi o kadar azaltılmaktadır.



Çift bükümlü (twisted-pair) Kablolar

- Kısa mesafelerde koruyucusuz (unshielded twisted pair, UTP) olarak kullanılabileceği gibi mesafenin uzaması halinde çevreden gelebilecek elektromanyetik etkiyi engellemek amacıyla bir koruyucu katman içine de yerleştirilebilir (shielded twisted pair, STP).
 - Kaplamalı kablolarla tel çiftlerini örten bir metal bir koruma ve bunun üzerinde plastik bir kılıf bulunmaktadır.
 - Kaplamasız da ise sadece plastik kılıf tel çiftlerini bir arada tutar.

Çift bükümlü (twisted-pair) Kablolar



Kaplamasız Bükümlü çift



Kaplamlı Bükümlü çift

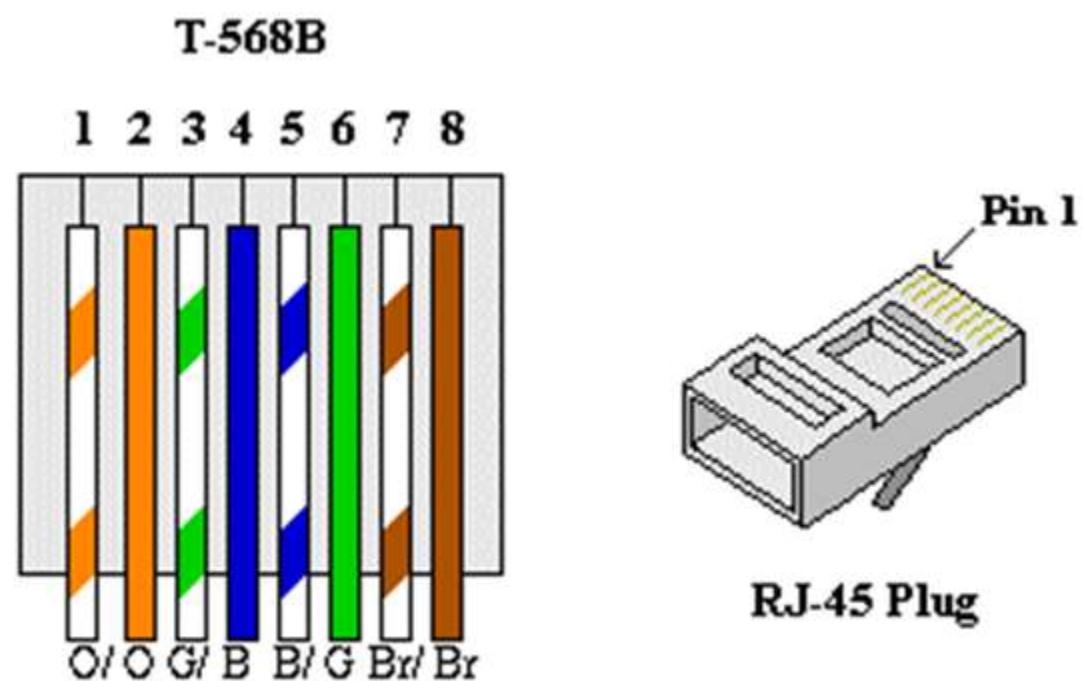
- STP kullanılırken dikkat edilmesi gereken en önemli nokta, dıştaki metal zırh'ın düzgün bir şekilde topraklanmasıdır.
- Aksi halde zırh elektromanyetik dalgaları toplayan bir anten vazifesi görür. Ayrıca zırh'ın kablonun hiçbir noktasında zedelenmemiş olması da çok önemlidir.

Çift bükümlü (twisted-pair) Kablolar

- Telefon hattı için gerekli olan kablo 4 tel içerir ve RJ-11 kodlu jack ile telefon hattına bağlanmaktadır.
- Bilgisayarlarda ise UTP kablo RJ-45 kodlu konnektör ile ağ kartımıza bağlanır ve 8 bakır tele sahiptir.

Bağlantı Türlerine Göre Çift bükümlü (twisted-pair) Kablolar- Düz kablo (Straight-Through Cable)

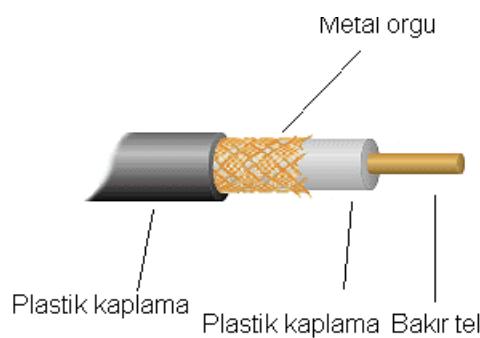
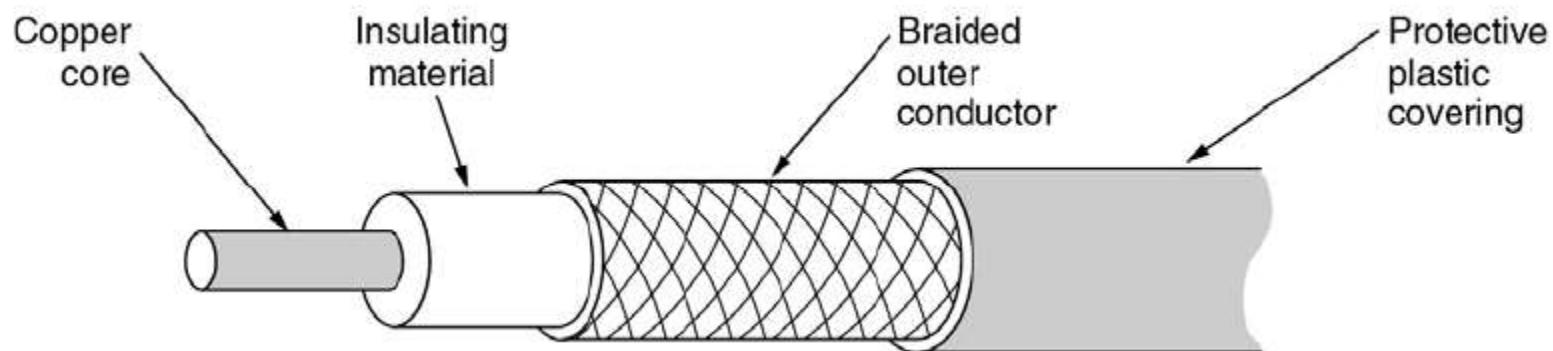
- Düz kablo standartı 568-B'ye göre belirlenmiş renk sırasına göre karşılıklı iki ucun bire bir bağlanmasıdır.
- Renklerin sıralaması şu şekildedir:
 1. Turuncu-beyaz
 2. Turuncu
 3. Yeşil-beyaz
 4. Mavi
 5. Mavi-beyaz
 6. Yeşil
 7. Kahverengi-beyaz
 8. Kahverengi



Koaksiyel (Eş Eksenli) Kablo

- Koaksiyel kablo merkezdeki iletken bir telin etrafı yalıtkan bir katmanla kaplanmış ve onun da etrafına sarılmış iletken olmayan başka bir dış katmandan oluşur.
- Koaksiyel kablo araya girme ve sinyal zayıflamalarına karşı diğer kablolarla oranla daha dayanıklıdır.
- Bu kablo daha uzun mesafelerde twisted-pair kabloya nazaran daha iyidir ve daha az basit teçhizat gerektirip , daha güvenilir ve hızlı bir veri akışı sağlar.
- Ağ uygulamalarına göre farklı koaksiyel kablolar mevcuttur.
- Koaksiyel kabloda bizim için önemli olan ve değişkenlik arzeden değer kablonun empedansı veya ohm değeridir. Bu değer kablonun belirli bir uzunlukta elektrik akımına karşı gösterdiği dirençtir.

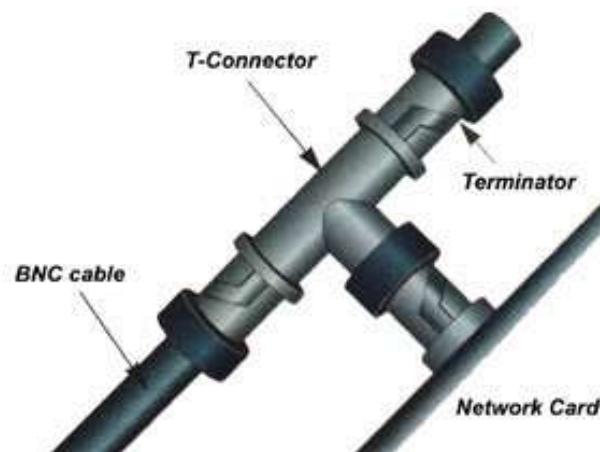
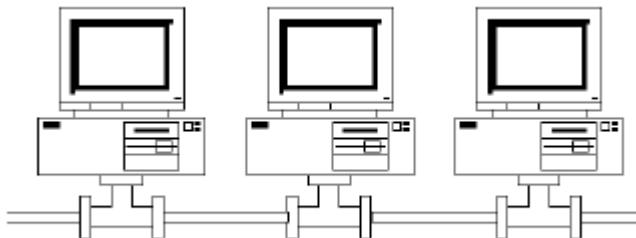
Koaksiyel (Eş Eksenli) Kablo



Koaksiyel Kablonun Yapısı

Koaksiyel Kablo Bağlantıları

- Eşeksenli kablolar BNC konnektörleri ile sonlandırılır ve bilgisayar arkasındaki aktarım aygıtı takılacak T-şeklindeki bağlayıcıılara takılırlar.
- Kablonun iki ucundan biri mutlaka topraklanmalıdır.



Fiber Optik Kablo

- 1966 yılında Charles Kao ve George Hockham cam fiber üzerinden veri aktarımı da yapabileceğinin fikrini ortaya atmışlardır.
- Sonraki dönemlerde fiber üzerindeki kayıp oranları az seviyelere indirilmiştir ve fiber veri aktarımı için bakır'a göre çok daha avantajlı bir konuma gelmiştir.
- Düşük sinyal kayıpları nedeniyle fiber ile bakır kablolarla göre daha yüksek hızlarda ve çok daha uzun mesafelerde veri aktarımı mümkündür.

Fiber Optik Kablo

- Bu mesafe repeater kullanılmadan 2 Km'ye kadar çıkabilir. Bakır UTP kablolarında bu mesafe 100m ile sınırlıdır.
- Fiber'in hafif ve ince yapısı bakır kablo kullanmanın zor olduğu ortamlarda kullanılabilmesini sağlar.
- Bütün bunlar fiber'in önemli özelliklerini olmakla beraber, fiber'in en önemli özelliği elektromanyetik alanlardan hiç etkilenmemesidir.

Fiber Optik Kablo

- Fiber iletken olmadığı için elektriksel yalıtımin zorunlu olduğu yerlerde kullanılabilir.
- Fiber kimyasal fabrikalar, askeri üsler gibi küçük bir elektrik akımının patlamaya neden olabileceği ortamlar için de idealdir.
- Son olarak UTP veya diğer kabloların aksine, fiber bir kablodan bilgi çalmak çok daha zordur.

Fiber Optik Kablo

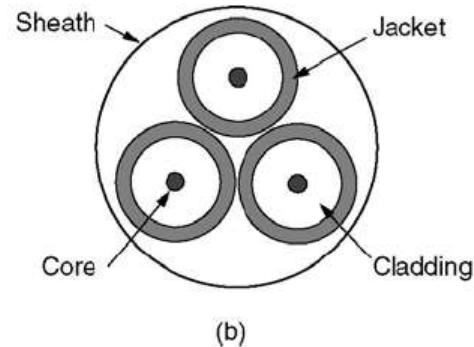
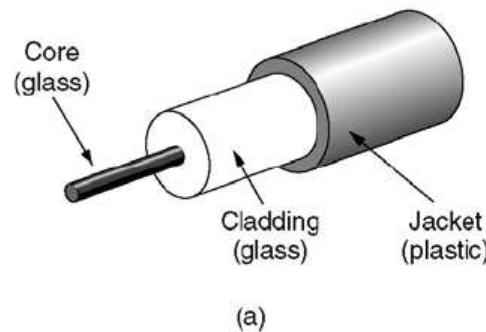
- Optik fiber, sayısal veri sinyallerini ışık vuruşları şeklinde taşır.
- Son derece ince bir cam boru çevresine sarılmış eşmerkezli kaplama (cladding) bir cam katmandan oluşur.
- Veri iletişimini için ışık sinyali kullanıldığından daha zor bozulur ve zayıflar.

Fiber Optik Kablo

- Kablo başına iki fiber kullanılmaktadır.
- Bir tanesi gönderici bir tanesi de alıcıdır.
- Araya girmelere izin vermezler ve son derece hızlıdırlar.
- Fiberoptik ses ve veri iletişimini için ideal bir kablo türüdür.
- Fakat maliyeti yüksek, döşenmesi oldukça zor olmakta ve kolay kırılabilirler.

Fiber Optik Kablo

- ☐ Işık kablonun merkezindeki çok ince cam liften iletılır. Işık herhangi bir tekrarlayıcıya gerek kalmaksızın kilometrelerce gidebilmektedir.
- ☐ Cladding (kaplama) , merkezdeki cam lif içerisinde geçen ışığı geri yansıtmaktadır. Bu ayrıca ışık iletkeni üzerinden akan ışığı dış ışık etkenlerden korur.
- ☐ En dışta ise kılıf (jacket) bulunur ve camı dış fiziksel etkilerden korur.



Fiber Optik Kablolamanın Avantaj ve Dezavantajları

Avantajları

Yüksek hız.

Uzak mesafelerde daha hesaplıdır.

Başka elektriksel dalgalardan etkilenmemektedir.

Network omurgası için uygundur.

Ses, veri ve video aktarım desteği vardır.

Hattın dinlenmesi zordur.

Dezavantajları

Diğer kablolama tiplerine göre daha pahallıdır.

Kablolama yeterince esnek değildir ve keskin bir şekilde bükülemez.

Yüksek trafikte noktadan noktaya bağlanmış olmalıdır.

Kuruluşu destek ve tecrübe gerektirmektedir.

Kablosuz İletim Ortamları

- Kablolama yapılamayacak durumlarda ve mesafelerde Kablosuz iletişim kullanılabilmektedir.
- Veri kablosuz iletişim sistemleri aracılığıyla da serbest uzaydan elektromanyetik dalgalar halinde iletilebilir.
- Elektromanyetik dalgalar, elektronların hareketleriyle oluşur ve serbest uzayda yayılırlar.
- Bir elektrik devresine eklenen uygun büyüklükteki bir anten, elektromanyetik dalgaları yayabilir ve uzaktaki bir alıcı (başka bir anten) tarafından alınmasını sağlayabilir.
- Tüm kablosuz iletişimler bu ilkeye göre çalışmaktadır.

Kablosuz İletim Ortamları

□ Mikrodalga ve RF Teknolojileri

- Mikrodalga Antenler
- Bluetooth
- Hücresel şebekeler

□ Kızıl Ötesi Teknolojisi

- Infrared teknolojisi
- Lazer teknolojisi

WAN Teknolojileri

- WAN uygulamalarında, iki nokta arasındaki iletim yolu, genelde 3. firmaların (Türk Telekom, ISP vb..) sunduğu hizmetlerdir.
- Bağlantının kurulabilmesi için bu iletim yolunun ya kiralanması ya da abonelik yoluyla kullanılması gereklidir.
- WAN teknolojilerinde parametreler;
 - Band genişliği ve bunun optimum değerde kullanılması,
 - Maliyet,
 - Öngörülen hizmet kalitesini sağlaması ve garanti etmesi,

WAN Teknolojileri - Çevirmeli Ağ (dial-up)

- Standart telefon hatlarından fax/modem kartlarıyla yapılan bağlantıdır.
- Internet'e bağlanmak için bir Internet Servis Sağlayıcı şirketinden servis almak gereklidir.
- ISS'nin belirlediği telefon numaraları aranarak bağlantı sağlanır.
- Sağlanan bağlantı tek bilgisayar tarafından kullanılabilir.
- İletim hızı kullanılan modem aygıtının hızına bağlıdır.
- En fazla 56 Kbps hızında iletişim sağlanabilir.
- Kullanımı oldukça azalmıştır.

WAN Teknolojileri - ISDN

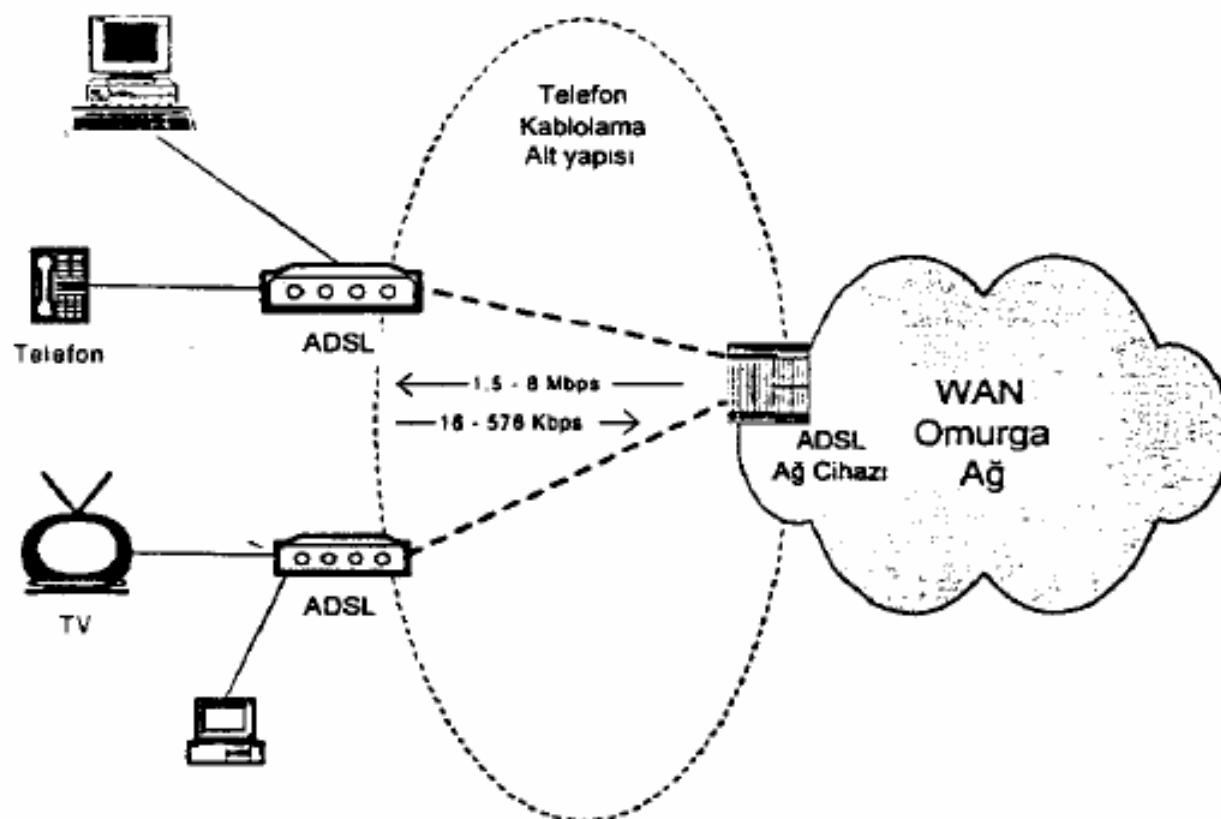
- Integrated Services Digital Network (Tümleşik Hizmetler Sayısal Şebekesi) sözcüklerinin baş harflerinden oluşan ISDN; ses, görüntü, veri gibi her türlü bilginin sayısal bir ortamda birleştirilip aynı hat üzerinden iletilmesinin sağlandığı bir haberleşme ağıdır.
- İletim kalitesi normal telefon hattından daha yüksektir.
- ISDN hata oranı düşük, güvenli ve geniş bir haberleşme imkanı sağlar.
- Bulut teknolojisine dayanır ve sayısal bir iletişim sistemi sunar.

ISDN Hizmetleri

- ISDN; BRI (Basic Rate Interface) ve PRI (Primary Rate Interface) olarak adlandırılan iki tür hizmet sunmaktadır.
- Bu iki hizmet esnek yapısından dolayı WAN bağlantılarında gereksinim duyulan geniş bir istege çözüm bulabilmektedir.
- Genel olarak büyük LAN'ların ISDN'e bağlantısında PRI, küçük ofis yada ev kullanıcılarının bağlantısında ise BRI kullanılmaktadır.

ADSL

- ADSL, kullanıcılara bakır telefon hattı üzerinden konuşmanın yanı sıra yüksek hızlarda asimetrik veri haberleşmesi ortamı sağlayan bir teknolojidir.
- Ortamın alış yönündeki hızı 1.5 Mbps'ten başlayıp 8 Mbps'e, gönderiş yönünde ise 16 Kbps'ten 576 Kbps'e kadar çıkabilir.



ADSL uygulaması

İnternet

- 60'lı yıllarda ABD'de ARPANET adı altında başlatılan askeri bir iletişim projesi iken, 70'li yılların başında Amerikan üniversitelerine de bu projeden yararlanma imkânı verilmesinin ardından yaygın olarak kullanılmaya başlayan en büyük ağdır (genel ağ - küresel ağ).
- İnternet haberleşmesinde TCP/IP (Transmission Control Protocol/Internet Protocol) iletişim protokolü kullanılır .

İntranet (Özel Ağ – İç Ağ)

- Belirli bir kuruluş içindeki TCP/IP tabanlı bir ağ sistemine verilen isimdir (şirket içi Internet).
- İntranet'ler ağ geçitleri ile diğer ağlara veya Internet'e bağlanabilir. Internet çıkışı genellikle *Firewall* olarak bilinen her iki yönde de ileti trafiğini kontrol eden bir güvenlik sistemi üzerinden sağlanmaktadır.
- İntranet'te genellikle sanal IP kullanılır. Böylece sadece Internet çıkışı için tek bir gerçek IP kullanılarak iç ağdaki tüm bilgisayarlara Internet erişimi verilebilir.

IPv4 & IPv6

- Internet'e bağlı her bilgisayara yada iletişim cihazına bir adres verilmesi için 4 adet 8 bit büyüklüğünde (0-255 arasında) sayı kullanılmaktadır. (Örn: 193.255.140.17)
- Bu adresleme yöntemi ile teorik olarak en fazla $2^{32} = 4.3$ milyar adres verilebilmektedir.
- IPv6 olarak bilinen yeni adresleme yöntemi ile $2^{128} = 3.4 \times 10^{38}$ adres verilebilecektir.
- IPv6 çıkışınca eski adresleme yöntemi IPv4 olarak isimlendirilmiştir .

MAC Adresi

- IP numarası verilebilen kablolu yada kablosuz her ağ kartının 48 bitlik bir MAC adresi bulunur.
Örn: 00-23-C3-45-00-B3
- Ağ iletişiminde kullanılan çerçeveler gerçekte bu MAC adreslerini kullanarak iletim yaparlar.

DNS (Domain Name Server)

- IP adreslerinin hatırlanması zor olacağı için http://www.google.com gibi simgesel adresler (URL: Uniform Resource Locator) kullanılmaktadır.
- Web adresi olarak ta bilinen bu simgesel adreslerin IP numarası karşılıkları DNS olarak isimlendirilen sunucularda tutulmaktadır.
- Eğer sistemimize bir DNS tanımlamazsak, istenilen WEB sayfasına erişmek için o sayfanın sunucusunun IP adresini yazmamız gereklidir.

DHCP (Dynamic Host Configuration Protocol)

- Dinamik İstemci Ayarlama Protokolü, bir TCP/IP ağındaki makinelere IP adresi, ağ geçidi veya DNS sunucusu gibi ayarların otomatik olarak yapılması için kullanılır.
- Günümüzde neredeyse tüm ev ve halka açık ağlarda kullanılmaktadır, iş veya daha kontrollü bir bağlantı sağlanan yerlerde ise statik IP adresi tercih edilir.

BİLGİSAYAR MÜHENDİSLİĞİNE GİRİŞ

Sayı ve Kodlama Sistemleri

8.1.1. On Tabanlı Sayı Sistemi

Alışageldiğimiz onlu sayı dizisi taban olarak on sayısını kullanır. Örnek olarak 6789 sayısını tabana göre açarsak:

$$\begin{array}{rcl} 9 \times 10^0 = 9 & \times & 1 = 9 \\ 8 \times 10^1 = 8 & \times & 10 = 80 \\ 7 \times 10^2 = 7 & \times & 100 = 700 \\ \underline{6 \times 10^3 = 6} & \times & 1000 = 6000 \\ \text{T O P L A M} & & 6789 \end{array}$$

Bu biçimde daha genel olarak yazmak istersek,

$$(S)_{10} = T_n T_{n-1} \dots T_1 T_0, K_1 K_2 \dots K_m$$

Burada T'ler tamsayıların, K'lar ise kesirli sayıların dizilişlerini gösterir.

$$0 < T_i K_j < 9 \quad i=0, \dots, n \quad j=1, \dots, m$$

$$(S)_{10} = \sum_{i=0}^n T_i, 10^i + \sum_{j=1}^m K_j, 10^{-j}$$

Onlu sayı sisteminde en küçük rakam 0 (sıfır) dır. En büyük rakam ise $10-1=9$, rakam sayısı 10, k hane (eve) yazılabilen en büyük sayı 10^k-1 , k hane yazılabilen sayı sayısı ise 10^k , dır.

8.1.3. İkili Sayı Sistemi

Genel olarak ikili sayı sisteminde yazılış B tabanlıda olduğu gibidir. Sadece B tabanı yerine iki tabanı yer alacaktır. Bu sistemde de en küçük rakam 0 (sıfır) dır. En büyük rakam $2-1=1$ 'dir. Rakam 2, kullanılan rakamlar 0 ve 1'dir. k hane yazılabilen en büyük sayı 2^k-1 , k hane yazılabilen değişik sayısı ise 2^k kadardır.

k hane ile birbirinden değişik 2^k sayı yazılabilir. $k=3$ alınırsa üç hane yazılabilen sayılar:

İkili sayı	Onlu sayı
0 0 0	0
0 0 1	1
0 1 0	2
0 1 1	3
1 0 0	4
1 0 1	5
1 1 0	6
1 1 1	7

İkili sayıların Onlu sayılara Çevirimi (Binary to Decimal Conversion)

- $110101_2 = ?$

$$\begin{aligned}110101_2 &= 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\&= 32 + 16 + 0 + 4 + 0 + 1 \\&= 53_{10}\end{aligned}$$

Onluk – ikilik çevrimi (decimal to binary conversion)

- 1 bayt (8 bit) veri 2^8 yani 256 farklı değer alabilir

$$(00000000)_2 = (0)_{10} \quad (11111111)_2 = (255)_{10}$$

- Örn: $(175)_{10} = (?)_2$ **10101111**

$$\begin{array}{r} 175 \\ -174 \\ \hline 1 \end{array} \quad \begin{array}{r} 87 \\ -86 \\ \hline 1 \end{array} \quad \begin{array}{r} 43 \\ -42 \\ \hline 1 \end{array} \quad \begin{array}{r} 21 \\ -20 \\ \hline 1 \end{array} \quad \begin{array}{r} 10 \\ -10 \\ \hline 0 \end{array} \quad \begin{array}{r} 5 \\ -4 \\ \hline 1 \end{array} \quad \begin{array}{r} 2 \\ -2 \\ \hline 0 \end{array}$$

2 2 2 2 2 2 2

• Kesirlerin Okunması

Onlu sayı sisteminde tam basamaklar okunurken en sağdan başlayarak birer, onlar, yüzler, binler, onbinler vs gibi okunurken kesirleri okumada en soldan başlayıp onda birler, yüzde birler, binde birler ve onbinde birler diyerek okumaya devam ederiz. Örneğin;

$345.789 = 345$ tam binde 789 diye okunur.

İkili sistemde de benzer olarak, noktanın sağındakiler ikide birlerden başlayıp, dörtte birler, sekizde birler, onaltıda birler gibi devam eder.

$111.11 = 7$ tam dörtte 3

$110.101 = 6$ tam sekizde 5

$101.01011 = 5$ tam otuzikide 11

- **Toplama**

Onlu sayı sisteminde aynı düzeydeki hanelerdeki rakamların toplamı en büyük rakam olan 9'dan büyük olduğunda, bu toplamdan 10 ve 10'un katları çıkarılır ve bir üst düzeydeki haneye 10'ların sayısı eklenir. İkili sistemde ise taban 2 olduğu için iki ve ikinin katları sayısı bir sonraki haneye eklenecek toplama yapılır. Örneğin;

$$\begin{array}{r} 1011 & 11 \\ 0101 & 5 \\ 1100 & 12 \\ + \underline{1110} & + \underline{14} \\ 101010 & 42 \end{array}$$

- **Çıkarma**

Onlu sayı sistemine benzeyen çıkarmada aynı düzey hanelerdeki sayılar birbirinden çıkarılır. Çıkarma işlemi gerçekleşmiyorsa bir üst düzey haneden bir sayı alınır ve bu sayı düşük düzeyli haneye taban sayısı olarak geçirilerek çıkarma işlemi yapılır. Örneğin:

$$\begin{array}{r} 11011 & 27 \\ + 10101 & + 21 \\ \hline 00110 & 6 \end{array}$$

$$1 \text{ ler evinde} \quad 1-1 = 0$$

$$2 \text{ ler evinde} \quad 1-0 = 1$$

$$4 \text{ ler evinde} \quad 2-1 = 1$$

$$8 \text{ ler evinde} \quad 0-0 = 0$$

$$16 \text{ lar evinde} \quad 1-1 = 0$$

- Tümleyen kullanılarak çıkarma yapılması

Bir sayının tümleyenini bulmak için en kolay yol, her hanelerdeki sayıların o tabana göre en büyük rakam ile olan farkları alınır ve buna bir eklenerek bulunur. Örneğin 678 sayısının tümleyeni $321+1=322$ 'dir. Çünkü 678 sayısının her basamağını 9'dan çıkarırsak 321 elde ederiz. Buna 1 eklersek 322 eder. Söz konusu 2'li sistem olursa tümleyeni bulunacak sayıda 1'ler yerine 0, 0'lar yerine 1 koyup elde edilen sayıya 1 eklersek tümleyeni buluruz. Bilgisayarda ikili çıkarma tümleyen yoluyla toplamaya dönüştürülerek yapılır.

2 tür tümeleme vardır:

- 1'e tümeleme: 678 sayısının 1'e tümleyeni 321
- 2'ye tümeleme: 678 sayısının 2'ye tümleyeni
 $321+1=322$

İkili Sayıların Tümleyenini Alma

- 1'e TÜMLEME
- İkili sayı sisteminde bir sayının tümleyenini almak o sayının 0 olan bitlerini 1, 1 olanları da 0'a çevirerek gerçekleştirilir (tersini veya degilini almak da diyebiliriz).
- Sonuçta sayı ile o sayının tümleyeni toplandığında tüm bitlerin 1 olduğu durum elde edilir.
- Örn: Sayımız : $(10110100)_2 = (180)_{10}$
 Tümleyeni : $(01001011)_2 = (75)_{10}$

İkili Sayıların Tümleyenini Alma

- 2'ye TÜMLEME
- Sayının tümleyenine 1 eklenmesi ise «2'ye göre tümleyen» olur.

Örn: Sayımız : $(10110100)_2 = (180)_{10}$
Tümleyeni : $(01001011)_2 = (75)_{10}$
2'ye tümleyeni: $(01001100)_2 = (76)_{10}$

• Çarpma

İkili sayı sistemindeki çarpma ile onlu sayı sistemindeki çarpma kuralları aynıdır. Örneğin,

$$\begin{array}{r} 1011011 \\ \times \quad 101 \\ \hline 1011011 \\ 0000000 \\ + 1011011 \\ \hline 111000111 \end{array} \qquad \begin{array}{r} 91 \\ \times \quad 5 \\ \hline 455 \end{array}$$

• Bölme

İkili sistemde bölme kuralları onlu sistemde olduğu gibi uygulanır. Örneğin,

$$\begin{array}{r} 111000111 \\ - 101 \\ \hline 1000 \\ - 101 \\ \hline 110 \\ - 101 \\ \hline 111 \\ - 101 \\ \hline 101 \\ - 101 \\ \hline 000 \end{array} \qquad \begin{array}{c|cc} 101 & 1011011 \\ \hline & 1011011 \end{array} \qquad \begin{array}{r} 455 \quad 5 \\ - 45 \quad | 5 \\ \hline 05 \\ - 5 \\ \hline 0 \end{array}$$

8.1.4. Sekizli Sayı Sistemi

Sekizli sayı sisteminde taban sekiz olduğu için genel yazılış şöyledir:

$$(S)_8 = T_n T_{n-1} \dots T_1 T_0, K_1 K_2 \dots K_m$$

$$0 \leq T_i K_j \leq 7 \quad i = 0, \dots, n \quad j = 1, \dots, m$$

$$(S)_8 = \sum_{i=0}^n T_i \cdot 8^i + \sum_{j=1}^m K_j \cdot 8^{-j}$$

Sekizli sayı sisteminde en küçük rakam: 0, en büyük rakam: $8-1=7$, rakam sayısı: 8, k haneye yazılabilen en büyük sayı: 8^k-1 , k haneye yazılabilen değişik sayı: 8^k

Aritmetiksel işlemlerde kurallar aynen onlu ve ikili sistemde olduğu gibidir. Örneğin,

$$\begin{array}{r} 653 \\ + 362 \\ \hline 1235 \end{array}$$

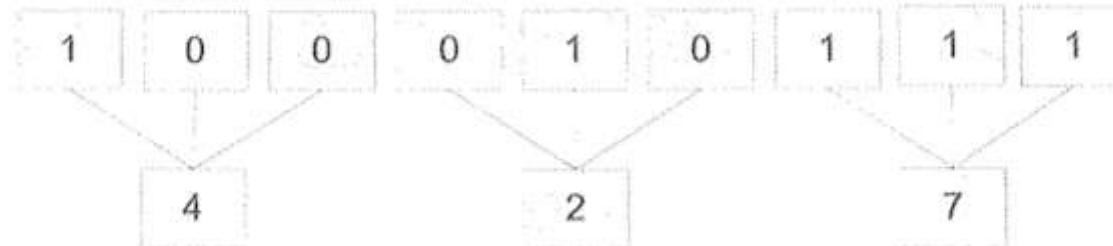
$$1\text{'ler hanesinde} : 3+2 = 5$$

$$8\text{'ler hanesinde} : 5+6 = 11 = 1 \times 8 + 3 \quad 3 \text{ elde}$$

$$64\text{'ler hanesinde} : 1+6+3 = 10 = 1 \times 8 + 2 \quad 2 \text{ elde}$$

$$512\text{'ler hanesinde} : 1 = 1$$

İkili bir sayıyı sekizliye çevirmek için sağdan başlayarak üçer üçer ayırsak ve bunların onlu karşılıklarını bulursak işlem gerçekleşir.



$$(100010111)_2 = (427)_8$$

Şekil 8.1. İkili sayıların sekizli sayılaraya dönüştürülmesi

8.1.5. Onaltılı Sayı Sistemi

Onaltılı sayı sisteminde taban 16 olduğu için tek bir basamağa sığmayacaktır. Bu nedenle 0, 1, 2, 3, 4, 5, 6, 7, 8, 9'a kadar olan basamaklar 10'lu sistemde olduğu gibi alınırken 10 yerine A, 11 yerine B, 12 yerine C, 13 yerine D, 14 yerine E ve 15 yerine de F kullanılır. Genel yazılış,

$$(S)_{16} = T_n T_{n-1} \dots T_1 T_0, K_1 K_2 \dots K_m$$

$$0 \leq T_i K_j \leq 15 \quad i = 0, \dots, n \quad j = 1, \dots, m$$

$$(S)_{16} = \sum_{i=0}^n T_i \cdot 16^i + \sum_{j=1}^m K_j \cdot 16^{-j}$$

Örneğin: $(S)_{16} = 2BE$

$$\begin{array}{rcl} 14 \times 16^0 & = 13 \times 1 & = 14 \\ 11 \times 16^1 & = 11 \times 16 & = 176 \\ \underline{2 \times 16^2} & = 2 \times 256 & = 512 \\ & & 702 \end{array}$$

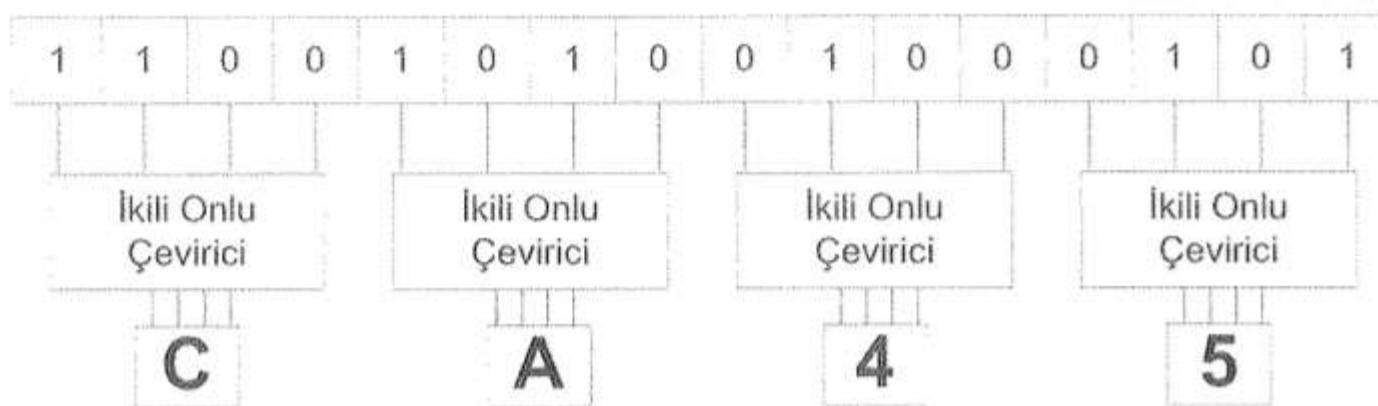
$$(2BE)_{16} = (702)_{10}$$

Decimal	Hexadecimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

Onaltılı sayı sisteminde en küçük rakam 0, en büyük rakam $16-1=15$ (F), rakam sayısı 16, k haneye yazılabilen en büyük sayı 16^k-1 , k haneye yazılabilen değişik sayı 16^k 'dır. Bu sistemde de aritmetik kurallar diğerleri ile aynıdır. Örneğin $2BD+A96 = D53$ 'dır. Yani,

$$\begin{array}{ll} \text{1'ler hanesinde} & 13+6 = 19 = 1 \times 16 + 3 \\ \text{16'lar hanesinde} & 1+11+9 = 21 = 1 \times 16 + 5 \\ \text{256'lar hanesinde} & 1+2+10 = 13 = D \end{array} \quad \begin{array}{l} 3 \text{ elde } 1 \\ 5 \text{ elde } 1 \end{array}$$

İkili sayıları 16'lı sayılaraya çevirmek için sağdan başlayarak dörder dörder ayırip 16'lı karşılıklarını yazarız.



Şekil 8.2 İkili sayıların onaltılı sayılaraya dönüştürülmesi

$$(1100101001000101)_2 = (CA45)_{16}$$

Gördüğü gibi ikili sayıları onaltılı sayıya çevirmek oldukça kolaydır.

8.1.6. Tabanlar Arası Dönüşürmeler

- **B tabanlı bir sayının onlu bir sayıya dönüştürülmesi**

$$(S)_B = T_n T_{n-1} \dots T_1 T_0, K_1 K_2 \dots K_m$$

$$(S)_{10} = T_n B^n + T_{n-1} B^{n-1} + \dots + T_1 B^1 + T_0 B^0 + K_1 B^{-1} + K_2 B^{-2} + \dots + 1$$

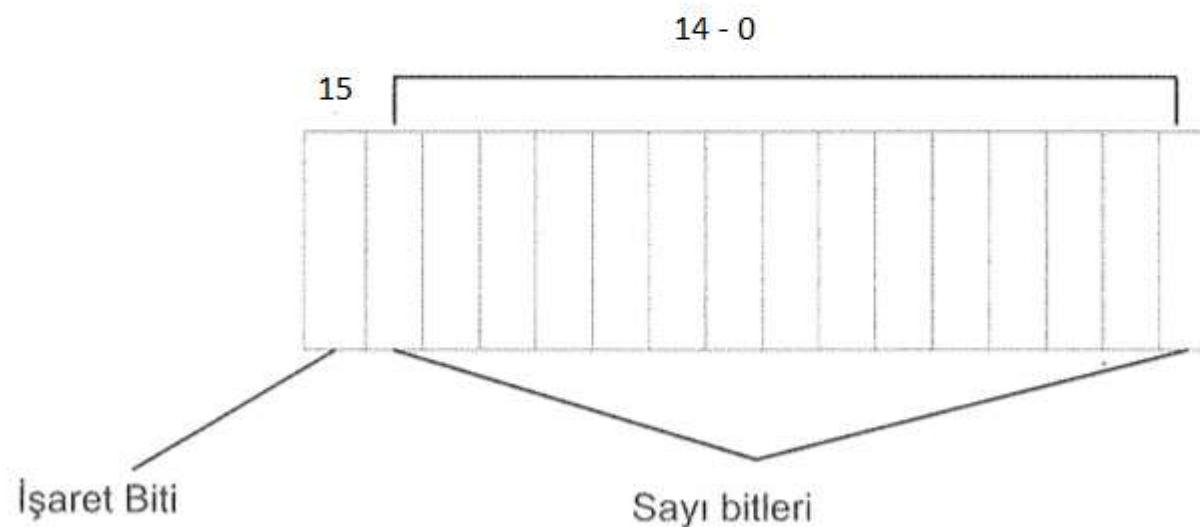
$$(S)_{10} = \sum_{i=0}^n T_i \cdot 16^i + \sum_{j=1}^m K_j \cdot 16^{-j}$$

Örnek: $(S)_2 = 10101.1011$ sayısını onlu sayıya dönüştürelim.

$$\begin{aligned}(S)_{10} &= 1x2^4 + 0x2^3 + 1x2^2 + 0x2^1 + 1x2^0 + 1x2^{-1} + 0x2^{-2} + 1x2^{-3} + 1x2^{-4} \\ &= 16 + 4 + 1 + \frac{1}{2} + \frac{1}{8} + \frac{1}{16} = 21.6875\end{aligned}$$

8.2. Bilgisayarda Tamsayıların Gösterimi

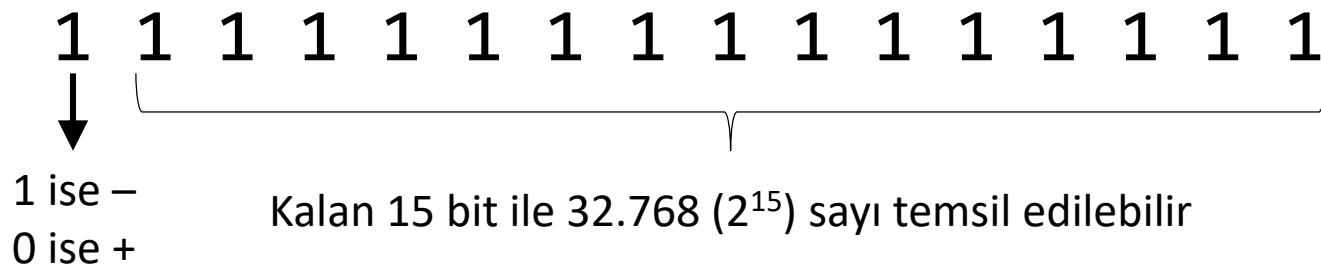
Onlu (ontabanlı) sayılar, genel olarak önce ikili sayılardaki eşitlerine çevrilir ve **tamsayıyı** veya **kayan noktalı** (floating point) olarak sunulur. İlk önce tamsayılara bakalım. Eğer 8 bitlik bir saklayıcımız varsa sıfır (0) ile $2^8 - 1$ veya 255 arasında sayılan gösterebiliriz. Saklayıcımızın sadece ikili sayıları tutması nedeniyle sekiz bit ile ancak bu büyülükteki bir sayıya izin verir. Eğer sayıcımız 16 bitlik ise sıfır (0) ile $2^{16} - 1$ (65535) arasındaki sayıları tutabiliriz. Bir Byte'in 8 bit olduğu düşünülürse 16 bitlik bir saklayıcı 2 Byte'lık yer tutar. Buna **yarım sözcük** (Half Word) denilir.



Şekil 8.3 Tamsayılar için yarım sözcük (2 Byte) biçimi

Negatif sayılar nasıl temsil edilir?

- Negatif tamsayıları temsil etmek için 1 bitlik bilginin işaret (- veya +) için kullanılması düşünülebilir.
- Örneğin 16 bit veri boyutu için:



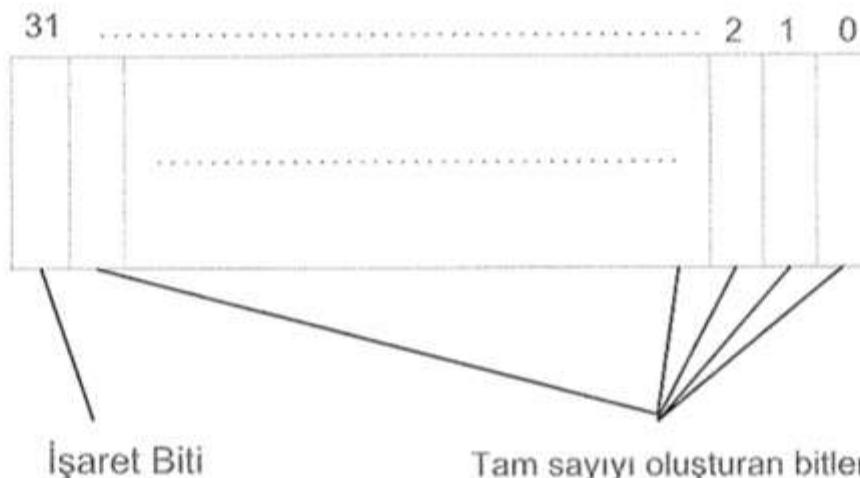
- Bu yöntemin olumsuz yanı sıfırın hem negatif hem de pozitif olarak değerlendirilmesidir (16 bitin tamamı 0 ise '+0', işaret biti 1 diğerleri 0 ise '-0')).

Bu yöntem ile -32.767 ile +32.767 arasındaki sayıları temsil edebiliriz.

Sayıının büyüklüğü, saklayıcının bit sayısına bağlıdır. Burada N saklayıcındaki bit sayısı ise en büyük sayı $2^N - 1$ ile sınırlı olacaktır. Tamsayıları gösterirken sayının eksi ya da artı olduğunu da göstermek gerekir. Bunun için en soldaki bit alınır. Eğer en soldaki bit 1 (**bir**) ise sayı *eksi*. 0 (**sıfır**) ise *artıdır*. Bu durumda 16 bitlik bir saklayıcıda -2^{15} ile $+2^{15}$ yani -32768 ile 32767 arasında sayılara izin verilecektir. Bunun nedeni sıfır değerinin de bir kombinasyon oluşturmasıdır. Bunu sınamak için iki bitlik bir saklayıcıyı alalım:

Bit 1	Bit 0	Onlu Değer
0	1	+1
0	0	0
1	0	-0  -1
1	1	-1  -2

Burada 2^2 veya 2^N kombinasyon vardır ve bir kombinasyon sıfır içindir. Eksi sayılar ikinin tümleyenini olarak sunulurlar; en soldaki bit sayının işaretini gösterir.



Şekil 8.4. Tamsayılar için tam sözcük (4 Byte) biçimi

Kaç bit gereklidir?

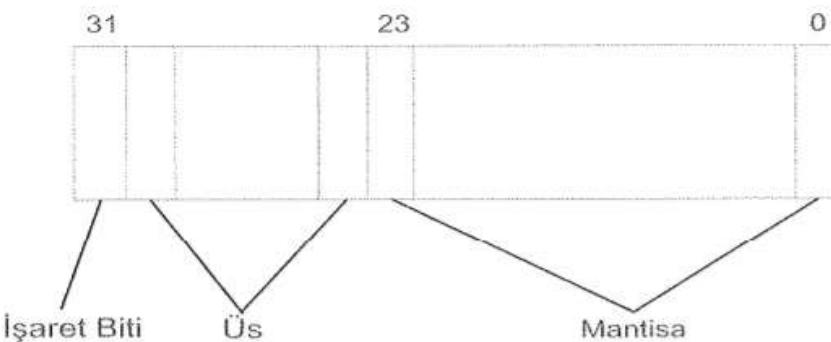
- n bit ile temsil edilebilecek en büyük sayı = $2^n - 1$ 'dir.
 - $n = 4$ ise 15, $n = 8$ ise 255, $n = 16$ ise 65535, ...
- Soru: $(1.000.000)_{10}$ sayısını 2'li sayı sisteminde temsil edebilmek için kaç bit gereklidir?
 - $2^n - 1 \geq 1.000.000$ olması gereklidir. Bu durumda n en az 20 olmalıdır ($2^{10} = 1024$ olduğuna göre $2^{20} = 1024^2$)
 - NOT: Bilgisayarda sayılar genellikle 1 bayt, 2 bayt, 4 bayt veya 8 bayt gibi büyülüklerde saklanır. 3 bayt ya da 5 bayt gibi veri büyülükleri programlama dillerinde yoktur. 1 milyon sayısı için 2 bayt (16 bit) yetmeyeceği için 4 bayt (32 bit) bir veri büyülügü seçmeliyiz.

32 bit ile $2^{32} - 1 = 4.294.967.295'$ e kadar sayıları temsil edebiliriz.

8.3. Kayan Noktalı Sayıların Sunumu

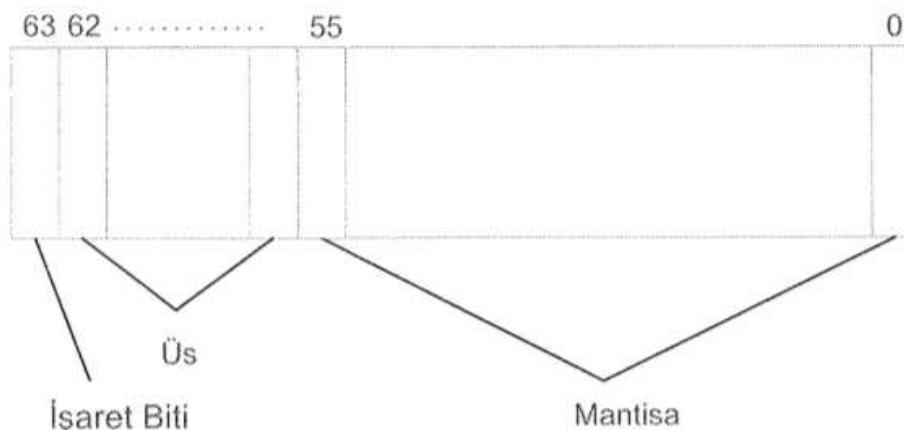
5.724 gibi bir sayıyı sunmak için veya çok büyük bir sayıyı tutmak için nasıl davranışları. Böyle sayıları tutmak için kayan ondalık nokta içeren bir sistem kullanılır. Bu sistem mühendislerin veya fen bilimcilerin kullandığı gösterimle benzerdir.

- Ondalıklı sayılar bilgisayar biliminde kayan noktalı sayılar (floating point numbers) olarak bilinir. Bunun sebebini bir örnek ile açıklayabiliriz:
- 123.4567 sayısı aşağıdaki şekillerde gösterilebilir:
 $12.34567 \times 10^1 = 1.234567 \times 10^2 = 0.1234567 \times 10^3$
- Göründüğü gibi noktanın solunda sıfır ulaşana kadar n basamak kaydırıldığımızda sayının yanına $\times 10^n$ eklenir. Bilgisayarda sadece 1234567 (mantis) ve n (üs) değerlerini tamsayı olarak saklamak yeterli olacaktır.
- Eğer ondalıklı sayı 32 bit büyüklükte saklanacaksı 23 bit mantis için 8 bit üs için 1 bit ise işaret için kullanılır (64 bit için ise sırasıyla 52, 11, 1 şeklinde olacaktır).



Şekil 8.5. Kayan noktalı sayılar için tam kelime (4 Byte) biçimi

Kayan noktalı ve tamsayıları işlemek için genelde 32 bitlik saklayıcılar kullanılır. Her Byte’ın 8 bit olduğu düşünülürse, $32/8=4$ Byte’lık saklayıcılar kullanılmalıdır. Bellekte dört Byte’lık yere bir **tam sözcük** denilir. Kayan noktalı sayılarında mantisadaki duyarlılığı artırmak için **çift sözcük (double word)**’luk yerler kullanılabilir. Buna **çift duyarlık (double precision)** da denilir.



Şekil 8.6. Kayan noktalı sayılar için çift sözcük (8 Byte) biçimi

Çift Duyarlık, duyarlığını belli ölçüde artırırken bellekte daha fazla yer kaplamaktadır. Tipik olarak büyük bilgisayarların çoğunda 32 bitlik sayılar kullanılır. Burada 8 bit üs ve işaret için ayrılrken 24 bit **mantisa** için kullanılır. [Bartee, 1981]

Devamı Haftaya ...

BİLGİSAYAR MÜHENDİSLİĞİNE GİRİŞ

Sayı ve Kodlama Sistemleri

BÖLÜM 2

8.4. ABeCesel (Alfabetic) ve Diğer Karakterlerin Sunumu

8.4.1 Kodlama Sistemleri

Sayılar ve karakterleri göstermede kullanılan bir kaç kodlama sistemi vardır. Fakat bunlardan iki tanesi çok yaygın kullanılmaktadır. İlkı, ASCII (American Standard Code For Information Interchange) (Aski okunur), diğeri ise EBCDIC (Extended Binary Coded Decimal Interchange Code) (Ebsidik okunur)'dır.

- **EBCDIC ve ASCII kodları**

En yaygın olarak, karakterleri göstereme de kullanılan üç kod: BCD, EBCDIC ve ASCII'dır. Her bir kod aşağıda gösterildiği gibi bölge ve sayısal alanlara bölünmüştür.

BCD de kullanılan altı bitlik biçim

<i>bölge</i>	<i>sayısal</i>
0 1	2 3 4 5

EBCDIC ve ASCII de kullanılan sekiz bitlik biçim

<i>bölge</i>	<i>sayısal</i>
0 1 2 3	4 5 6 7

ASCII (American Standard Code for Information Interchange)

- Bilgisayarda metin türü verilerin işlenebilmesi veya saklanabilmesi için büyük ve küçük harflerin, noktalama işaretlerinin ve klavyedeki diğer yardımcı karakterlerin ikili kodlar ile ifade edilmesi gereklidir.
- 50'li yıllarda her üretici firma farklı bir kodlama kullanırken, 1963 yılında yayınlanan ASCII standarı ile tüm bilgisayarların aynı kodlamayı kullanması sağlanmıştır. İlk etapta oluşturulan 7 bitlik kodlara sahip (128 farklı kod içeren) ASCII tablosuna, sonradan 1 bit daha ilave edilerek 256 kodluk «genişletilmiş ASCII tablosu» oluşturulmuştur.

Standart ASCII Tablosu

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

- 128 kodun 33 tanesi ekrana basılmayan, 95 tanesi basılabilen karakterden oluşur.
 - 0-31 arasındaki sayılar, yazıcı gibi bazı çevresel aygıtlar için kullanılan denetim karakterlerine atanmıştır.
 - Örn: 12 (hex C) **form besleme/yeni sayfa** (FF: Form Feed) işlevine ayrılmıştır. Bu komut yazıcıya bir sonraki sayfanın başına atlama bilgisi verir.
 - Örn: ENTER tuşuna bastığımızda 10 (hex A) ve 13 (hex D) kodları üretilir. 10 **satır başı** (LF: Line Feed) 13 ise **alt satır** (CR: Carriage Return) karşılığıdır. İkisi de ekranada görünmez, alt satırın başına gitme işini gerçekleştirir.

Genişletilmiş Ascii Tablosu

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Space	64	40	100	@	Ø	96	60	140	`	~
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	:	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL

ASCII Veri – Sayısal Veri Farkları

106’yı ASCII kodunda tutabilmek için her karakter için 7 ve bu üç karakter içinde 21 bit gereklidir. Bu onaltı sayılar her karakter için bu özel koda çevrilirse:

1 karakterini	0110001
0 karakterini	0110000
6 karakterini	0110110

olarak tanıtırız.

Sayısal 106

$$(106)_{10} \rightarrow (0110\ 1010)_2$$

Metin 106

$$\text{“106”} \rightarrow 00110001\ 00110000\ 00110110$$

Genişletilmiş (Extended) ASCII

- Standart ASCII tablosu İngilizce için yeterli olsa da diğer dillerde olan bazı harfleri, matematiksel semboller ve tablo çizgileri oluşturan karakterler gibi özel simgeleri içermediği için bu tablo genişletilerek 256 koda çıkartılmıştır.
- Örneğin â, ä, Ä, Ç, ç, Ö, ö, Ü, ü gibi harfler, π , α , β gibi matematiksel semboller, \ll , \lll , \llcorner , \lrcorner , \mid gibi kutu çizim simgeleri 128-255 aralığında (genişletilmiş kısımda) yer alır.
 - Örn: Yandaki tablo karakterler ile çizilmiştir

0-127 Standard

128-255 Genişletilmiş

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	Ø	128	80	Ҫ	160	A0	á	192	C0	ܲ	224	E0	ܾ
1	01	Start of heading	33	21	!	65	41	À	129	81	ü	161	A1	í	193	C1	ܴ	225	E1	ܶ
2	02	Start of text	34	22	"	66	42	Ɓ	130	82	é	162	A2	ó	194	C2	ܵ	226	E2	ܷ
3	03	End of text	35	23	#	67	43	Ҫ	131	83	â	163	A3	ú	195	C3	ܹ	227	E3	ܻ
4	04	End of transmit	36	24	\$	68	44	߱	132	84	ä	164	A4	ñ	196	C4	-	228	E4	ܰ
5	05	Enquiry	37	25	%	69	45	߱	133	85	à	165	A5	ܵ	197	C5	ܶ	229	E5	ܸ
6	06	Acknowledge	38	26	&	70	46	߱	134	86	ܶ	166	A6	ܷ	198	C6	ܸ	230	E6	ܺ
7	07	Audible bell	39	27	'	71	47	߱	135	87	ܵ	167	A7	ܹ	199	C7	ܶ	231	E7	ܻ
8	08	Backspace	40	28	(72	48	߱	136	88	ܶ	168	A8	ܸ	200	C8	ܶ	232	E8	ܰ
9	09	Horizontal tab	41	29)	73	49	߱	137	89	ܶ	169	A9	ܷ	201	C9	ܶ	233	E9	ܸ
10	0A	Line feed	42	2A	*	74	4A	߱	138	8A	ܶ	170	AA	ܶ	202	CA	ܶ	234	EA	ܰ
11	0B	Vertical tab	43	2B	+	75	4B	߱	139	8B	ܶ	171	AB	ܶ	203	CB	ܶ	235	EB	ܶ
12	0C	Form feed	44	2C	,	76	4C	߱	140	8C	ܶ	172	AC	ܶ	204	CC	ܶ	236	EC	ܰ
13	0D	Carriage return	45	2D	-	77	4D	߱	141	8D	ܶ	173	AD	ܶ	205	CD	=	237	ED	ܰ
14	0E	Shift out	46	2E	.	78	4E	߱	142	8E	ܶ	174	AE	«	206	CE	ܶ	238	EE	ܶ
15	0F	Shift in	47	2F	/	79	4F	߱	143	8F	ܶ	175	AF	»	207	CF	ܶ	239	EF	ܰ
16	10	Data link escape	48	30	0	80	50	߱	144	90	ܶ	176	B0	ܶ	208	DO	ܶ	240	FO	ܰ
17	11	Device control 1	49	31	1	81	51	߱	145	91	ܶ	177	B1	ܶ	209	D1	ܶ	241	F1	ܶ
18	12	Device control 2	50	32	2	82	52	߱	146	92	ܶ	178	B2	ܶ	210	D2	ܶ	242	F2	ܶ
19	13	Device control 3	51	33	3	83	53	߱	147	93	ܶ	179	B3	ܶ	211	D3	ܶ	243	F3	ܶ
20	14	Device control 4	52	34	4	84	54	߱	148	94	ܶ	180	B4	ܶ	212	D4	ܶ	244	F4	ܶ
21	15	Neg. acknowledge	53	35	5	85	55	߱	149	95	ܶ	181	B5	ܶ	213	D5	ܶ	245	F5	ܶ
22	16	Synchronous idle	54	36	6	86	56	߱	150	96	ܶ	182	B6	ܶ	214	D6	ܶ	246	F6	ܶ
23	17	End trans. block	55	37	7	87	57	߱	151	97	ܶ	183	B7	ܶ	215	D7	ܶ	247	F7	ܶ
24	18	Cancel	56	38	8	88	58	߱	152	98	ܶ	184	B8	ܶ	216	D8	ܶ	248	F8	ܶ
25	19	End of medium	57	39	9	89	59	߱	153	99	ܶ	185	B9	ܶ	217	D9	ܶ	249	F9	ܶ
26	1A	Substitution	58	3A	:	90	5A	߱	154	9A	ܶ	186	BA	ܶ	218	DA	ܶ	250	FA	ܶ
27	1B	Escape	59	3B	:	91	5B	߱	155	9B	ܶ	187	BB	ܶ	219	DB	ܶ	251	FB	ܶ
28	1C	File separator	60	3C	<	92	5C	߱	156	9C	ܶ	188	BC	ܶ	220	DC	ܶ	252	FC	ܶ
29	1D	Group separator	61	3D	=	93	5D	߱	157	9D	ܶ	189	BD	ܶ	221	DD	ܶ	253	FD	ܶ
30	1E	Record separator	62	3E	>	94	5E	߱	158	9E	ܶ	190	BE	ܶ	222	DE	ܶ	254	FE	ܶ
31	1F	Unit separator	63	3F	?	95	5F	߱	159	9F	ܶ	191	BF	ܶ	223	DF	ܶ	255	FF	ܶ

Unicode

- 256 karakterlik ASCII kod tablosunda Yunan, Kiril, Arap ve Japon alfabesi gibi farklı alfabelere ait karakterlerin olmadığı (Türkçedeki Ş, ş, Ğ, ğ, İ ve î karakterleri de yok) dikkatinizi çekmiştir.
- 90'lı yıllara kadar bu afabelere özgü farklı yazı tipleri (font) geliştirilerek bu soruna çözüm bulunmuş, 1991 yılında yayınlanan ve her elemanın 16-bit bir kod ile temsil edildiği 65536 farklı eleman içerebilen Unicode (Universal Code: Evrensel Kod) ile tüm dillere ait karakterleri, matematiksel sembollerini, vs. içeren ortak bir kod tablosu oluşturulmuştur.

Ses Verileri

Ses bir dalga olduğu için (frequency) **sıklığı** ve (amplitude) **genlik** özellikleri vardır. Sıklığın birimi Hertz (Hz) ile gösterilirken genliğin birimi Desibel (dB) ile tanımlanır. Genlik düşükse ses kalın (bas), yüksekse ince (tiz) olarak duyulur. Genlik ise yüksek ise (sesli), alçak ise (sessiz) duyulur.

Sesin özniteliklerinin bilgisayar ortamında kayıt edilebilmesi için **analog'dan sayısal'a** dönüştürülmesi **ADC** (Analog to Digital Converter) aracılığı ile gerçekleşir. Örnekleme denilen bu işlemin sayısal kaydı sırasında kullanılacak bit uzunluğu ("bit resolution") **bit çözünürlüğü** olarak bilinir. Bit dizelerinden ses elde etmek için ise **DAC** (Digital to Analog Converter) kullanılır.

Ses bilgisi (Wav—"sound Wave"), (MP3—"MPEG audio layer 3") kayıtlanırken en yaygın bu ("format") **birimlerde** kullanılır. Müzik aletlerinin notalarının nasıl çalınacağı komutlar ile gerçekleştirmeye ("Synthesis") **birleşim** denir. Bunun en yaygın örnek ("Musical Instrument Digital Interface "-MIDI) **Müzik Aleti Sayısal Arayüzü**dü **MIDI** dosyaları içinde müzik aletinin notayı ne kadar süre ile çaldığı bilgisi de tutulur. Ayrıca gerçekçi müzik üretimi için ("Wave Table Syntesis") **dalga tablosu birleşim** teknigi kullanılır.

Resim verileri

Resim verilerini bilgisayarda tutmak için resmi oluşturan noktalarına ayırmamız gereklidir. Daha sonra her noktaya (“Red Green Blue”-RGB) **Kırmızı Yeşil Mavi** bileşi minden oluşan bir renk atayabiliriz. Bu noktaları yan yana ve alt alta dizdiğimizde gelen rüntüyü oluşturabiliriz. Resmi oluşturan nokta sayısına (“resolution”) **çözünürlük** denir. Resmi oluşturan noktalara da (“Picture Elements”-Pixel) **resim öğesi** denir. Bir resim öğesi için ayırdığımız bit dizisinin büyüklüğüne (“color depth”) **renk derinliği** denilir. 16 bitlik renk derinliği ile (“Hi Color”) **yüksek renk** 655336 renkten biri görüntüsine atanabılırken 24 bit ile 16,7 milyon (“True Color”) **gerçek renk** ten biri atanabilir. Resimleri resim öğesi şeklinde saklamaya (“raster based”) **izgara tabanlı** saklama denilir. Resim bilgisi saklamak için yaygın kullanılan dosya biçimleri *jpg*, *gif*, *png* ve *bmp* dir. **Jpeg/Jpg** (“Joint Photographic Experts Group”) renkli ve gri tonlamalı fotoğraf türlerini kayıplı sıkıştırma denilen yüksek sıkıştırma oranlı saklama yapabilecek bir dosya türüdür. **Gif** (“Graphic Interchange Format”) **Grafik Değişen Biçim**’deki resimler en çok 256 renk içerebilir. Bu biçim hareketli ve arka planı saydam resimleri destekler. **Png** (“Portable Network Graphics”) taşınabilir ağ grafikleri, *gif*’e oranla daha yüksek sıkıştırma oranları ve saydamlığının değişik oranlarda olmasını sağlayabilen **Bmp** (“Bitmap”) bit haritası resim bilgilerini bit bit saklar.

Video verileri

Video gösterimlerini elde etmek için resim karelerinin saniyede yaklaşık 25-30 tanesini ard arda göstermek, bir bakıma sinema tekniğini kullanmak gerekecektir. Ancak çok sayıda resim kullanılacağı için büyük miktarda disk alanın gerek duyulacak ve verileri Internet hatlarında taşınabilmesi de ikinci bir sorun oluşturabilecektir. Burada sıkıştırır.

Özel Anahtar Şifreleme

Şifreleme teknikleri, sıkılıkla (public key) **kamusal/açık anahtar** veya (private key) **özel anahtar** olarak sınıflandırılır. Özel anahtar şifreleme, hem gönderici hem de alıcının tekil bir anahtarı bilmelerini gerektirir. Burada, tipik olarak, kullanımda olan özel bir yöntemin gereklerini yerine getirecek büyük bir sayı dikkatlice seçilmelidir.

Varsayıyalım ki:

- **K**, hem gönderici hem de alıcı tarafından bilinen bir anahtardır.
- **M**, şifrelenecek iletidir.
- **E** ise **K** ve **M** ye dayanan iletleri şifreleyecek fonksiyondur.

Bu durumda şifrelenmiş ileti :

$$C = E(K, M)$$

D fonksiyonu, **K** anahtarları ve **C** şifrelenmiş iletiyi kullanarak şifreyi çözecektir.

$$M = D(K, C)$$

Bu yolla, iki güvenlik **paydaşı**, tekil anahtar (**K**) yi kullanarak şifrelenmiş iletileri karşılıklı değiştirebilir.

Kamu Anahtar Şifreleme

Giderek birçok gerçekleştirimde yaygınlaşıyor. Diğer özellikler arasında, kamu anahtar dağıtım problemini azaltıyor. Kamu anahtar şeması aşağıdaki gibi çalışıyor:

Her paydaş iki anahtara sahiptir:

- **Pr**, bir özel anahtardır ve hiç kimse tarafından bilinmez.
- **Pu**, kamu anahtarı, herkes tarafından bilinir.

Pu aynı zamanda bir **dizine** de konmuş olabiliyor. Şifreleme fonksiyonu **E**, **M** iletisini ve alıcının özel anahtarını gerektiriyor. Bu durumda şifrelenmiş ileti **C**:

$$C = E(Pu(\text{Alıcı}), M)$$

Herhangi biri, şifrelenmiş bir iletiyi herhangi bir alıcıya gönderebiliyor.

D, şifre çözme fonksiyonu, **C** şifrelenmiş ileti ve **Pr** alıcının özel anahtarını kullanıyor. Böylece, sözü edilen alıcı iletiyi çözebiliyor. Burada,

$$M = D(Pr(\text{Alıcı}), C)$$

RSA kamu anahtar sistemi, ismini mucidinden alan ve en iyi bilinen sistemdir. RSA şifreleme ve şifre çözme fonksiyonlarını yaratmak için sayılar kuramı sonuçlarını zekiçe kullanır. Burada söz konusu olan özel anahtar iki büyük **asal sayının çarpımından** oluşan bir sayı olarak ortaya çıkabilir. [Hart, J, M, Rosenberg, B-1995]

ÖRNEKLERLE KONU ANLATIMI

1. (10p) Hız, maliyet ve boyut açısından Register, RAM, Cache Bellek ve HDD hafıza elemanlarını karşılaştırınız. Çözüm adımlarının açıklamaları yazılmadan verilen cevaplar geçersiz sayılacaktır. Düz metin alanında cevaplanması istenilen sorularda verilen linkler dikkate alınmayacaktır.

3. (10p) a) RISC ve CISC komut türlerine göre işlemcileri karşılaştırınız.
b) Bir işlemcinin İç yapısını çiziniz. Her bir donanımın görevini yazınız.
Çözüm adımlarının açıklamaları yazılmadan verilen cevaplar geçersiz sayılacaktır. Düz metin alanında cevaplanması istenilen sorularda verilen linkler dikkate alınmayacaktır.

**A dizisinin eleman
değerlerinden
çift sayı olanları 0,
tek sayı olanları 1
yapan algoritma**

Harddisk → 1 TB sığa

RAM → 16 GB kapasite

İşlemci → 3 GHz, 6 MB Cache

Ekran Kartı → 2 GB harici hafıza

7. (15p) Aşağıda verilen Boole cebri denklemini en sade haline getirerek, buna karşılık gelen lojik devreyi çiziniz. Çözüm adımlarının açıklamaları yazılmadan verilen cevaplar geçersiz sayılacaktır. Düz metin alanında cevaplanması istenilen sorularda verilen linkler dikkate alınmayacaktır.

$$F = XYZT + X'Y'ZT' + X'YZT' + X'YZ' + XYZ'T' + X'Y'ZT + XY'$$

$$F = XYZT + X'Y'ZT' + X'YZT' + X'YZ' + XYZ'T' + X'Y'ZT + XY'$$

$$F = XYZT + X'Y'ZT' + X'YZT' + X'YZ' (T+T') + XYZ'T' + X'Y'ZT + XY'(Z+Z')(T+T')$$

$$F = XYZT + X'Y'ZT' + X'YZT' + X'YZ'T + X'YZ'T' + XYZ'T' + X'Y'ZT + \\ X'Y'ZT + X'Y'ZT' + X'Y'Z'T + X'Y'Z'T'$$

⊕

		XY			
		00	01	11	10
ZY	00	1	1	1	
	01		1		1
	11	1		1	1
	10	1	1		1

8. (15p) a) Ahmet çözümünürlüğü 20 MP, hızı 30 fps olan bir kamera ile 5 dk video çekerek, bağlantı hızı 256 Mbps olan bir iletim hattından Mehmet'e göndermektedir. Hattın tam kapasite çalıştığını ve kameranın bir pikselinin 1 bit olduğunu dikkate alarak bu aktarımın kaç dk sürecekini açıklayarak hesaplayınız.

b) IPv6, Linker ve Huffman Kodlama kavramlarını birer örnek üzerinden açıklayınız.

Çözüm adımlarının açıklamaları yazılmadan verilen cevaplar geçersiz sayılacaktır. Düz metin alanında cevaplanması istenilen sorularda verilen linkler dikkate alınmayacaktır.

Çözüm:

20 MP = 20 Megabit

30 FramePerSecond = 30×20 Megabit (1 saniyedeki video boyutu)

5 dk = $5 \times 60 = 300$ saniye

1 saniyede 30×20 Megabit veri oluşursa 300 saniye kaydedilen videonun boyutu $300 \times 30 \times 20$ Megabit olur

256 Mbps = 1 saniyede 256 Megabit veri aktarımı demektir.

O zaman iletim hattı

1 saniyede 256 Megabit aktarıyorsa

X saniyede $300 \times 30 \times 20$ Megabit aktarır (doğru orantı)

X = 703 saniye = 11,72 dakika (yaklaşık)

1) 10 GB lık veriyi 256 Mbps hızda sahip bir iletim hattı ile tam kapasite kaç dk çalıştırarak iletebileceğimizi açıklayarak hesaplayınız?

Temel mikrobilgisayar sistemi elemanlarını çizerek gösteriniz.

Assembly programlamada kullanılan yönerge kavramının ne olduğunu detaylı olarak açıklayarak örnekler veriniz.

TCP/IP mimarisinin katmanlarını yazınız.

Uygulama-Ulaşım-Yönlendirme-Fiziksel

Her pikselin bir bit olduğunu dikkate alarak; çözünürlüğü 16 MP, hızı 100 fps olan bir kamera ile 8 dk video çekilirse kaç GBlık hafıza kullanılacağını bulunuz.

$$16 \cdot 10^6 \cdot 100 \cdot 8 \cdot 60 / (8 \cdot 1024 \cdot 1024 \cdot 1024) = 96 \text{ GB}$$

- 1. (20p)**
- a) Assembly dilinde adresleme yöntemlerinin isimlerini yazınız.
 - b) Bir assembly komutunun çalışma aşamalarını yazınız.

Her pikselin bir bit olduğunu dikkate alarak; çözünürlüğü 16 MP, hızı 60 fps olan bir kamera ile 10 dk video çekilirse kaç GBlık hafıza kullanılacağını bulunuz.

$$16 \cdot 10^6 \cdot 60 \cdot 10 \cdot 60 / (8 \cdot 1024 \cdot 1024 \cdot 1024) = 72 \text{ GB}$$

- 1. (30p)**
- a)** Soket numarası, DNS, IPv6 ve Çerçeve kavramlarını açıklayınız.
 - b)** En az 8 tane ağ cihazı yazınız.
 - c)** Ağ topolojilerini şekillerini vererek yazınız.
-
- b)** hub, switch, modem, router, firewal, ethernet, gateway
 - c)** Halka, yıldız, ortak yol, ağaç, örgü

- a) 512 Kbps hızı sahip bir iletim hattı tam kapasite 8 dk çalışırsa iletilen veri miktarı, 1 GBlık harddiskin yüzde kaçını doldurur?
- b) Bir mikroişlemcide bir komutun çalışma adımlarının neler olduğunu yazınız.
- c) Asembly programlamada kullanılan en az 5 adresleme yöntemlerini yazınız.

- a) $30 \text{ MB} / 1000 \text{ MB} = \%3$**
- b) Getir – Çöz – Çalıştır - Sonucu Yaz**
- c)**

b) 1024 Mbps hızda sahip bir iletim hattı tam kapasite 15 dk çalışırsa kaç GigaByte veri ileter?

$$\text{b)} \frac{1024 * 1024 * 1024 * 15 * 60}{(1024 * 1024 * 8 \text{ bit})} = 112,5 \text{ GB}$$

b) 512 Mbps lik iletim hızına sahip bir hat tam kapasite kaç dk çalışırsa 200 MB'lık bir veriyi iletebilir? Açıklayarak hesaplayınız.

$$\text{b)} \frac{x * 512 * 1024 * 1024}{(8 * 1024 * 1024)} = 200 \text{ ise}$$
$$x = 3 \text{ dk } 20 \text{ s}$$

c) Çözünürlüğü 8 MP, hızı 30 fps olan bir kamera ile 5 dk video çekilirse kaç MB lık hafıza kullanılır.

$$c) 8 \cdot 10^6 \cdot 30 \cdot 5 \cdot 60 / (8 \cdot 1024 \cdot 1024) = 9000 \text{ MB}$$

b) 1024 Mbps hızda sahip bir iletim hattı tam kapasite 15 dk çalışırsa kaç GiggaByte veri iletir?

d) RISC ve CISC mimarilerinin karşılaştırmasını yapınız.

1. (15p) 128 Mbps hızda sahip bir iletim hattı tam kapasite 4 dk çalışması sonucu iletilen veri bir harddiske kaydedilmektedir. Harddiskin tam kapasitesi 80 GB olduğuna göre, aktarılan veri bu harddiskin yüzde kaçını dolduracağını açıklayarak hesaplayınız?

Çözüm:

$$4 \cdot 60 \cdot 128 \cdot 1024 \cdot 1024 / (8 \cdot 1024 \cdot 1024 \cdot 1024) = 5 \text{ GB}$$
$$5 / 80 = 6.25 \%$$

1. (30p) Aşağıdaki soruların cevaplarını bulunuz.

- a) 5 Terabyte'lık bir harddisk hafıza alanı açısından kaç tane 512 Megabyte'lık harddiskle eşdeğerdir.
- b) Bir mikroişlemcide bir komutun çalışma adımlarının neler olduğunu yazınız.
- c) En az 6 işletim sistemi örneği yazınız.

Çözüm:

- a) $5 \times 1024 \times 1024 / 512 = 10240$ tane
- b) Getir – Çöz – Çalıştır - Sonucu Yaz
- c) Windows-Linux-Unix-Android-iOS

- 1. (15p)**
- a)** 2 Terabyte içerisinde kaç tane 512 MB vardır.
 - b)** 512 Kbps hızı sahip bir iletim hattı tam kapasite çalışırsa 30 MegaByte lık veri kaç dakikada iletilir?
 - c)** XNOR kapısının tanım tablosunu çiziniz.

Çözüm:

- a)** 4096
- b)** 8 dk

- b)** Assembly dilinde adresleme yöntemlerinden en az 5 tanesini yazarak, birisi için örnek kod yazınız.
- b)** Saklayıcı, doğrudan, dolaylı, ivedi, bağıl adresleme
Add a,r7