

Bilgisayar Sistemi İşleyişi

- ❑ I/O cihazları ve CPU aynı anda çalışabilir
- ❑ Her bir cihaz denetleyicisi belli bir tip cihazın kontrolünden sorumludur
- ❑ Tüm cihaz denetleyicilerinin bir **yerel tampon belleği (local buffer)** vardır
- ❑ CPU ana hafıza ile yerel tampon bellekler arasında çift yönlü veri taşır
- ❑ I/O işlemi, cihazdan, denetleyicinin yerel tampon belleğine doğrudur
- ❑ Cihaz denetleyicisi, işeminin bittiğini, işlemciye **kesinti (interrupt)** göndererek bildirir

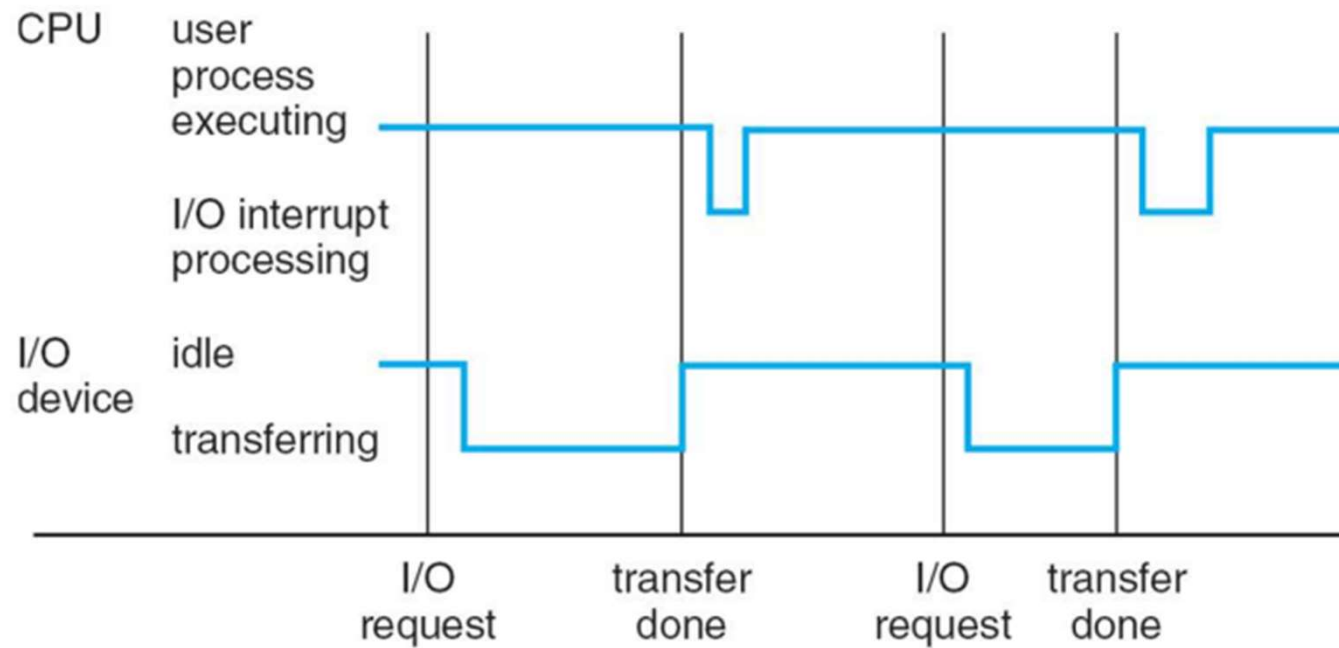
Kesintilerin Genel Özellikleri

- ❑ Kesintiler kontrolü, o kesintiye ait **kesinti servis rutinine (interrupt service routine)** yönlendirir
- ❑ Servis rutinleri, kesinti sonucu yapılması gereken işi gerçekleştiren yazılım parçacıklarıdır
- ❑ Hangi servis rutininin hangi hafıza adresinde bulunduğu **kesinti vektöründe (interrupt vector)** bulunmaktadır
- ❑ Bilgisayar, kesinti sonunda yarıda kesilen işleme geri dönebilmek için, kesilen işlemin işletilen son komutunun adresini saklamalıdır
- ❑ Kayıp kesintilere engel olmak için kesinti işletildiği sürece yeni kesinti gönderimine izin verilmez
- ❑ **Tuzak (trap)** yazılım tarafından oluşturulan kesintilerdir
- ❑ Tuzaklara yazılım hataları ya da kullanıcı istekleri neden olur
- ❑ İşletim sistemleri kesintilerle yönlendirilirler (interrupt driven)

Kesintilerin İşletilmesi

- ❑ İşletim sistemi CPU'nun durumunu kaydeder: **yazmaçlar (registers)** ve **program sayacı (program counter)**
- ❑ Hangi tür kesintinin gerçekleştiğini belirler:
 - ❑ **sorgulama (polling)** – hangi cihazdan gerçekleştiği bulunmalıdır
 - ❑ **vektör kesinti sistemi (vectored interrupt system)** – cihazı belirten kod, kesinti ile birlikte gönderilir
- ❑ Her bir kesinti için hangi işlemin gerçekleştirileceğini ayrı bir kod parçası belirler

Kesinti Zaman Çizelgesi



Direk Hafıza Erişim Yapısı

- ❑ **Direk Hafıza Erişimi** – Direct Memory Access (DMA)
- ❑ Hafıza hızına yakın bilgi aktarması yapabilen yüksek hızlı I/O cihazları için kullanılır
- ❑ Cihaz denetleyicisinin, CPU'nun çalışmasını bölmeden, veri bloklarını cihazın tampon belleğinden direk olarak hafızaya aktarmasıdır
- ❑ Her byte için kesinti göndermek yerine, her bir blok için bir kesinti gönderilir

Depolama Birimi Yapısı

- ❑ **Ana hafıza (main memory)** – CPU'nun direk erişebileceği tek geniş depolama birimidir
- ❑ **İkincil depolama birimi (secondary storage)** – kalıcı bir şekilde bilgilerin depolandığı, ana hafızanın uzantısı olan depolama birimidir
- ❑ **Manyetik diskler (magnetic disks)** – manyetik kayıt meteryaliyle kaplı sert metal veya cam tabakalar
 - ❑ Disk yüzeyi genellikle mantıksal olarak **izlere (tracks)** bölünür
 - ❑ Her bir iz **sektörlere (sectors)** bölünür
 - ❑ **Disk denetleyicisi (disk controller)** bilgisayar ile cihaz arasındaki mantıksal etkileşimi sağlar

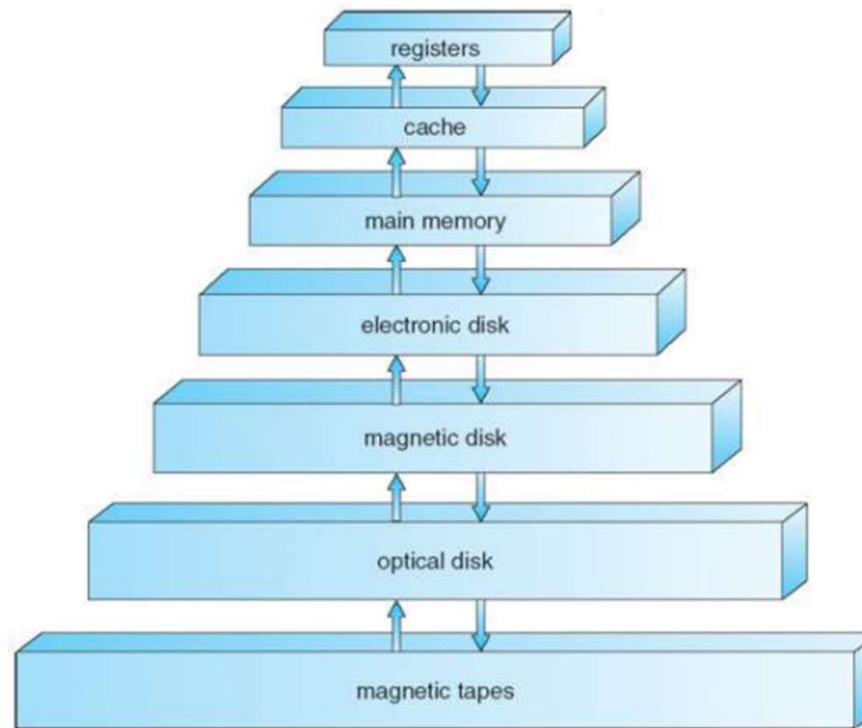
Depolama Birimi Hiyerarşisi

- ❑ Depolama birimlere hiyerarşik bir şekilde organize edilirler
 - ❑ Hız (Speed)
 - ❑ Maliyet (Cost)
 - ❑ Gelgeçlik (volatility)
- ❑ **Ön belleğe alma (caching)** – bilgiyi daha hızlı olan depolama birimine geçici olarak alma işlemidir
- ❑ Ana bellek ikincil depolama birimi için en son **ön bellek (cache)** birimidir

Depolama Birimi Hiyerarşisi

- ❑ Depolama birimlere hiyerarşik bir şekilde organize edilirler
 - ❑ Hız (Speed)
 - ❑ Maliyet (Cost)
 - ❑ Gelgeçlik (volatility)
- ❑ **Ön belleğe alma (caching)** – bilgiyi daha hızlı olan depolama birimine geçici olarak alma işlemidir
- ❑ Ana bellek ikincil depolama birimi için en son **ön bellek (cache)** birimidir

Depolama Cihazı Hiyerarşisi



Ön Belleğe Alma

- ❑ Bir bilgisayarda pek çok seviyede (donanım, işletim sistemi, yazılım) gerçekleştirilen önemli bir prensip
- ❑ Kullanılan bilgi yavaş depolama biriminden hızlı depolama birimine kopyalanır
- ❑ Aranılan bilgi öncelikle daha hızlı depolama biriminde mi (ön bellek) kontrol edilir
 - ❑ Eğer oradaysa, bilgi direk ön bellekten alınır (hızlı)
 - ❑ Eğer değilse, ön belleğe alınır ve oradan kullanılır
- ❑ **Ön bellek**, ön belleğe alınacak bilgiden daha küçüktür
 - ❑ Ön bellek yönetimi önemli bir tasarım problemidir
 - ❑ Ön bellek boyutu ve **yenileme politikası (replacement policy)**

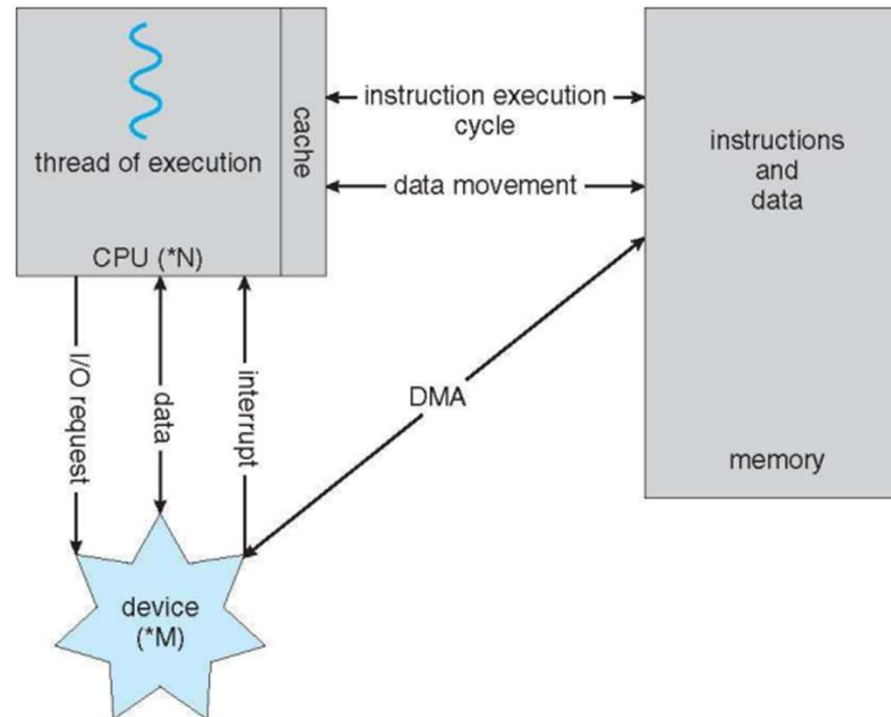
Tekli veya Çoklu İşlemciler

- ❑ Pek çok sistem tek bir genel amaçlı işlemci kullanır (örn: gömülü sistemler).
 - ❑ Aynı zamanda, pek çok sistem de özel amaçlı işlemciler kullanır
- ❑ **Çokişlemcili sistemler (multiprocessors systems)** giderek yaygınlaşmakta ve önem kazanmaktadır
 - ❑ **Paralel sistemler (parallel systems)** ve sıkıca bağlantılı sistemler (tightly-coupled systems) olarak da bilinirler
 - ❑ Avantajlar
 1. Artan **üretilebilirlik (throughput)**
 2. Ekonomik olarak katlanma (economy of scale)
 3. Artan **güvenilirlik (reliability)** – graceful degradation veya fault tolerance

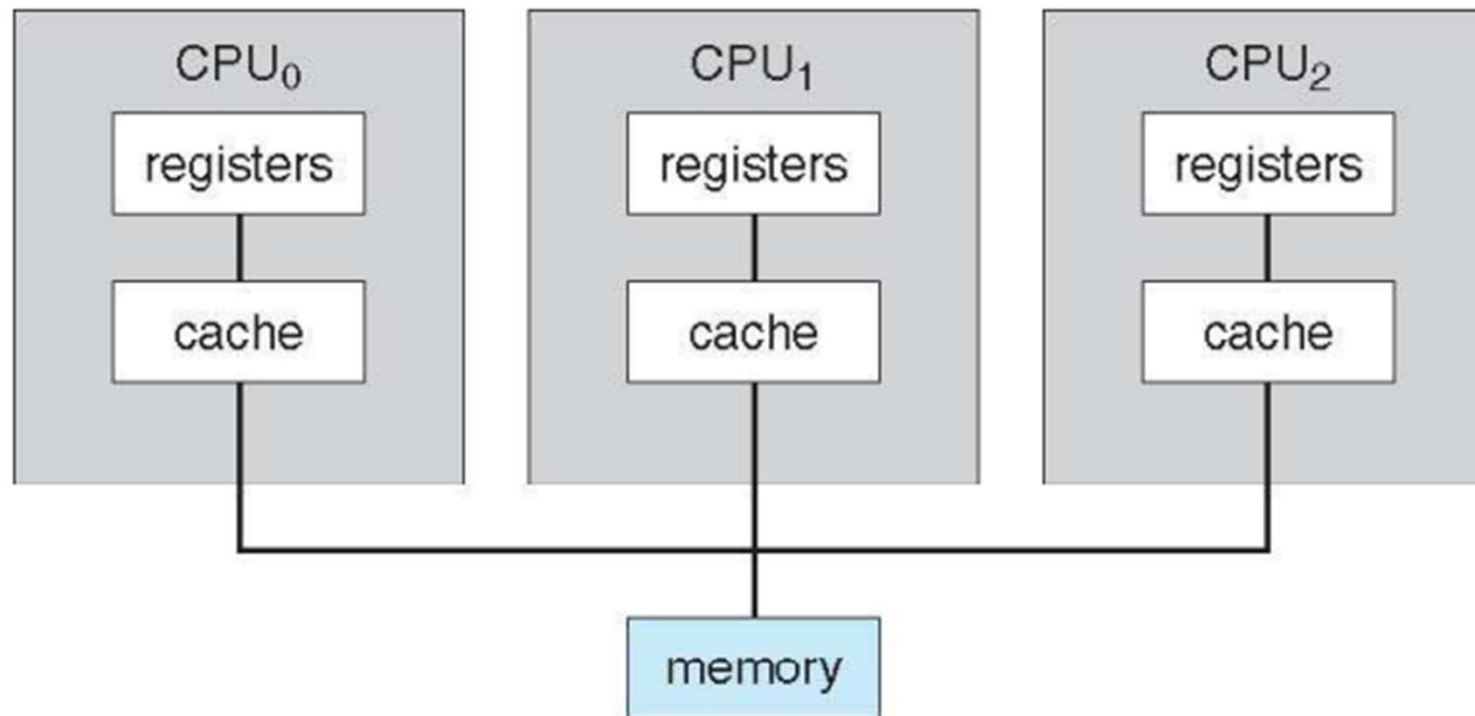
Çoklu İşlemciler

- ❑ İki farklı tür
 1. Asimetrik Çoklu İşlemciler (Asymmetric Multiprocessing)
 2. Simetrik Çoklu İşlemciler (Symmetric Multiprocessing)
- ❑ Asimetrik çoklu işlemciler – Görev dağıtan bir işlemci var, diğerleri görev bekliyor (master-slave)
- ❑ Asimetrik çoklu işlemciler özellikle ilk zamanlarda kullanılıyor
- ❑ Simetrik Çoklu İşlemciler (SMP) – tüm işlemciler her tür işi yapıyor

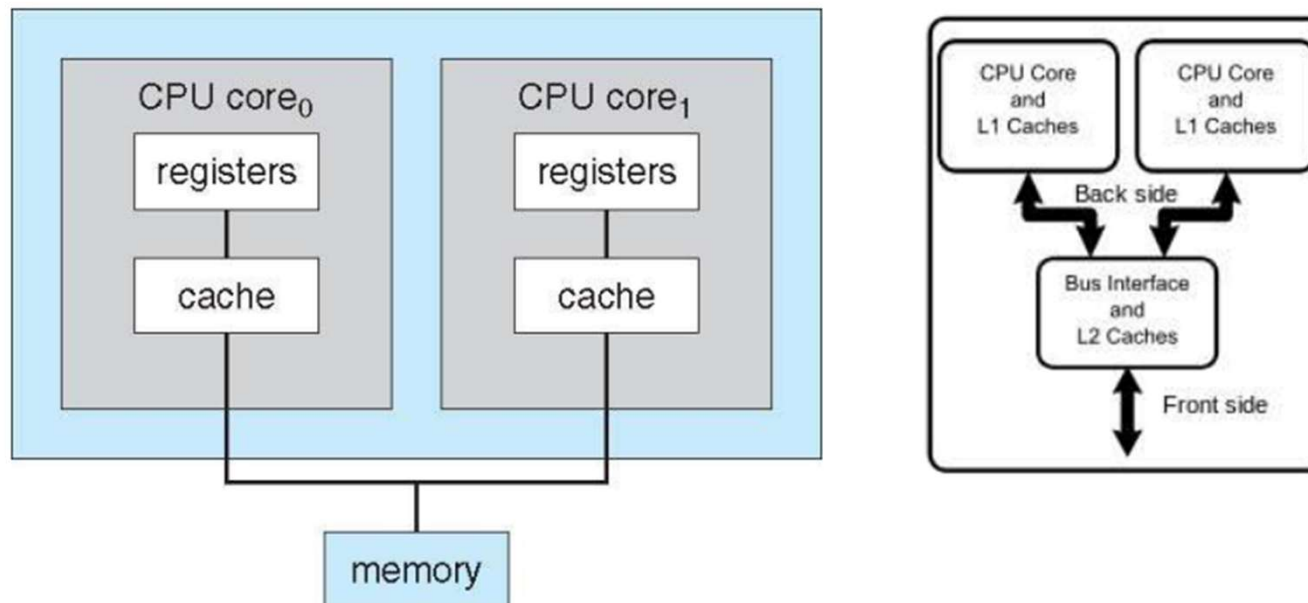
Modern Bilgisayarlar Nasıl Çalışır



Simetrik Çoklu İşlemci Mimarisi



Simetrik Çoklu İşlemci Mimarisi



Avantajlar/Dezavantajlar?

Küme Bilgisayarlar

- ❑ Küme Bilgisayarlar (clustered computers)
- ❑ Çoklu işlemcili sistemler gibi, fakat birden fazla sistem birlikte çalışıyor
 - ❑ Genellikle depolama birimi, storage-area network (SAN) ile paylaşılıyor
 - ❑ Arızalara dayanıklı yüksek bulunurluk (high-availability) sağlayan bir servis
 - ❑ **Asimetrik kümeleme (asymmetric clustering)** – bir tane gözlem makinası, diğerleri çalışıyor
 - ❑ **Simetrik kümeleme (symmetric clustering)** – birden fazla uygulama çalıştıran ve aynı zamanda birbirini gözlemleyen makinaya (node) sahip

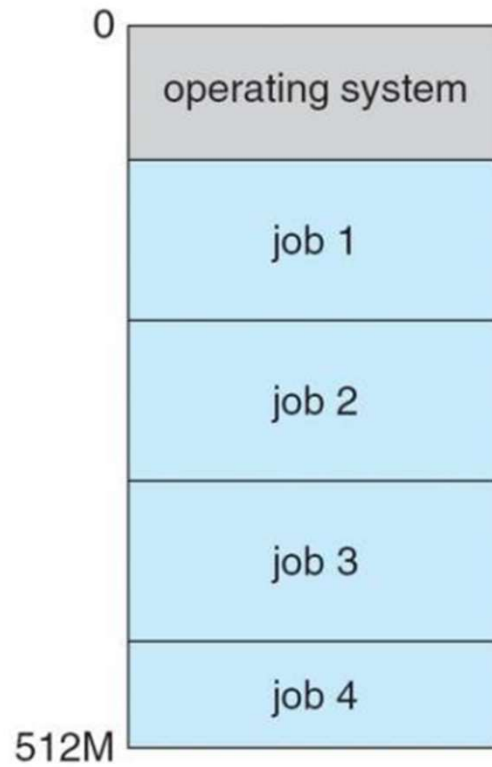
Yüksek Performanslı Hesaplama

- ❑ Bazı kümeler yüksek performanslı hesaplama **high-performance computing (HPC)** sağlıyor
- ❑ Uygulamalar **paralleleştirmeyi (parallelization)** kullanacak şekilde yazılmalı

Zaman Paylaşımı

- ❑ **Zaman Paylaşımı (timesharing veya multitasking)**, CPU'nun, işleri çalıştırırken, işler arasında çok hızlı geçiş sağlayarak kullanıcıya bilgisayarı interaktif (interactive) şekilde kullanıyormuş hissi vermesidir
 - ❑ **Cevap süresi (response time)** 1 saniyeden az olmalıdır
 - ❑ Her bir kullanıcı hafızada çalışan en az bir programa sahiptir
 - ❑ Eğer aynı anda birden fazla iş çalışmak için hazırsa ❑ **İşlemci zamanlaması (CPU scheduling)**
 - ❑ Eğer işlemler hafızaya sığmıyorsa, **değiş-tokuş işlemi (swapping)** işlemleri, çalıştırmak gerektiğinde hafızaya alır ya da gerektiğinde hafızadan çıkarır
 - ❑ **Sanal hafıza (virtual memory)** tümüyle hafızada bulunmayan işlemleri çalıştırmayı sağlar

Çok Programlı Sistemlerde Hafıza Dilimi



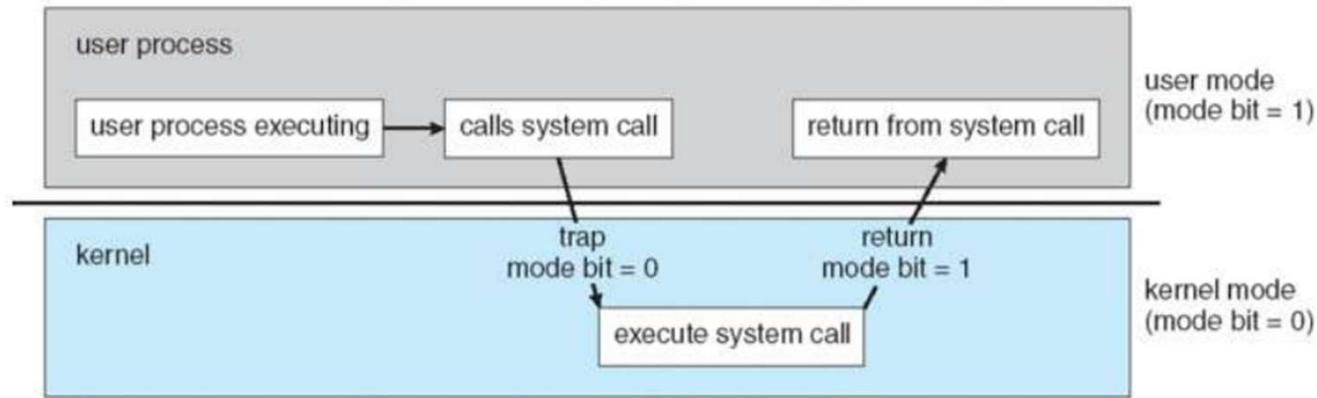
İşleme Sorunları

- ❑ Donanım tarafından kesinti gönderilebilir
- ❑ Yazılım hataları veya istekleri **tuzığa (exception veya trap)** neden olabilir
 - ❑ Sıfıra bölünme, işletim sistemleri servislerini çalıştırmaya kalkma
- ❑ Diğer işleme sorunları:
 - ❑ Sonsuz döngü
 - ❑ İşlemlerin birbirini değiştirmeye çalışması
 - ❑ İşlemlerin işletim sistemini değiştirmeye çalışması

Çift Modlu İşleme

- ❑ Çift-modlu işleme (dual-mode operation), işletim sistemini ve diğer sistem bileşenleri korumayı sağlar
 - ❑ Kullanıcı modu (user mode) ve çekirdek modu (kernel mode)
 - ❑ Donanım tarafından sağlanan mod biti (mode bit)
 - ❑ Sistemin kullanıcı kodu mu yoksa çekirdek kodu mu çalıştığını ayırt etmekte kullanılır
 - ❑ Bazı komutlar ayrıcalıklı (privileged) olarak tanımlıdır ve sadece çekirdek modunda çalıştırılabilirler
 - ❑ Sistem çağruları modu, çekirdek moduna çevirir.
 - ❑ Sistem çağrısı bittiğinde mod, kullanıcı moduna çevrilir

Kullanıcı Moddan Çekirdek Moda Geçiş



- ❑ **Zamanlayıcı (timer)** sonsuz döngülere ve işlemci kilitlemelerine engel olur
 - ❑ Belli bir zaman diliminden sonra kesme gönderilir
 - ❑ İşletim sistemi sayacı azaltır
 - ❑ Sayaç sıfırlandığında kesme oluşturulur
 - ❑ Zamanlayıcı program devreye girmeden sorun çıkaran işlem devre dışı bırakılır veya sonlandırılır

İşlem Yönetimi

- ❑ İşlem (process) çalışmakta olan programdır
- ❑ Program pasif bir şeyken, işlem aktif bir şeydir
- ❑ İşlemler görevlerini yerine getirmek için kaynaklara ihtiyaç duyarlar
 - ❑ CPU, hafıza, I/O, dosyalar
 - ❑ Başlangıç verisi
- ❑ İşlemin sonlandırılması kullanılan kaynakların sisteme iade edilmesini gerektirir

İş Parçacığı Yönetimi

- ❑ **İş parçacığı (thread)** bir program çalışırken aynı anda yapılması gereken başka işler varsa bunları çalıştırmak için kullanılır
- ❑ Tek iş parçacıklı (single-threaded) işlemler, çalıştırılacak bir sonraki komutun hafızadaki konumunu belirten tek bir **program sayacına (program counter)** sahiptir
 - ❑ İşlem sonlanana kadar, komutları tek tek sırayla çalıştırır
- ❑ **Çok iş parçacıklı (multi-threaded)** işlemler her bir iş parçacığı için ayrı bir program sayacına sahiptir
- ❑ Tipik olarak sistemlerde, pek çok işlem, birkaç kullanıcı ve pek çok işletim sistemi işlemi aynı anda bir veya birden fazla işlemcide çalıştırılır
- ❑ **Aynı anda kullanım (concurrency)** işlemcilerin birden fazla işlem veya iş parçacığı arasında ortak kullanımını gerektirir

İşlem Yönetim Faaliyetleri

İşletim sisteminin, işlem yönetimi ile ilişkili faaliyetleri:

- Kullanıcı ve sistem işlemlerinin oluşturulması ve bitirilmesi
- İşlemlerin duraklatılması ve devam ettirilmesi
- İşlemlerin senkronizasyonu için mekanizmalar sağlanması
- İşlemlerin birbiri ile iletişim kurabilmesi için mekanizmalar sağlanması
- Kilitlenmelerin (deadlock)** sağlıklı yönetilmesi için mekanizmalar sağlanması

Hafıza Yönetimi

- ❑ Tüm veriler işlem öncesi ve sonrası hafızadadır
- ❑ Komutların çalıştırılabilmesi için hafızada olması gerekir
- ❑ Hafıza yönetimi
 - ❑ Neyin hafızada olması gerektiğine karar verir
 - ❑ Hedefi, işlemci kullanımını ve kullanıcılara verilen yanıtları optimize etmektir

Hafıza Yönetim Faaliyetleri

- Hafızanın hangi bölümlerinin kim tarafından kullanıldığını takip etmek
- Hangi işlemlerin ve verilerin hafızaya alınacağına ya da hafızadan çıkarılacağına karar vermek
- Gerektiğinde yeni hafıza alanı ayırmak ya da kullanılmış alanları iade etmek

Depolama Birimi Yönetimi

- ❑ İşletim sistemi, depolama birimleri için tek ve mantıksal arayüz sunar
 - ❑ Fiziksel özellikleri mantıksal depolama birimine soyutlar: **dosya (file)**
 - ❑ Tüm birimler cihaz tarafından kontrol edilir (i.e., disk, DVD)
 - ❑ Değişken özellikler: erişim hızı, kapasite, veri transfer hızı, erişim yöntemi (sırayla veya direk)
- ❑ **Dosya sistemi yönetimi**

Dosya Sistemi Yönetimi

- ❑ Dosyalar dizinler kullanılarak organize edilir
- ❑ Pek çok sistemde dizinlere veya dosyalara erişim kontrol edilmelidir:
erişim kontrolü (access control)
- ❑ Dosya sistemi ile ilişkili işletim sistemi aktiviteleri
 - ❑ Dosya ve dizinlerin oluşturulması veya silinmesi
 - ❑ Dosyaların veya dizinlerin değiştirilmesi için mekanizmanın sağlanması
 - ❑ Dosyaların ikincil depolama birimi ile eşleştirilmesi
 - ❑ Dosyaların kalıcı depolama birimlerine yedeklenmesi

Mass-Storage Management

- ❑ Genellikle diskler, hafızaya sığmayan verileri ya da uzun süre tutulacak verileri tutmakta kullanılır
- ❑ Verilerin tutarlı yönetimi çok önemlidir
- ❑ Bilgisayarın genel hızı disk alt sistemi ve algoritmalarının performansına çok bağlıdır
- ❑ İlgili işletim sistemi faaliyetleri:
 - ❑ Boş alan yönetimi
 - ❑ Depolama alanı ayrımı
 - ❑ Disk zamanlaması
- ❑ Bazı depolama birimlerinin hızlı olması gerekmez
 - ❑ CD, DVD, Manyetik teypler
 - ❑ Gene de yönetilmelidir
 - ❑ WORM (write-once, read-many-times) ve RW (read-write) erişim modlarında çalışabilirler

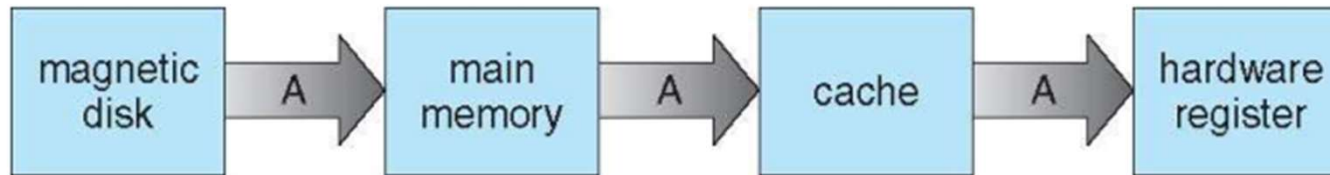
Depolama Birimi Performansları

- Depolama birimi seviyeleri arasında bilgi aktarımı, **kullanıcının isteğine bağlı** ya da **kullanıcı isteğinden bağımsız** gerçekleşebilir

Level	1	2	3	4
Name	registers	cache	main memory	disk storage
Typical size	< 1 KB	> 16 MB	> 16 GB	> 100 GB
Implementation technology	custom memory with multiple ports, CMOS	on-chip or off-chip CMOS SRAM	CMOS DRAM	magnetic disk
Access time (ns)	0.25 – 0.5	0.5 – 25	80 – 250	5,000.000
Bandwidth (MB/sec)	20,000 – 100,000	5000 – 10,000	1000 – 5000	20 – 150
Managed by	compiler	hardware	operating system	operating system
Backed by	cache	main memory	disk	CD or tape

A Tamsayısının Diskten Yazmaa Aktarımı

- ❑ ok iřlemli ortamlar, en gncel deęeri kullanmak konusunda dikkatli olmalıdır (depolama hiyerarřisinin neresinde tutuluyorsa tutulsun)



ok iřlemcili sistemlerde **n bellek tutarlılıęı** donanım seviyesinde saęlanmalı ve tm iřlemciler en gncel deęere sahip olmalıdır

- ❑ Daęıtık ortamlarda durum daha da karmařıktır
 - ❑ Verinin birden fazla kopyası bulunabilir

I/O Alt Sistemi

- ❑ İşletim sisteminin amaçlarından biri donanım cihazlarının karmaşıklıklarını kullanıcıdan gizlemektir
- ❑ I/O alt sisteminin sorumlulukları:
 - ❑ I/O işlemlerinin hafıza yönetimini yapmak
 - ❑ **Tampon bellek işlemleri (buffering)** – veriyi bir yerden diğer yere aktarırken geçici olarak saklamak
 - ❑ **Ön bellek işlemleri (caching)** – veriyi geçici olarak daha hızlı depolama birimine aktarmak
 - ❑ **Kuyruklama (spooling)** – bir işin çıktısını diğer işin girdisi haline getirmek
 - ❑ Genel cihaz sürücüsü arayüzü
 - ❑ Özel donanım cihazları için sürücüler

Koruma ve Güvenlik

- ❑ **Koruma (protection)** – İşlemlerin veya kullanıcıların herhangi bir kaynağa erişiminin işletim sistemi tarafından kontrol edilmesi
- ❑ **Güvenlik (security)** – sistemin içerden ve dışardan gelen saldırılara karşı savunulması
 - ❑ Geniş kapsamlı: DoS saldırıları, virüsler, solucanlar, kimlik bilgileri hırsızlığı
- ❑ Sistemler öncelikle kullanıcıları, kimin ne yapabileceğine göre sınıflandırılırlar
 - ❑ **Kullanıcı adı (user IDs, security IDs)** her kullanıcı için isim ve ilişkili numarayı içerir
 - ❑ Kullanıcı adı daha sonra, erişim kontrolü amacıyla, kullanıcının sahip olduğu tüm dosya ve işlemlerle ilişkilendirilir
 - ❑ **Grup adı (group ID)** da benzer şekilde bir grup kullanıcıyı belli işlem ve dosyalarla ilişkilendirmek ve erişim kontrolü sağlamak amacıyla kullanılır

Bilgisayar Ortamları

❑ Geleneksel bilgisayarlar

- ❑ Sınırlar zamanla değişiyor

❑ Ofis Ortamı

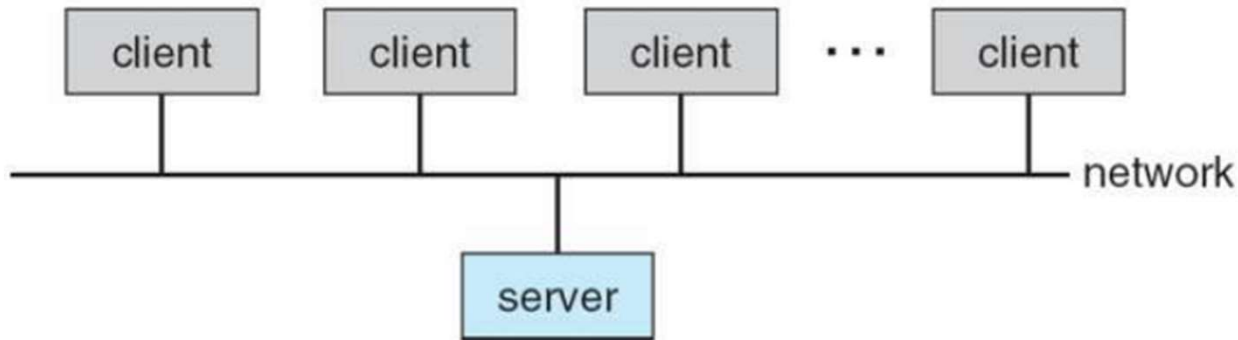
- ❑ **Terminaler** ana bilgisayarlara bağlı ve ana bilgisayar kaynakları kullanıcılar arasında paylaştırılıyor
- ❑ Kişisel bilgisayarlar bir ağa bağlı,
- ❑ Şimdi, **portallar** ile aynı kaynaklara yerel ağ üzerinden veya uzaktan erişim mümkün

❑ Ev Ortamı

- ❑ Önceden bağımsız bilgisayarlar
- ❑ Daha sonra modemlerle Internet'e bağlılar
- ❑ Şimdi, birbirlerine bağlı ve **güvenlik duvarına (firewall)** sahip

İstemci-Sunucu Sistemleri

- ❑ Zamanla akıllı kişisel bilgisayarlar, akılsız terminallerin yerini aldı
- ❑ Şu an pek çok sistem **sunucu (server)** olarak kullanılıyor, ve istemcilerin (clients) isteklerine cevap veriyor
 - ❑ **İşlem-sunucuları (compute-server)** istemcilere çeşitli servisler sağlayan bir arayüz sunar (örn. veritabanı)
 - ❑ **Dosya sunucuları (file-server)** istemcilere dosyaları kaydetmeyi ve indirmeyi sağlayan bir arayüz sunar



Uçtan-Uca Sistemler

- ❑ Uçtan-uca sistemler (Peer-to-Peer Systems, P2P)
- ❑ Dağıtık sistemlerin bir başka örneği
- ❑ P2P istemci ve sunucu arasında ayırım yapmaz
 - ❑ Her bir sistem bir uç olarak ele alınır
 - ❑ Her bir uç istemci, sunucu veya iki şekilde birden davranabilir
 - ❑ Uçlar öncelikle bir P2P ağına bağlanmalıdır
 - ❑ Kendini bu ağdaki merkezi kayıt sistemine kaydetmelidir,
- veya
 - ❑ Keşif protokolü (discovery protocol) ile istekte bulunmalı veya daha önce bulunulan istekleri karşılamalıdır
- ❑ Örnek: Napster ve Gnutella

Web Tabanlı Sistemler

- ❑ Artık PC'ler sunucu olarak kullanılabilir
- ❑ Giderek daha çok cihaz Web'e bağlanıyor
- ❑ Web trafiğini yönetmek için yeni tür sunucular ortaya çıkıyor. Örnek: **yük dengeleyiciler (load balancers)**
- ❑ Yeni işletim sistemleri (örn: Linux, Windows 7) artık sunuculara ait özellikleri de barındırıyor ve hem istemci hem de sunucu olabilir

Açık Kodlu İşletim Sistemleri

- ❑ Bu işletim sistemleri, **kapalı makine formatı (closed-source)** yerine **kaynak kod (source-code)** formatında sunuluyor
- ❑ **Free Software Foundation (FSF)** ile başladı - “copyleft” **GNU Public License (GPL)**
- ❑ Örnekler: **GNU/Linux, BSD UNIX (Mac OS X işletim sistemi temeli), ve Sun Solaris**