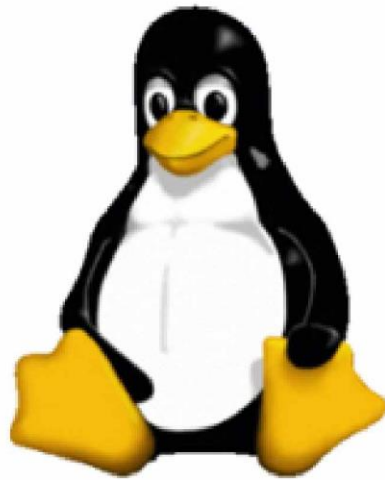




BİLGİSAYAR MÜHENDİSLİĞİNE GİRİŞ

DERS 6
İşletim Sistemleri

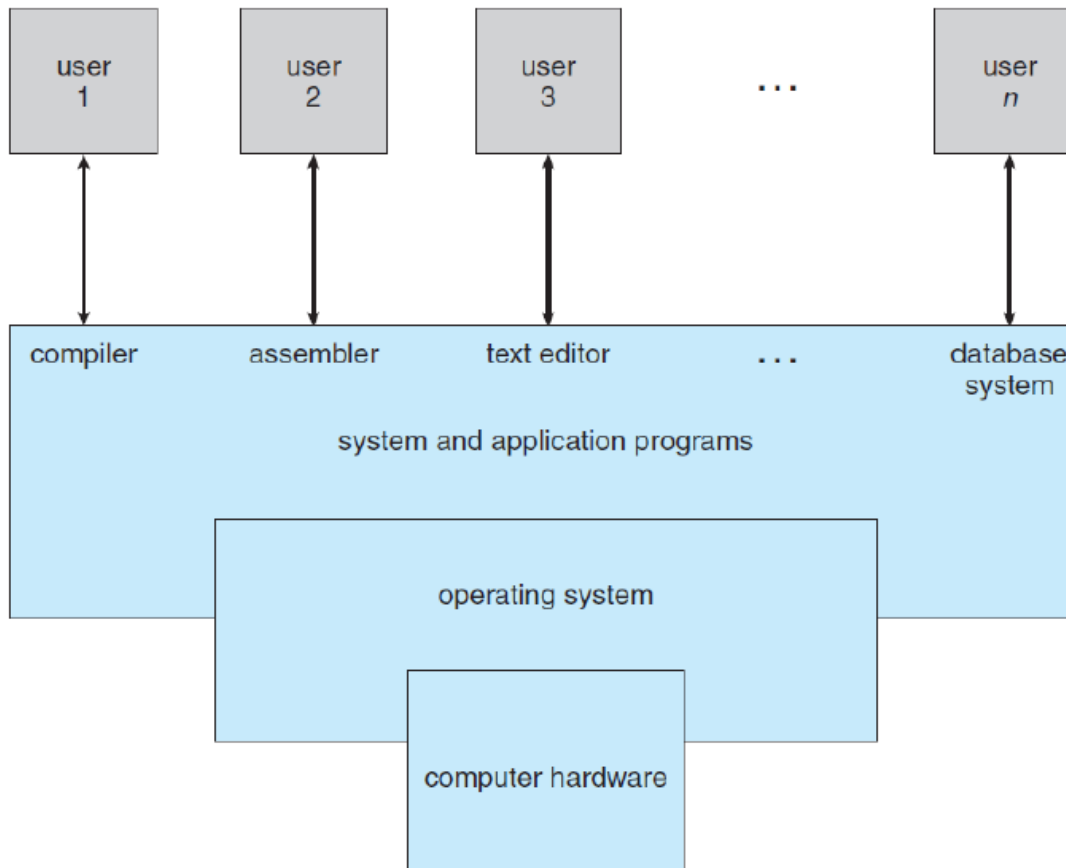


İşletim sistemi ne iş yapar?

- **İşletim sistemi bilgisayar donanımını yöneten bir programdır.**
- Kullanıcı ile bilgisayar donanımı arasında aracı olarak görev yapar.
- Bazı işletim sistemlerinde (server) **etkinlik**, bazılarında ise **kullanılabilirlik** (mobil, client) önemlidir.
- **Bir bilgisayar sistemi genel olarak 4 bileşene ayrılabilir:**
 - Donanım
 - İşletim sistemi
 - Uygulama programları
 - Kullanıcılar
- **Donanım** (CPU, memory, I/O cihazları) temel kaynakları sağlar.
- **Uygulama programları** (kelime işlemciler, derleyiciler, Web tarayıcılar) bu kaynakların kullanıcı problemlerinde nasıl kullanılacağını belirler.

İşletim sistemi ne iş yapar?

- İşletim sistemi, farklı kullanıcılar için farklı **kaynakların** uygun kullanımını koordine eder.





Arayüz

Command Line Interface (CLI) or
Command Line User Interface (CUI)

```
Komut İstemi

Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. Tüm hakları saklıdır.

C:\Users\BEYZA>cd desktop

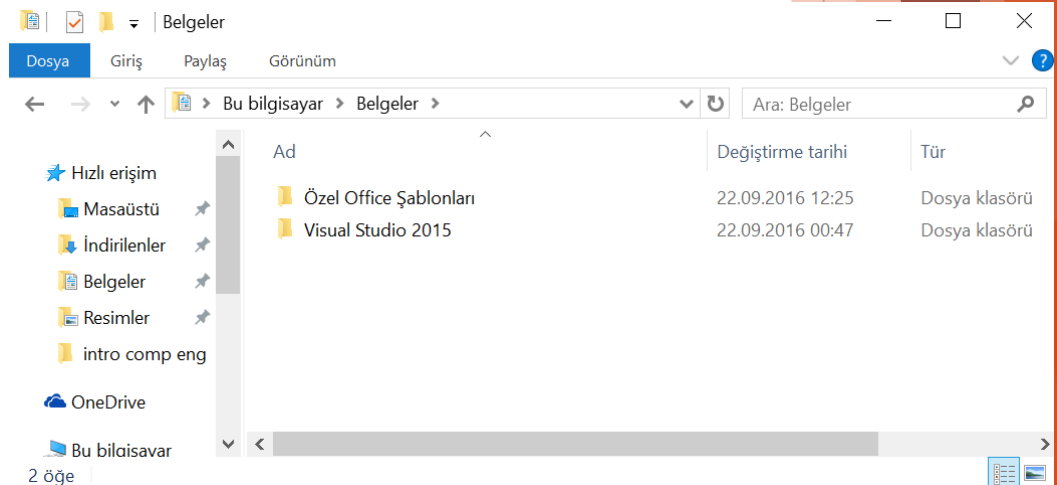
C:\Users\BEYZA\Desktop>dir
Volume in drive C is BOOTCAMP
Volume Serial Number is D608-4315

Directory of C:\Users\BEYZA\Desktop

20.10.2016  21:26    <DIR>          .
20.10.2016  21:26    <DIR>          ..
27.09.2016  19:56             103 iletisim.txt
20.10.2016  21:24    <DIR>          intro comp eng
28.09.2016  21:09             134 klup oneri.txt
11.10.2016  20:01              7 newTech.txt
               3 File(s)                244 bytes
               3 Dir(s) 191.614.713.856 bytes free

C:\Users\BEYZA\Desktop>
```

Graphical User Interface (GUI)



İşletim sistemi ne iş yapar?

Kullanıcı açısından bakış

- Kullanıcı bakışı **kullanılan arayüze bağlı olarak değişmektedir.**
- **Çoğu kullanıcı,** monitöre, klavyeye, fareye ve sistem birimine sahip bir **kişisel bilgisayar kullanır.**
- Bu durumda, **işletim sistemi** çoğunlukla **kolay kullanım için tasarlanır, kaynakların** nasıl paylaşıldığı ve **verimliliği,** çok kullanıcı yerine **tek kullanıcıya göre tasarlanır.**
- Diğer bir durumda ise, **çok sayıda kullanıcı bir ana bilgisayara bağlanır ve kaynakları paylaşırlar.**
- Burada, işletim sistemi **kaynak kullanım oranını maksimize edecek şekilde tasarlanır.**
- **Mobil cihazlar ve gömülü sistemler (ev cihazları, otomobil) için** tasarlanan işletim sistemlerinin de kendine özgü özellikleri vardır.

İşletim sistemi ne iş yapar?

Sistem açısından bakış

- Bilgisayar açısından işletim sistemi, **donanımla çok yakından ilişkiye sahip olan bir programdır.**
- Bu açıdan işletim sistemi, **kaynak** (CPU time, hafıza, depolama birimi, I/O cihazları, ...) **kullanımını planlayan programdır.**
- İşletim sistemi, **I/O cihazlarını ve kullanıcı programlarını kontrol eder** ve programların çalışması sırasında **hataları önlemeye yönelik işlemleri yönetir.**

İşletim sistemi ne iş yapar?

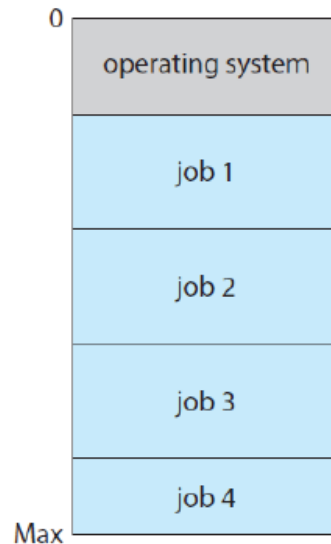
İşletim sistemi tanımı

- **Bilgisayarların temel amacı**, kullanıcı programlarının çalıştırılması ve kullanıcı problemlerinin kolay ve hızlı bir şekilde çözülmesidir.
- Bilgisayar donanımlarının tek başına kullanımı çok zordur, bu yüzden uygulama yazılımları geliştirilir.
- Yaygın kabul edilen tanımlamada, **işletim sistemi** bilgisayarda sürekli çalışan programdır ve **kernel (çekirdek)** olarak adlandırılır.
- Mobil işletim istemleri sadece kernel'a sahip değildir, **middleware**'e de sahiptirler.
- **Middleware**, uygulama geliştiricilere ek servisler sağlayan **framework (platform)** yazılımlarıdır.
- Apple iOS ve Google Android, **middleware** yazılımları ile **veritabanı**, **multimedya** ve **grafik** desteği sağlayan mobil işletim sistemleridir.

İşletim sistemi; bilgisayarlarda çalışan, **donanım** ve yazılım kaynaklarını yöneten ve çeşitli **uygulama** yazılımları için yaygın servisleri sağlayan bir yazılımlar bütünüdür.

İşletim sistemi yapısı

- İşletim sistemi, programların çalıştırılması için ortam sağlamaktadır.
- İşletim sistemleri birden çok programı çalıştırabilir (multiprogramming).
- Multiprogramming çalışabilen işletim sistemi çok sayıda işi aynı anda hafızada tutar.
- Tüm işler disk üzerindeki job pool içinde tutulur.)



İşletim sistemi yapısı

- Multiprogramming işletim sistemi bir işi alır ve çalıştırmaya başlar.
- Çalışan işte **bekleme olduğunda** başka bir işe geçiş yaparak çalışmaya devam eder.
- **Multitasking (time sharing)** işletim sistemlerinde CPU işler arasında çok hızlı geçişler yapar. (Geçiş için işte bekleme oluşması gerekli değildir.)
- Multitasking işletim sistemlerinde **kullanıcı** herhangi bir iş ile etkileşime geçebilir. **Tepki süresinin çok kısa olması gereklidir!**
- Hafızaya yüklenen ve çalıştırılmakta olan programa **process** denilir.
- Eğer hafızada ayrılan yerden daha çok sayıda iş hafızaya alınmak için hazır ise, **hafızaya alınacak olanı seçmeye job scheduling** denir.
- Aynı anda hafızada birden fazla iş hazır ise, **hangisinin ilk önce çalışacağına karar vermeye CPU scheduling** denilmektedir.

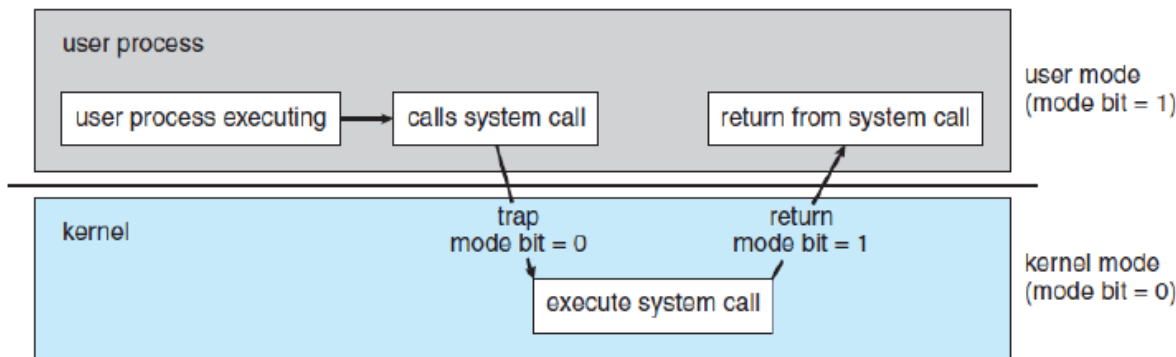
İşletim sistemi işlemleri

- Modern işletim sistemleri, kesilmelerle yönetilirler (**interrupt driven**).
- Eğer çalışan process yoksa, hiçbir I/O cihazı servis sağlamıyorsa, kullanıcılardan etkileşim yoksa, işletim sistemi bekleme durumundadır ve hiçbir iş yapmaz.
- Bir trap (veya exception), yazılım tarafından üretilen interrupt'tır ve işletim sisteminin iş gerçekleştirmesini sağlar.
- Bir işletim sisteminde çalışan programlardan birisi hata ürettiğinde sadece o programın etkilenmesi istenir.
- Ancak, bazı durumlarda diğer programların çalışma hızı etkilenebilir, verileri değişebilir veya işletim sisteminin kendisi bile çalışmaz hale gelebilir.
- İyi tasarlanmış işletim sistemleri bu şekilde hatalı programların (**malicious**) diğerlerini etkilemesini engeller.

İşletim sistemi işlemleri

Dual mode ve multimode işlem

- İşletim sisteminin doğru çalışmasını sağlamak için, **işletim sistemi kodu ile kullanıcı programının kodunun ayırt edilmesi gereklidir.**
- **Mode bit** kullanılarak, kullanıcı modu (**user mode = 1**) ve kernel modu (**supervisor, system, privileged = 0**) ayrımı (**dual mode**) yapabilirler.



- Sistem boot edildiğinde kernel moddadır ve uygulama programı çalışmaya başlayınca user moda geçer.
- Birden fazla bit kullanılarak **multimode** oluşturulabilir (test mode, ...).

İşletim sistemi işlemleri

Timer

- Bir kullanıcı programının **sonsuz döngüye girmesi** veya hata oluşması durumunda, **sistem servislerini çağıramaması sonucunda işletim sistemine dönülemez.**
- Bu sorunu gidermek için **timer** kullanılır.
- Timer her programa geçildiğinde set edilir ve aşağıya doğru sayar.
- **Timer 0 değerine ulaşınca kontrol işletim sistemine alınır.**
- İşletim sistemleri her program için belirlenen **timer süresini sabit veya değişken alabilmektedirler.**

Process yönetimi

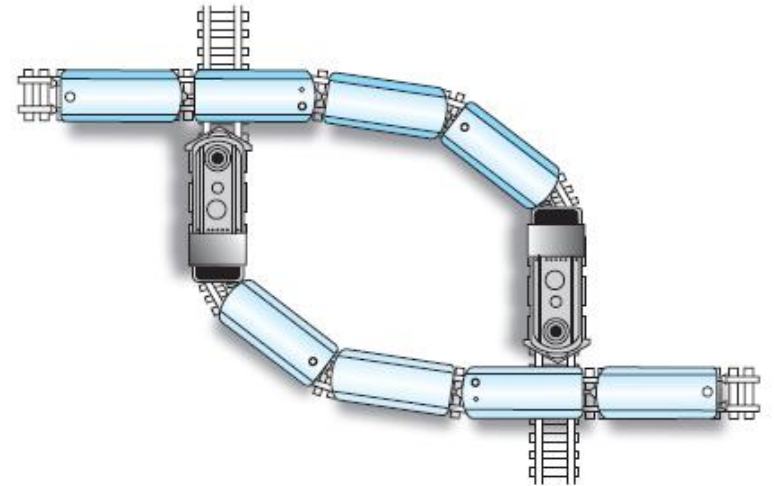
- CPU tarafından çalıştırılmakta olan program **process** olarak adlandırılır.
- Bir process yapması gereken işi tamamlamak için, **CPU süresine, hafızaya, dosyalara, I/O cihazlarına ihtiyaç duyar.**
- Process çalıştığı sürece bu kaynaklardan ihtiyaç duyduğunu kullanır.
- Çalışması sonlanınca işletim sistemi ayrılmış **kaynakları serbest bırakır.**
- Bir **program pasif** varlıktır (**passive entity**), bir **process** ise **aktif** varlıktır (**active entity**).
- **Program counter (PC)**, CPU içerisinde register'dır ve sonraki çalıştırılacak komutun adresini tutar.
- **Single-threaded process** bir PC'ye sahiptir, **multithreaded process** birden çok PC'ye sahiptir.

Process yönetimi

- **Bir işletim sistemi process yönetiminde aşağıdaki işlerden sorumludur:**
 - CPU üzerindeki process ve thread'lerin zamanlaması,
 - Kullanıcı ve sistem process'lerinin oluşturulması ve silinmesi,
 - Process'lerin askıya alınması ve devam ettirilmesi,
 - Process'lerin senkronizasyonu,
 - Process'lerin haberleşmesi.

Kilitlenme

- ▶ İki veya daha fazla işlemin birbirlerinin kullandıkları kaynakları bekledikleri için çalışmalarının durmasıdır.
- ▶ Örnek olarak bir işlem yazıcıya erişim sağlamışken CD-ROM'u bekliyorsa
- ▶ Diğer işlem de CD-ROM'u kullanırken yazıcıyı bekliyorsa kilitlenme gerçekleşir.
- ▶ Bu durumda genellikle işlemlerden biri öldürülür.



Memory yönetimi

- Hafıza, modern işletim sistemlerinde işlemlerin gerçekleşmesinde temel elemandır.
- **CPU, tüm programları hafıza üzerinden çalıştırır.**
- CPU, disk üzerindeki bir program parçasına ihtiyaç duyduğunda, I/O çağrısı ile önce hafızaya aktarır.
- Genel amaçlı bilgisayarlar **CPU verimliliğini artırmak** için birden **çok programı hafızada tutarlar** ve hafıza yönetimi gerçekleştirirler.
- İşletim sistemi **hafıza yönetiminde aşağıdaki işlerden sorumludur:**
 - **Hafızanın hangi kısmının kullanıldığının** ve kimin tarafından kullanıldığının izlenmesi,
 - **Hangi process'in** (veya process parçasının) **hafızaya alınacağına** veya hafızadan atılacağına **karar verilmesi**,
 - Hafızadaki **boş alanların tahsis edilmesi** veya serbest bırakılması.

Storage yönetimi

Dosya sistemi yönetimi

- Her depolama birimi, hız, kapasite, veri aktarım oranı ve erişim yöntemi gibi farklı özelliklere sahiptir.
- İşletim sistemi, **depolama biriminin özelliklerini soyutlamak için** mantıksal depolama birimi olarak **file (dosya)** tanımlar.
- Dosyalar, sayısal, alfabetik, alfanümerik veya binary veri bulundurabilir.
- Dosyaya birden fazla kullanıcı erişebilir. **Her kullanıcı için erişim denetiminin (okuma, yazma, ekleme) yapılması gereklidir.**
- İşletim sistemi dosya yönetiminde aşağıdaki işlerden sorumludur:
 - Dosya oluşturma ve silme,
 - Dizin oluşturma ve silme,
 - Dosya ve dizin manipülasyon işlemleri.

Storage yönetimi

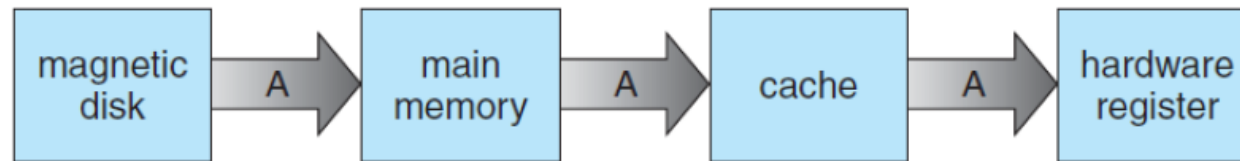
Mass-storage yönetimi

- Tüm programlar hafızaya alınmadan önce disk üzerinde saklanır.
- Disk yönetimi işletim sisteminin temel görevlerindendir.
- Manyetik tape, CD ve DVD sürücüler, üçüncül (tertiary) depolama cihazlarıdır.
- Bu cihazlar, WORM (write-once, read-many-times) veya RW (read-write) olarak farklı özelliklere sahip olabilirler.
- İşletim sistemi disk yönetiminde aşağıdaki işlerden sorumludur:
 - Boş alan yönetimi,
 - Depolama alanı tahsisi,
 - Disk kullanım zamanlaması.

Storage yönetimi

Cache bellek

- Cache bellek, hafızadan daha hızlı ve CPU'ya daha yakın saklama birimidir.
- CPU, bir veriye ihtiyaç duyduğunda hafızadan alır ve bir kopyasını cache bellek üzerine aktarır.
- Tekrarlı isteklerde cache bellek üzerindeki kullanır.



- Cache bellek kapasitesi çok küçük olduğundan yönetimi çok önemlidir.
- Cache bellekte tutulacak veya atılacak verilerin belirlenmesi için replacement algoritmaları kullanılır.

Storage yönetimi

Cache bellek

- Cache bellekler **hafızadan çok küçük kapasiteye** sahip olan ancak **register'lerden daha fazla kapasiteye** sahip olan depolama birimleridir.
- Cache belleklerde erişim **adrese göre**, **register'larda isme göre** yapılır.

Level	1	2	3	4	5
Name	registers	cache	main memory	solid state disk	magnetic disk
Typical size	< 1 KB	< 16MB	< 64GB	< 1 TB	< 10 TB
Implementation technology	custom memory with multiple ports CMOS	on-chip or off-chip CMOS SRAM	CMOS SRAM	flash memory	magnetic disk
Access time (ns)	0.25 - 0.5	0.5 - 25	80 - 250	25,000 - 50,000	5,000,000
Bandwidth (MB/sec)	20,000 - 100,000	5,000 - 10,000	1,000 - 5,000	500	20 - 150
Managed by	compiler	hardware	operating system	operating system	operating system
Backed by	cache	main memory	disk	disk	disk or tape

- Birden fazla işlemcili sistemlerde cache tutarlılığının (cache coherence) sağlanması zorunludur.

Storage yönetimi

I/O sistemleri

- İşletim sistmlerinin amaçlarından birisi de **donanımların özelliklerinden kullanıcıyı soyutlamaktır.**
- Sadece **device driver** kendisine atanmış olan **cihazın özelliklerini bilir.**
- **I/O sistemi aşağıdaki bileşenlere sahiptir:**
 - Hafıza yönetim bileşeni (buffering, caching ve spooling),
 - Device driver arayüzü,
 - Donanımlar için driver.

Koruma ve Güvenlik

- Bir bilgisayar sistemi **birden fazla kullanıcının erişimine açıksa ve birden çok işlemcinin eş zamanlı işlemine izin veriyorsa, verilere erişimin düzenlenmesi zorunludur.**
- İşlemciye, dosyalara, hafıza segment'lerine ve diğer kaynaklara sadece **yetkisi olan processlerin erişimine izin verilmelidir.**
- Bilgisayar sistemindeki kaynaklara kullanıcıların veya process'lerin **erişiminin denetlenmesine koruma (protection) denilmektedir.**
- Protection oluşabilecek hataları arayüzde iken algılar ve sistemin **güvenilirliğini artırır.**
- Koruma altındaki bir sistem yetkili ve yetkisiz kullanıcıları birbirinden ayırt eder.

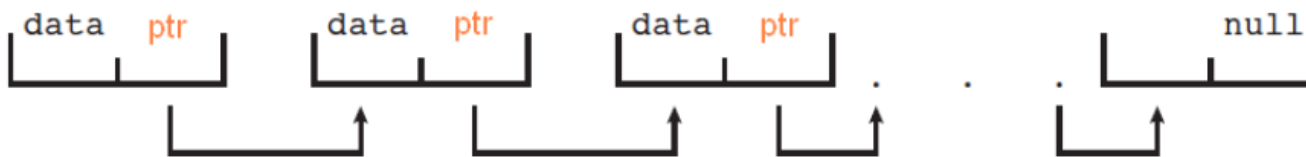
Koruma ve Güvenlik

- Bir sistem yeterli korumaya sahip olsa da **hatalara ve uygun olmayan erişimlere elverişli olabilir.**
- Örneğin, sisteme erişim yetkisi olan bir **kişinin bilgileri çalınabilir ve bilgileri silinebilir, kopyalanabilir, dosyaları kalıcı hasara uğrayabilir.**
- **Güvenlik (security),** bir sistemi dışarıdan veya içeriden **saldırılara karşı korumayı amaçlar.**
- Bu saldırılar, **virüsler, worm'lar, DoS, ...** gibi çok farklı şekillerde olabilir.
- **Bu saldırılardan korunmayı bazı işletim sistemleri görev** olarak düşünürken, **bazı işletim sistemleri** bu işleri **diğer yazılımlara bırakır.**
- Protection ve security, sistemdeki **tüm kullanıcıların birbirinden ayırt edilebilmesini gerektirir.**
- Çoğu işletim sistemi bunu **kullanıcı kimlikleri (user ID)** ile yapar.
- Grup ID veya sistem yöneticisi **(admin)** tanımlamaları da yapılabilir.

Kernel veri yapıları

Bağlı listeler

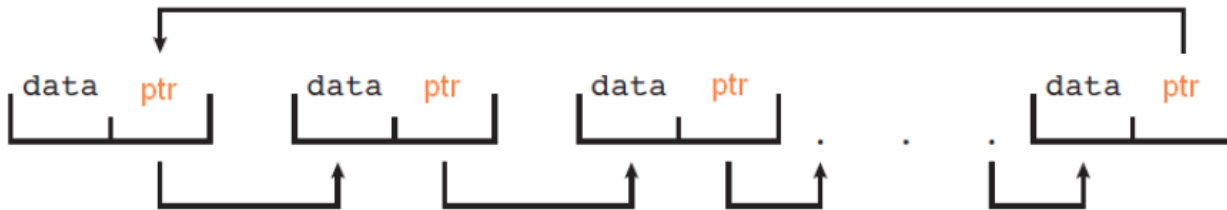
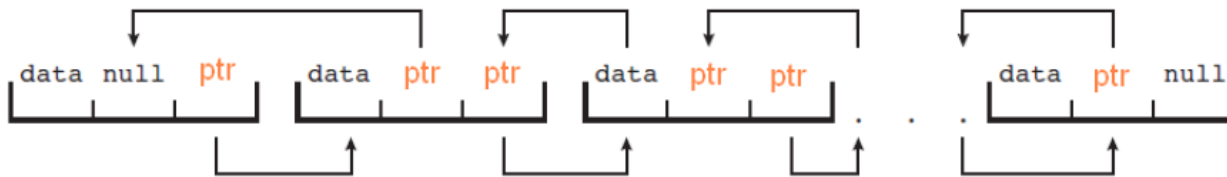
- Bir **dizi (array)** basit bir veri yapısıdır ve **elemanlara doğrudan erişim sağlar.**
- Dizilerde **elemanların boyutları sabittir** ve bir elemana doğrudan erişmek için önündeki eleman sayısı ile bir elemanın boyutu çarpılır.
- Ancak çoğu uygulamalarda **elemanlar farklı boyutlarda olabilir.**
- **Bu durumda, bağlı listeler (linked lists) kullanılır.**
- Bağlı listeler, bir grup veriyi art arda sıralı halde tutar ve doğrudan erişime olanak sağlar.



Kernel veri yapıları

Bağlı listeler

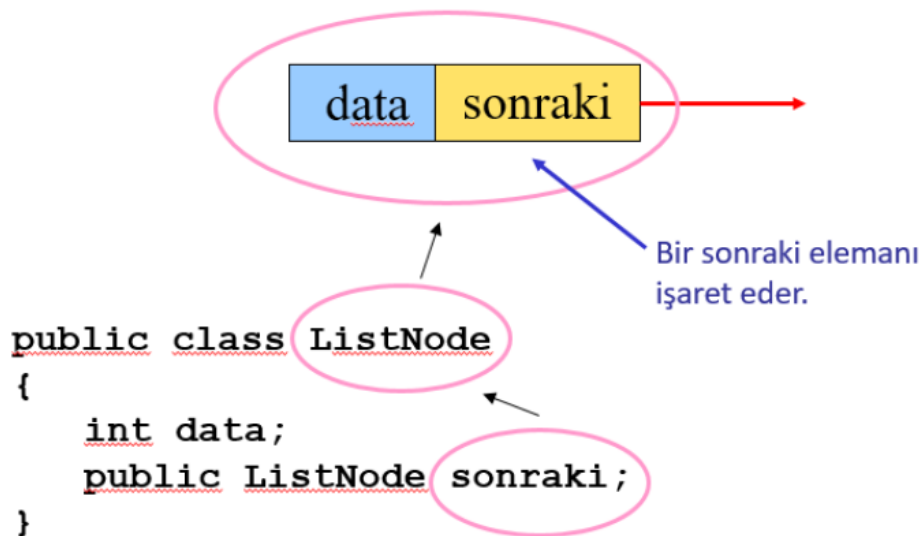
- Bağlı listeler bir bağlı (*singly linked list*), iki bağlı (*doubly linked list*) veya dairesel bağlı (*circularly linked list*) olabilir.



Kernel veri yapıları

Bağlı listeler

- Bağlı listelerde bir elemana erişim performansı $O(n)$ 'dir (worst case).
- Tek bağlı listenin tanımı aşağıdaki gibi yapılabilir.



Kernel veri yapıları

Yığın

- Yığınlar (**stack**) son gelen ilk çıkar (**last in first out - LIFO**) şeklinde çalışan veri yapılarıdır.
- Bir stack üzerine yeni eleman eklemek için **push**, bir stack üzerinden son eklenen elemanı almak için **pop** işlevleri kullanılır.
- İşletim sistemleri, iç içe fonksiyon çağırımlarında stack yapısını kullanır.
- Yeni fonksiyona geçiş yaparken ve geri dönerken, **interrupt** altıyordamına geçiş yaparken ve geri dönerken, parametreler, lokal değişkenler ve dönüş adresi stack üzerine saklanır.

Kernel veri yapıları

Kuyruk

- Kuyruklar (**queue**) ilk gelen ilk çıkar (**first in first out - FIFO**) şeklinde çalışan veri yapılarıdır.
- Bir kuyruk üzerine yeni eleman eklendiğinde en sona kaydedilir, bir kuyruk üzerinden eleman alındığında en baştaki alınır.
- İşletim sistemleri, **yazıcıya iş gönderirken, işlemci tarafından çalıştırılmak için bekleyen görevlerin yönetiminde kuyruk yapısını kullanır.**

Kernel veri yapıları

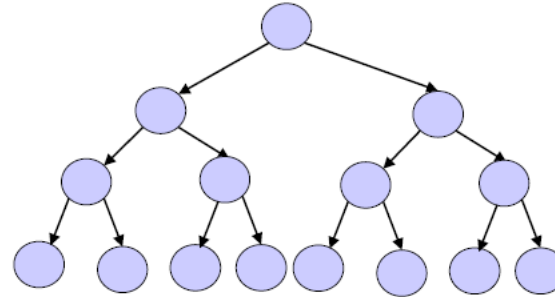
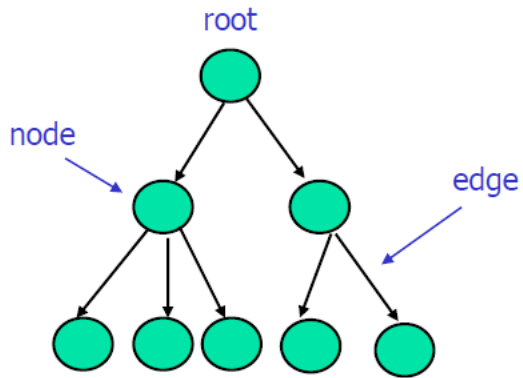
Ağaçlar

- Ağaçlar (**tree**) veriyi hiyerarşik şekilde göstermek için kullanılır.
- Ağaçlarda veriler **parent-child** ilişkisiyle tanımlanır.
- Genel olarak bir düğümde istenildiği kadar child olabilir.
- İkilik ağaçlarda (**binary tree**) ise bir düğüm iki child düğüme sahip olabilir.
- Binary arama ağaçlarda (**binary search tree**) bir elemana erişim performansı $O(n)$ 'dir (**worst case**).
- Dengelenmiş binary arama ağaçlarda (**balanced binary search tree**) bir elemana erişim performansı $O(\log n)$ 'dir (**worst case**).

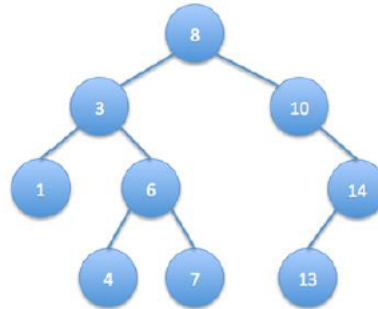
Kernel veri yapıları

Ağaçlar

- Bir ağaç (tree) veriyi hiyerarşik şekilde göstermek için kullanılır.



Dengelenmiş ikilik arama ağacı



Binary search tree

Kernel veri yapıları

Bitmap

- **Bitmap'ler n adet binary bit ile oluşturulan dizgidir (string).**
- Hash fonksiyonlarında veriye erişim performansı $O(1)$ 'dir.

001011101

- Çok sayıdaki kaynağın durumları ile ilgili bilgi (meşgul, kullanılabilir) bitlerle tutulabilir.
- Yukarıdaki bitler için 0, 1, 3 ve 7.kaynaklar kullanılabilir; 2, 4, 5, 6 ve 8.kaynaklar meşgul durumdadır.
- İşletim sistemi, disk bloklarının durumunu tutmak için bitmap kullanır.
- **İşletim sistemleri kernel algoritmalarında veri yapılarını sıklıkla kullanır.**



Ders bitti

Erciyes Üniversitesi
Selçuk Üniversitesi
Sakarya Üniversitesi
Kahramanmaraş Sütçü İmam Üniversitesi
ders notları kaynak ve içerik olarak kullanılmıştır.