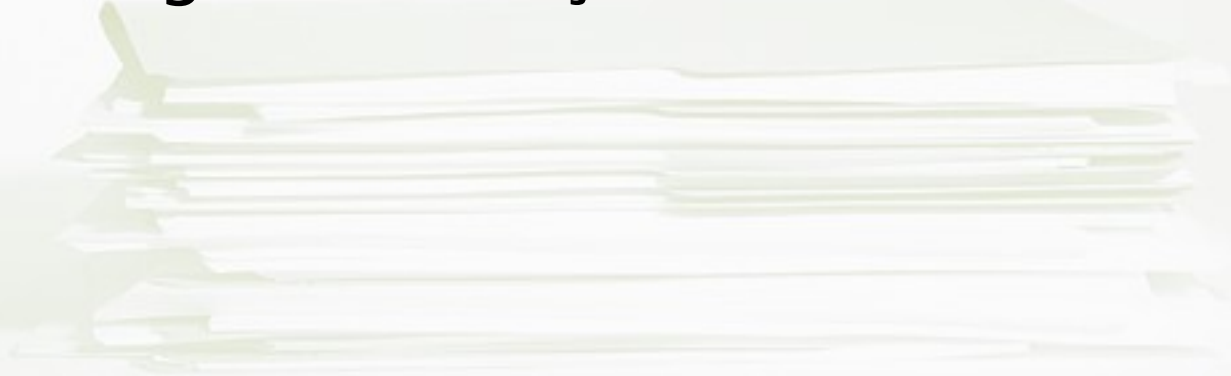


İşletim Sistemlerine Giriş - 2

Kaynakların Paylaşımı

Öğr.Gör. Dr. Şirin KARADENİZ



Kaynakların Paylaşımı

- Sistem, sistem kaynaklarını belli bir hiyerarşi içinde kullanıcının hizmetine sunar. Bir işletim sisteminde paylaşılan kaynaklar ise şunlardır:
 - Donanım: MİB, Bellek, Giriş/Çıkış birimleri vb...
 - Klasör ve Dosyalar
 - Yazılım: Programlar

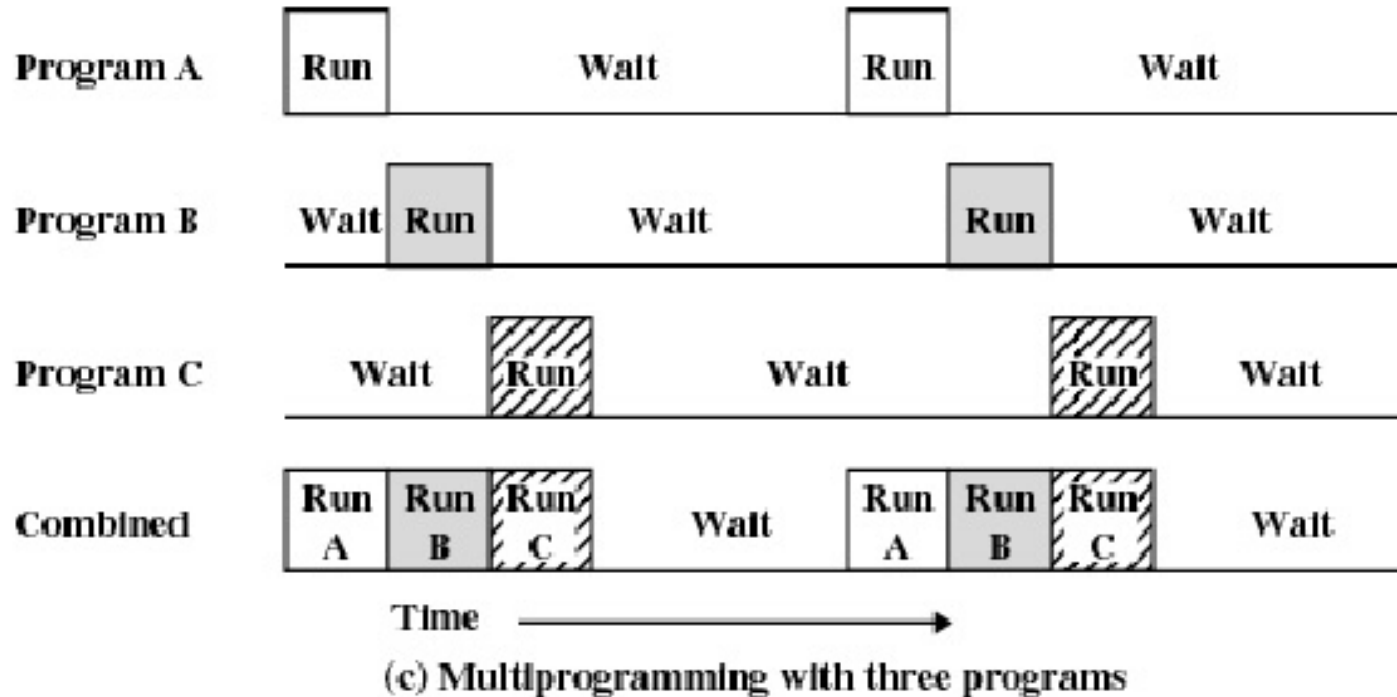
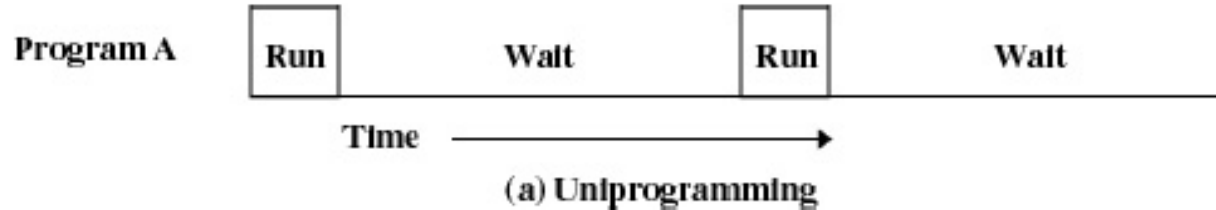
Kaynakların Paylaşımının Nedenleri

- Maliyet: Her kullanıcıya birbirinden bağımsız kaynak sağlamak zordur.
- Birinin geliştirdiği bir programı veya uygulamayı diğerleri de kullanabilir
- Aynı veri tabanı birden fazla kullanıcı tarafından kullanılabilir
- Bir programın birden fazla kullanıcı tarafından kullanılarak depolama birimlerinden tasarruf sağlanır ve geçersiz kaynak kullanımının önüne geçilebilir

Aynı anda çalışma (Concurrent execution)

- Bir bilgisayarın birden fazla programı aynı anda çalıştırmasıdır.
- Birden fazla program mantıksal olarak aynı anda çalışmakta iken fiziksel olarak sıralı olarak çalışmaktadır.
 - Örn: Çoklu programlamada MİB'ni programların paylaşması.

Çoklu programlama



Paralel çalışma (Parallel execution)

- Bir bilgisayarın birden fazla programı gerçekten aynı anda çalıştırmasıdır.
- Birden fazla program hem mantıksal hem de fiziksel olarak aynı anda çalışmaktadır.



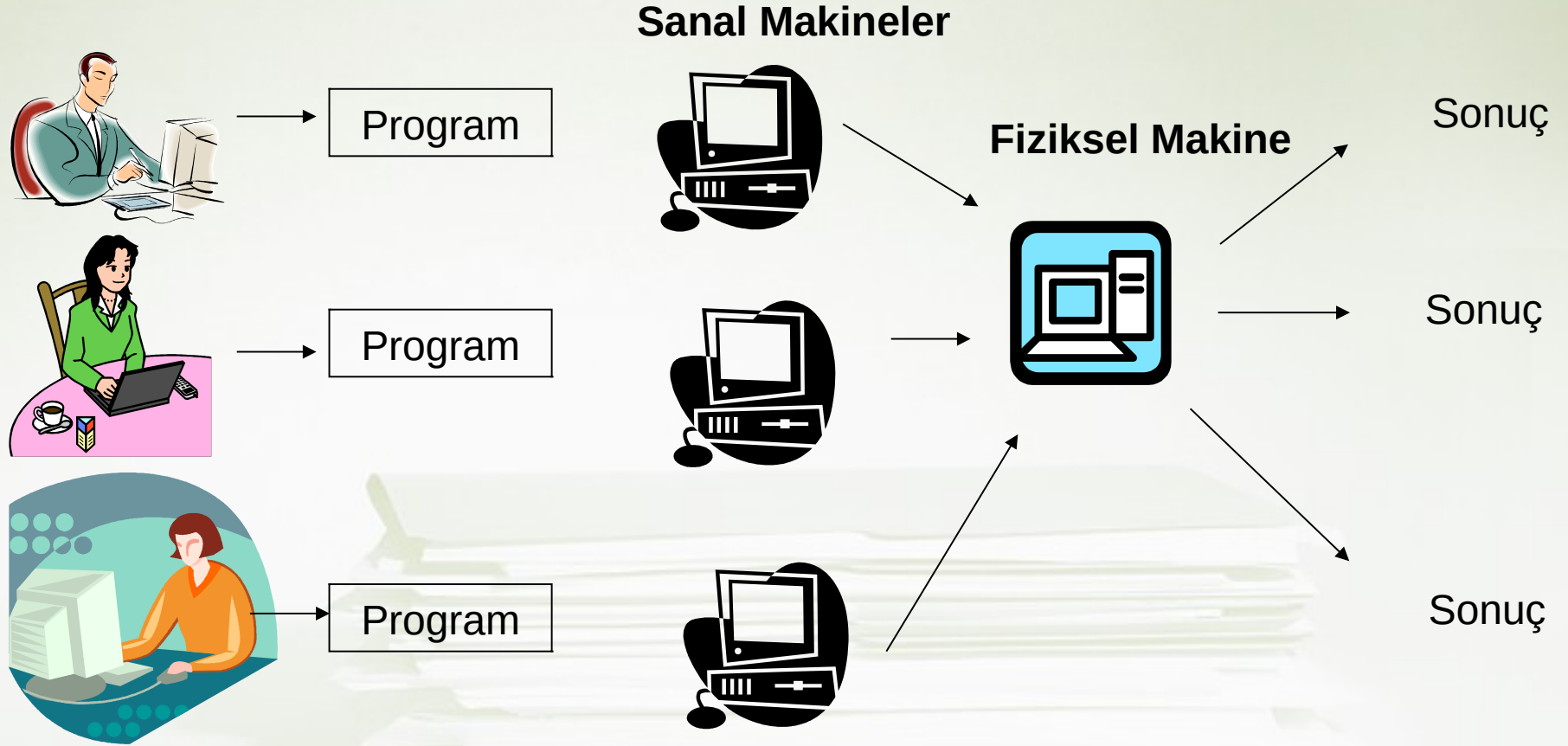
Paylaşım Türleri

- Saydam paylaşım (Transparently sharing):
 - Kaynakların paylaşımının işletim sistemi tarafından yürütmesidir. Burada kullanıcı kaynakların paylaştırıldığından habersizdir. Bu paylaşım türünün gerçekleştirilebilmesi için soyut makineler kullanılmaktadır.
- Açık paylaşım (Explicit sharing):
 - İşlemlerin genel kaynakları kendi politikalarına göre kullanmalarıdır. İşletim sisteminin kullanıcıya makinedeki kaynakları paylaşmasına izin vermesidir.

Sanal/Soyut Makine Sağlanması ve Saydam Kaynak Paylaşımı

- Kullanıcılara yalnızca kendine tahsis edilmiş bir makine varmış gibi çalışma ortamının sağlanmasıdır.
- Her bir sanal makine fiziksel makinenin bir simülasyonudur.
- Bu işlemleri yapabilmek için fiziksel makine donanımlarını saydam paylaşım tekniğini kullanarak sanal makineler arasında paylaşır.
- Sanal makine tarafından çalıştırılan programa genellikle 'process (işlem)' denilmektedir.

Sanal Makineler



Sanal makinelerin oluşturulmasında saydam paylaşım tekniğinin 2 tür paylaşımı kullanılır;

- Alan çoklama paylaşımı (Space-multiplexed sharing):
 - Bir kaynak bir veya daha fazla bölüme ayrılır, ardından her bir bölüm bir işlem'e atanır.
 - Örn: Bellek ve hard disk.
- Zaman çoklama paylaşımı (Time-multiplexed sharing):
 - Kaynak bir işlem tarafından belli bir süre kullanılır yani o işleme adanır ardından diğer işlem bu kaynağı belli bir süre kullanır.
 - Örn: MİB.

Çoklu programlama (Multiprogramming)

Sanal Makine 1 (S_1)



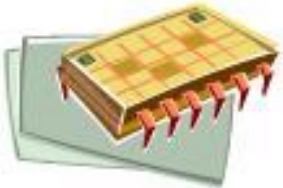
Sanal Makine 1 (S_2)



Sanal Makine 1 (S_3)



Zaman çoklama



Fiziksel MİB

Alan çoklama



Fiziksel Bellek

Açık paylaşım (Explicit sharing)

- İşlemlerin genel kaynakları kendi politikalarına göre kullanmalarındır. Dikkat edilecek iki önemli nokta ise;
 - Kaynak yalıtımı (Resource isolation) ve
 - İşbirliği yaparak paylaşım (Cooperatively sharing) dır.

Kaynak yalıtımı (Resource isolation)

- Sistem yerleşim politikasına göre kaynaklara ulaşımı ayırabilmelidir.
- İşletim sistemi, bir soyut makine tarafından kullanılan kaynağa diğer yetkisiz işlemlerin ulaşmasını engellemelidir.

Kaynak yalıtımı (Resource isolation)

- Örneğin bellek yalıtım mekanizmasında, bellek belli bölümlere ayrılarak her bir bölüm ayrı bir soyut makine tarafından kullanılmaktadır.
- Bu durumda sistem, bir soyut makineye ayrılmış bir bellek bölümüne başka bir soyut makinenin müdahale etmesini engellemelidir.
- Aynı şekilde MİB'nin işlemler tarafından sıralı olarak paylaşılmasında da, bir işlem MİB'ni kullanırken diğer işlemlerin bellek alanlarına MİB yoluyla müdahale etmemelidir.

İşbirliđi yaparak paylaşım (Cooperatively sharing)

- Sistem işlemlerin kaynakları istenildiđi takdirde işbirliđi yaparak paylaşmalarına izin verebilmelidir.
 - Örneđin bir işlem diđer bir işlemin sonucunu kullanabilmeli veya belirli bir bellek bölgesindeki bilgiler iki işlem tarafından kullanılması gerektiđinde buna izin verebilmelidir.

Sistem Çađrı Arayüzü

- Güvenilir bir işletim sisteminde donanım kaynaklarının doğru bir biçimde paylaşılması ve birbirinden yalıtılması gerekmektedir.
- Bunları yapabilmek için bir işletim sisteminin soyutlamaları, sistem çağrı arayüzü (system call interface) olarak da bilinen işletim sisteminin arayüzü kullanır.

API (Application Programming Interface)

- Tüm sistem yazılımları bir uygulama programlama arayüzü - API (Application Programming Interface) yoluyla ulaşılabilir.
- API; bir yazılım veya sistem yazılım parçasının programlama arayüzüdür. Veritabanları, VBasic editörü gibi.
- Programcılar uygulama programlama arayüzlerini kullanırlarken, işletim sistemi, sistem çağrı arayüzünü kullanır.

API (Application Programming Interface)

- Uygulama programlama arayüzleri sistem yazılım arayüzlerine ulaşarak bunların kullanıcı tarafından kullanılabilmesini sağlar.
- Microsoft Windows sistem çağrı arayüzünü “Win32 API” olarak adlandırmıştır.



Uygulama yazılımı, Sistem yazılımı ve İşletim Sistemi

İnsan-Bilgisayar Arayüzü

(Human-Computer Interface)



API

Uygulama Yazılımı

İşletim Sistemi Arayüzü

Sistem Yazılımı
(Soyut Kaynaklar)

Yazılım-Donanım Arayüzü

Güvenilir İşletim
Sistemi (Soyut
Kaynaklar)

Donanım Kaynakları

