**1.HAFTA:**

İşletim sistemi donanımı yönteten bir yazılımdır.

Bilgisayar sistemi 4 bileşen; donanım, işletim sistemi, uygulama programalar, kullanıcılar

**2.HAFTA:**

**3.HAFTA:**

**4.HAFTA:**

Gelmedim

**5.HAFTA:**

**Genel Notlar:**

Multitasking destekleyen sistemler: Windows,Linux.unix

DOS desteklemez

32 bit sistemi çalıştırmak için 64 mb ve 4gb ram olmalı

Önceliği düşük olan işlemlerin diske taşınmasına SWAPING (TAKAS) denir.

Sanal Bellek: RAM-HDD

Adres defteri sistem araçlarından biri değildir.

Sabit Diskin ilk sektörü: MBR’ idir

Kaynak kod uzantısı: .asm

Gömülü İşletim Sistemi PalmOS

Alttan yukarıya : Donanım- Sistem Yazılımları-Uygulama Yazılımları

Process : Çalıştırabilir bir programın, program verileri, program sayacı ve diğer bileşenlerden oluşan veri çatısı.

Dos html uzantılarını çalıştırmaz.

Dos donanım özellikleri değiştirme config.sys

7.HAFTA:

DEADLOCK

Köprüden iki keçi geçmeye çalışıyor. Biri hep diğerini bekliyor.

Multiprogramming ve multitasking ortamlarda ortaya çıkar.

MSDOS’ta ölümcül kilitlenme olmaz. Sınırlı kaynak =RAM

Request ile istek yapılır kaynak için

Kaynağı bekleyecek kaynak kullanılabilir değilse

Release ile kaynağı serbest bırakır

Bir process bir kaynağı kullanmadan önce istek yapar kullandıktan sonra serbest bırakacak.

Request

Use

Release

Mutex = Karşılıklı dışlama anlamına gelir

Kaynaklar fiziksel (yazıcı, ram) veya mantıksal (mutex, semafor) olabilir

DEADLOCK TANIMI VE ÖZELLİKLERİ

Deadlock durumunda process sonlanamaz. Sistem kaynağı bırakamıyor.

Bir process bir kaynağı tutarsa başka kaynak onu kullanamaz. Mutual Exclusion.

Hold and wait. Kaynağı tuttu, başka bir kayanağı bekliyor ihtiyacı var oğlum.

Circular wait: Processler birbirini beklemektedir. Her process birbirini bekliyor. P0, P1

Edge (kenar) processlerdir

Düğümler (verticles) ise kaynaklardır.

Döngü varsa deadlock olabilir. Döngü olduğu ama deadlock olmadığı şeyler olabilir. Herkes bir ley bekliyor.

Kaynağın hepsini kullanan var başka bir process o kaynağa erişmek ister ise nannayı yedik.

Bir kaynak kullanıyor ama başkası da onu kullanmak istiyor.

Deadlokc yönetimi için deadlock prevention ve deadlock-avoidance teknikleri kullanılır.

Deadlock prevention kaynak isteklerini sınırlandırarak deadlock engeller

Deadlock avoidance ise kaynağın ne zaman kullanacğaını bilmek istiyor

DEADLOCK ÖNLEME:

Mutual exclusion

4 durumda gerçekleşecek (mutual,circualar,hol and wait,nopreemtption)

Kesikli çizgi işi bitmeş process’leri betimler

BANKER ALGORİTMASI:

Aynı kanakta birden fazla olan sistemlerde uygulanamaz.

Aynı kaynakta birden fazla olan sistemlerde banker kullanılablilir.

Banker algoritması genel çıkabilir algoritma mantığını bilmeye gerek yoktur.