



## BM 307 Dosya Organizasyonu (File Organization)

---

Hazırlayan: M.Ali Akcayol  
Gazi Üniversitesi  
Bilgisayar Mühendisliği Bölümü



## Konular

---

- Indexed Sequential File Organization
  - Background
  - Basic Structure
  - Data Access
  - Değerlendirme
  - Overflow Area



## Indexed Sequential File Organization

### Background

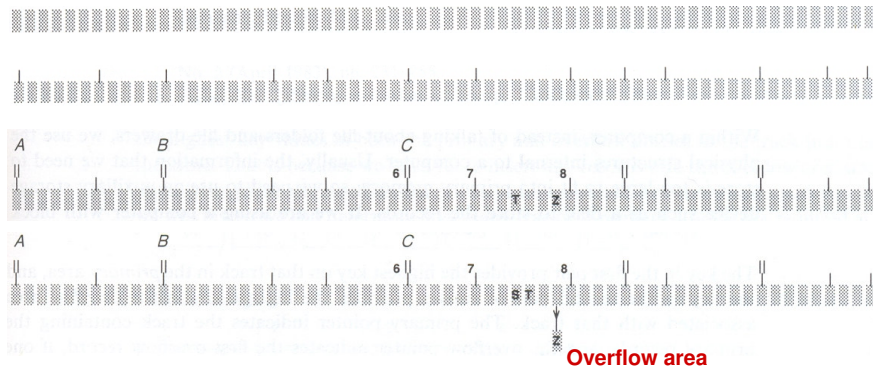
- Sequential file organization sıralı erişimde yüksek performansa sahiptir
- Direct file organization doğrudan erişimde yüksek performansa sahiptir
- Indexed sequential file organization hem sıralı erişimde hemde doğrudan erişimde kullanılmak amacıyla oluşturulur
- Sequential file organization sıralı erişimde ortalama kayıtların yarısına bakılmasını gerektirmektedir
- Indexed Sequential Structure, ISAM (Indexed Sequential Access Method) olarak adlandırılır



## Indexed Sequential File Organization

### Background

- Sıralı kayıtlar arasına tab (index) noktaları konulur
- Aranılan kaydın ilgili olduğu grup belirlendikten sonra sadece o gruptaki kayıtlar detaylı aranır





## Indexed Sequential File Organization

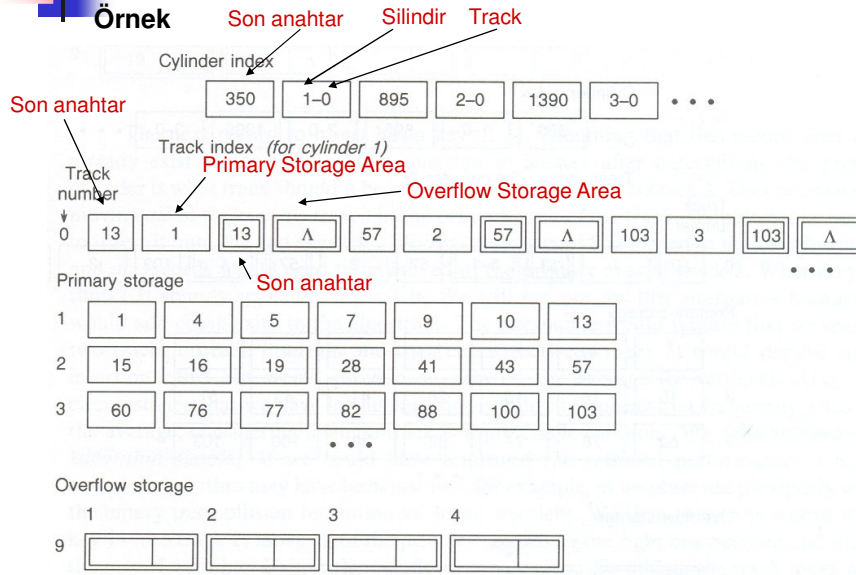
### Basic Structure

- Blok adreslenebilir disklerde birinci seviye indeks track'ları ve ikinci seviye indeks ise silindirleri gösterebilir
- Silindir indeks ilgili silindirdeki en yüksek anahtarı ve o silindir içindeki track indeksini tutar
- Bir silindirdeki her bir track iki çift bilgi tutar. Birisi primary depolama alanıyla ilgili diğeri ise overflow alanla ilgilidir
- İlk çift primary alandaki en yüksek anahtar bilgisini ve ikinci ise overflow alandaki en yüksek anahtar bilgisini tutar
- Primary pointer primary kayıtların olduğu track ve overflow pointer ise overflow'daki ilk kaydı gösterir



## Indexed Sequential File Organization

### Data Access

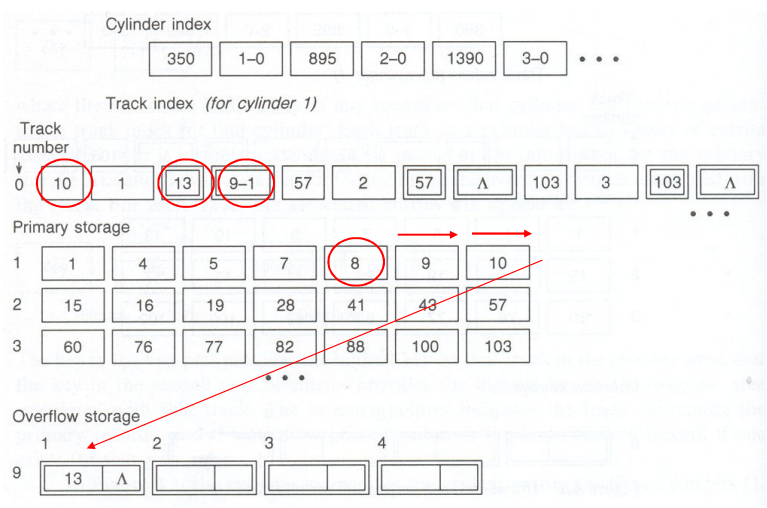


# Indexed Sequential File Organization

## Data Access

Örnek

8 anahtarı eklendi

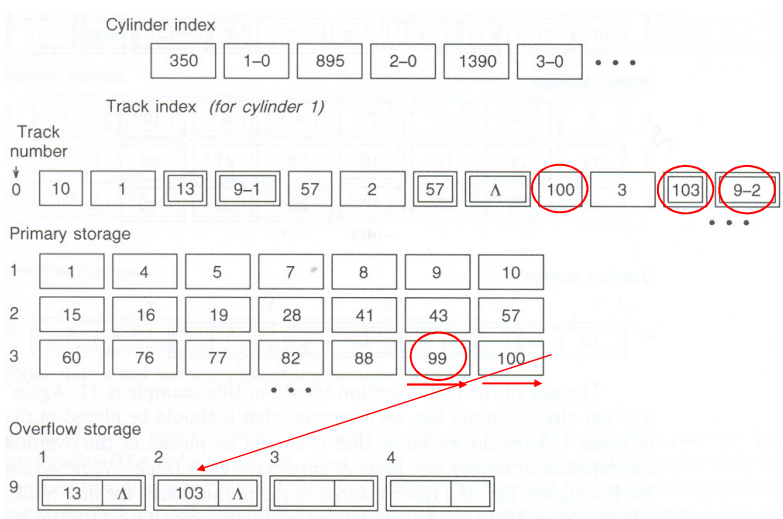


# Indexed Sequential File Organization

## Data Access

Örnek

99 anahtarı eklendi

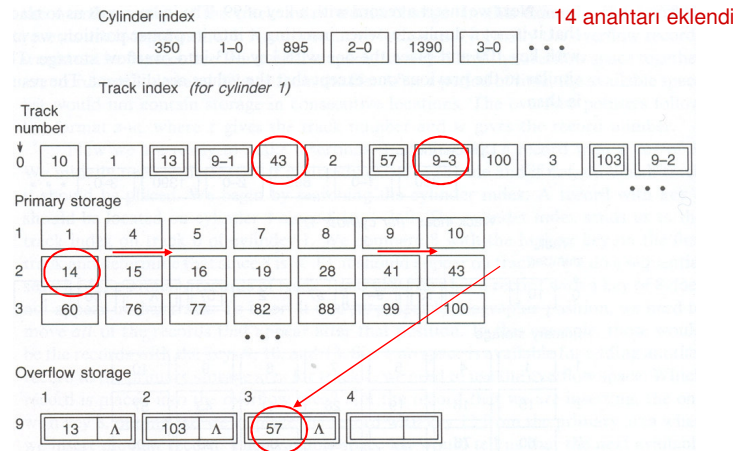




# Indexed Sequential File Organization

## Data Access

Örnek



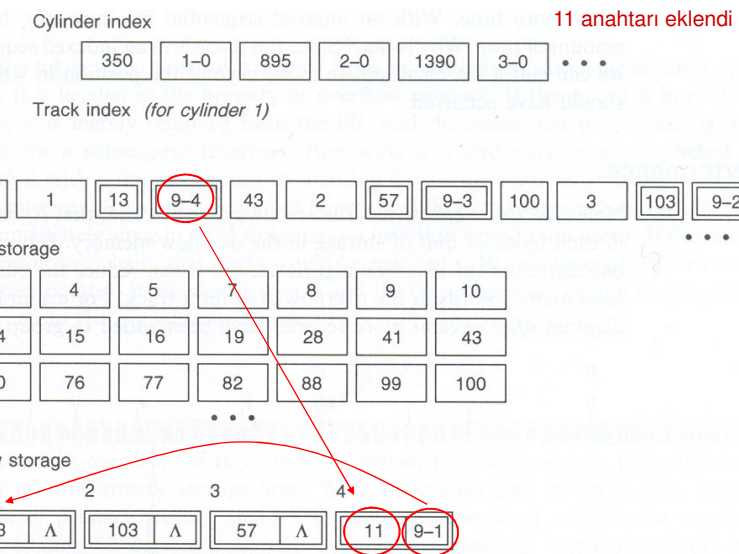
- 14 anahtarı 1.track sonuna eklenirse insertion ve retrieval performansı artar
- Algoritma daha kompleks hale gelir



# Indexed Sequential File Organization

## Data Access

Örnek





## Indexed Sequential File Organization

### Değerlendirme

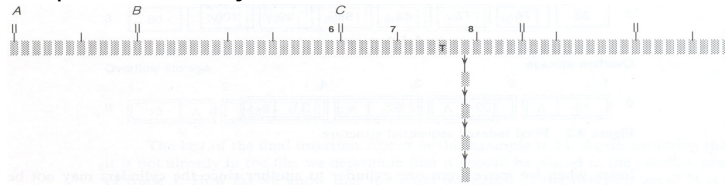
- Dosya üzerinde baştan başlayarak doğrudan okuma yapılamaz.
- Kayıtlar arasında sıralı erişim için silindir ve track bilgilerine ulaşılması gerekir
- Indexed Sequential File yapısında sıralı erişim Direct Organized File yapısına göre daha hızlıdır ancak Sequential Organized File yapısına göre daha yavaştır
- Bir kayda erişim için insert sırasındaki işlemler yapılır
- Bir kayıt için index bilgilerine bakılarak erişim yapıldığı için Direct Organized File yapısına göre retrieval süresi büyüktür
- Sequential Organized File yapılarında başarısız arama daha çok süre alır. Indexed Sequential File yapısında ortalama aynı süreyi alır. Kaydın olması gereken yere bakıldıktan sonra olup olmadığına karar verilir.



## Indexed Sequential File Organization

### Değerlendirme

- Bir track üzerindeki bir overflow area içerisindeki kayıt sayısı arttıkça arama performansı düşer



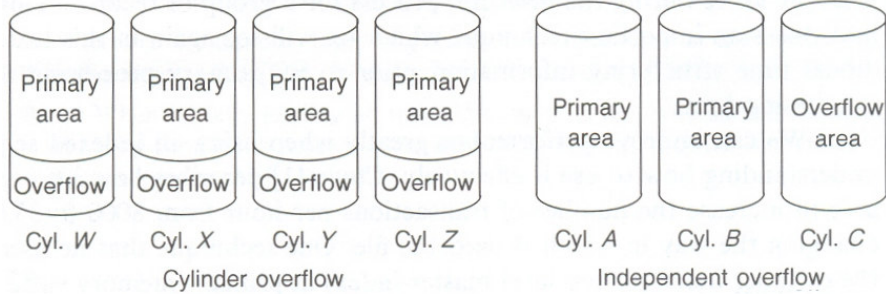
- Tüm yapı bir ağaç yapısı olarak düşünülebilir. Overflow area içerisine yapılan her kayıt alt ağacın yüksekliğini artırır
- Bu büyümeyi engellemek için periyodik olarak dosyanın yeniden organize edilmesi gerekir. Tüm bilgiler yeniden yerleştirilir
- Yeniden yapılandırma bilgisayarın kullanılmadığı zamanlarda yapılır
- Silme işlemi silinen elemanın bulunduğu yere göre yapılır
- Eğer silinen eleman primary hafızada ise silinen yere tombstone konulur veya boş kayıt yapılır
- Silinen eleman overflow area içerisindeyse bağlı liste yeniden düzenlenir



## Indexed Sequential File Organization

### Overflow Area

- Overflow area iki farklı şekilde oluşturulabilir. Cylinder ve Independent overflow
- Cylinder overflow'da overflow area primary hafızayla aynı cylinder üzerindedir
- Independent overflow'da birden fazla sayıdaki primary hafıza alanı aynı cylinder'ı overflow area olarak kullanır



## Indexed Sequential File Organization

### Overflow Area

- Cylinder overflow'da okuma yazma kafası yeniden konumlanmaya gerek kalmadan overflow area'daki bilgiyi okuyabilir
- Time-sharing sistemlerde bu avantaj kaybolabilir. Bir kullanıcı primary hafızayla overflow hafıza arasındaki geçişi engelleyebilir
- Independent overflow area kullanımında toplam overflow area için gereken alan daha azdır

#### Using Indexed Sequential

- Primary hafıza alanı doluyken ekleme işleminde elemanları önce sıralamak performansı artırır

A	B
39	72
47	68
72	47
17	39
22	36
68	22
36	17