



SAKARYA
ÜNİVERSİTESİ

Sakarya Üniversitesi

Bilgisayar ve Bilişim Bilimleri
Bilgisayar Mühendisliği Bölümü
Sistem Programlama Dersi

Yapanlar

Ali Kerem Kol-B221210042

Ömer Elmas- B221210582

Şube: 1-B

Öğretmen: Abdullah Sevin

1. Proje Tanımı

Bu projede, gerçek bir disk yerine tek bir dosya (disk.sim) üzerinde çalışan **Basit Dosya Sistemi Simülatörü** (“SimpleFS”) geliştirilmiştir. Amaç, işletim sistemi seviyesindeki temel dosya işlemlerinin (oluşturma, silme, okuma, yazma, yeniden adlandırma, kopyalama, taşınma, defragmentasyon, bütünlük kontrolü gibi) **yalnızca sistem çağrılarını** kullanarak nasıl gerçekleştirildiğini pratiğe dökmektir.

- **Dil:** C
- **Sistem çağrıları:** open, read, write, lseek, close, ftruncate, unlink vs.
- **Disk yapısı:**
 - disk.sim (1 MB)
 - İlk 4 KB: metadata bölümü
 - Geriye kalanı: veri bölümü

2. Proje Bileşenleri

1.fs.h

- Metadata ve FileEntry yapılarını tanımlar.
- Sabitler: blok boyutu, maksimum dosya sayısı, DISK_NAME, DISK_SIZE, METADATA_SIZE vb.

2.fs.c

- Tüm fs_* fonksiyonlarının tanımları burada yer alır.
- load_metadata(), save_metadata() ile metadata yönetimi.
- Her işlem için fs_log() çağrısıyla fs_log.txt'e kayıt tutulur.

3.main.c

- Konsol tabanlı menü arayüzü.
- Kullanıcının 1–19 arası seçimlerine göre ilgili fs_* fonksiyonu çağrılır.

4.Makefile

- gcc -Wall bayraklarıyla derleme.
- all ve clean hedefleri.

5.disk.sim (çalışma zamanında oluşturulur)

- Simüle edilmiş sanal disk dosyası.

6.fs_log.txt

- Proje çalıştıkça oluşan işlem loglarını içerir.

3. Fonksiyon Özeti

Her `fs_*` fonksiyonu, ödevde istenen komut setini tam olarak uygular. Aşağıda önemli başlıklarıyla kısa açıklamaları bulunuyor:

Fonksiyon	Açıklama
<code>fs_format()</code>	Disk içeriğini ve metadata'yı sıfırlar.
<code>fs_create(filename)</code>	Yeni dosya oluşturur. Aynı isim kontrolü ve maksimum dosya limiti.
<code>fs_delete(filename)</code>	Dosyayı “silinmiş” olarak işaretler, <code>total_files--</code> .
<code>fs_write(filename, data, size)</code>	Belirtilen veriyi dosyaya yazar. (Disk dolu kontrolleri eklenecek.)
<code>fs_read(filename, offset, size, buffer)</code>	Dosyadan doğru offset'ten veri okur. Sınır kontrolü mevcut.

Fonksiyon

Açıklama

`fs_ls()`

Aktif tüm dosyaların isim ve boyutlarını listeler.

`fs_rename(old, new)`

Dosya adını değiştirir, çakışma engellenir.

`fs_exists(filename)`

Dosyanın varlığını bool/0–1 döndürür.

`fs_size(filename)`

Dosyanın byte cinsinden boyutunu metadata'dan okur.

`fs_append(filename, data, size)`

Dosyanın sonuna veri ekler (append).

`fs_truncate(filename, new_size)`

Dosyayı belirtilen boyuta kısaltır.

`fs_copy(src, dest)`

Bir dosyanın tüm içeriğini diğerine kopyalar.

`fs_mv(old, new)`

`fs_copy` + `fs_delete` ile move işlemi.

`fs_defragment()`

Boş blokları birleştirir, parçalanmayı giderir.

Fonksiyon

Açıklama

`fs_check_integrity()`

Metadata ile veri alanı tutarlılığını kontrol eder.

`fs_backup(backup_name)`

disk.sim'i yedek dosyaya kopyalar.

`fs_restore(backup_name)`

Yedekten geri yükleme yapar.

`fs_diff(file1, file2)`

İki dosyanın byte byte farklarını karşılaştırır.

`fs_cat(filename)`

(!) Proje kapsamında doğrudan cat benzeri çıktı veren fonksiyon.

`fs_log(message)`

Her işlem `fs_log.txt`'e tarih etiketli olarak yazılır.

4. Kullanım Senaryosu

Program çalıştırıldığında:

\$./SimpleFS

```
SimpleFS - Dosya Sistemi Simülatörü
1. Dosya Oluştur
2. Dosya Sil
3. Dosyaya Yaz
4. Dosyadan Oku
5. Dosyaları Listele
6. Diski Biçimlendir
7. Dosya Adını Değiştir
8. Dosya Var mı Kontrol Et
9. Dosya Boyutunu Öğren
10. Dosyaya Ekle
11. Dosyayı Kısalt (Truncate)
12. Dosya Kopyala
13. Dosya Taşı
14. Diski Birleştir (Defragment)
15. Bütünlüğü Kontrol Et
16. Disk Yedeği Al
17. Disk Yedeğini Geri Yükle
18. Dosya İçeriğini Göster (cat)
19. İki Dosyayı Karşılaştır (diff)
20. Çıkış
Seçiminiz: █
```

Seçiminiz:

- Seçime göre gerekli parametreler kullanıcıdan scanf ile alınıp ilgili fs_* fonksiyonu çağrılır.
- Her işlem sonunda ekrana durum mesajı ve fs_log.txt'e kayıt.

5. Örnek Ekran Çıktıları

1. Dosya Oluşturma ve Listeleme

```
17. Disk Yedeğini Geri Yükle
18. Dosya İçeriğini Göster (cat)
19. İki Dosyayı Karşılaştır (diff)
20. Çıkış
Seçiminiz: 1
Dosya adını girin: deneme.txt

SimpleFS - Dosya Sistemi Simülatörü
1. Dosya Oluştur
2. Dosya Sil
3. Dosyaya Yaz
4. Dosyadan Oku
5. Dosyaları Listele
6. Diski Biçimlendir
7. Dosya Adını Değiştir
8. Dosya Var mı Kontrol Et
9. Dosya Boyutunu Öğren
10. Dosyaya Ekle
11. Dosyayı Kısalt (Truncate)
12. Dosya Kopyala
13. Dosya Taşı
14. Diski Birleştir (Defragment)
15. Bütünlüğü Kontrol Et
16. Disk Yedeği Al
17. Disk Yedeğini Geri Yükle
18. Dosya İçeriğini Göster (cat)
19. İki Dosyayı Karşılaştır (diff)
20. Çıkış
Seçiminiz: 5
Filename          Size (bytes)
-----
deneme.txt        0
```

2. Veri Yazma ve Okuma

SimpleFS - Dosya Sistemi Simülatörü

1. Dosya Oluştur
2. Dosya Sil
3. Dosyaya Yaz
4. Dosyadan Oku
5. Dosyaları Listele
6. Diski Biçimlendir
7. Dosya Adını Değiştir
8. Dosya Var mı Kontrol Et
9. Dosya Boyutunu Öğren
10. Dosyaya Ekle
11. Dosyayı Kısalt (Truncate)
12. Dosya Kopyala
13. Dosya Taşı
14. Diski Birleştir (Defragment)
15. Bütünlüğü Kontrol Et
16. Disk Yedeği Al
17. Disk Yedeğini Geri Yükle
18. Dosya İçeriğini Göster (cat)
19. İki Dosyayı Karşılaştır (diff)
20. Çıkış

Seçiminiz: 3

Dosya adını girin: deneme.txt

Veriyi girin: merhaba

SimpleFS - Dosya Sistemi Simülatörü

1. Dosya Oluştur
2. Dosya Sil
3. Dosyaya Yaz
4. Dosyadan Oku
5. Dosyaları Listele
6. Diski Biçimlendir
7. Dosya Adını Değiştir
8. Dosya Var mı Kontrol Et
9. Dosya Boyutunu Öğren
10. Dosyaya Ekle
11. Dosyayı Kısalt (Truncate)
12. Dosya Kopyala
13. Dosya Taşı
14. Diski Birleştir (Defragment)
15. Bütünlüğü Kontrol Et
16. Disk Yedeği Al
17. Disk Yedeğini Geri Yükle
18. Dosya İçeriğini Göster (cat)
19. İki Dosyayı Karşılaştır (diff)
20. Çıkış

Seçiminiz: 4

Dosya adını girin: deneme.txt

Offset ve boyut girin: 0 7

Okunan Veri: merhaba

3. Format Sonrası Listeleme

```
17. Disk Yedeğini Geri Yükle
18. Dosya İçeriğini Göster (cat)
19. İki Dosyayı Karşılaştır (diff)
20. Çıkış
Seçiminiz: 6

SimpleFS - Dosya Sistemi Simülatörü
1. Dosya Oluştur
2. Dosya Sil
3. Dosyaya Yaz
4. Dosyadan Oku
5. Dosyaları Listele
6. Diski Biçimlendir
7. Dosya Adını Değiştir
8. Dosya Var mı Kontrol Et
9. Dosya Boyutunu Öğren
10. Dosyaya Ekle
11. Dosyayı Kısalt (Truncate)
12. Dosya Kopyala
13. Dosya Taşı
14. Diski Birleştir (Defragment)
15. Bütünlüğü Kontrol Et
16. Disk Yedeği Al
17. Disk Yedeğini Geri Yükle
18. Dosya İçeriğini Göster (cat)
19. İki Dosyayı Karşılaştır (diff)
20. Çıkış
Seçiminiz: 5
Filename      Size (bytes)
-----
```

6. Değerlendirme

. Öğrenilenler:

- Sistem çağrılarının gerçek dünya senaryolarında nasıl kullanıldığı.
- Basit bir dosya sisteminin metadata yönetimi ve blok bazlı veri yerleşimi.
- Dosya tabanlı simülasyonlar ve hata durumlarının ele alınması.

. Karşılaşılan Zorluklar:

- Defragment algoritması tasarımı (boş blokları toplama)
- Bütünlük kontrolü: metadata ile veri bloklarının uyumunu sağlama

- Disk doluluk kontrolü

7. Kaynaklar

- Sistem Programlama dersi materyalleri
- Ders slaytları ve örnek kodları
- Unix/Linux man sayfaları (open, lseek, read, write, unlink)
- libfdt lecture notes