

Sakarya Üniversitesi  
Bilgisayar ve Bilişim Fakültesi  
Bilgisayar Mühendisliği  
İşletim Sistemleri  
**Proje Ödevi**

**Hazırlayanlar:**

Zehranur Sarı B221210102

Büşra Sevinç B221210084

Ayşe Aktaş B221210036

Elif Özhan B221210001

Gökçe Çiçek Yağmur B221210019

GitHub Linki:

<https://github.com/gokcherry/OSGrup26.git>

- **Projenin Amacı**

Bu projede Linux ortamında bir *shell* uygulaması geliştirilir. *Shell* kullanıcıdan aldığı komutları yorumlayarak işletim sistemi seviyesinde süreç yönetimi ve giriş/çıkış yönlendirme işlemlerini gerçekleştiren bir araçtır.

- **Özellikler**

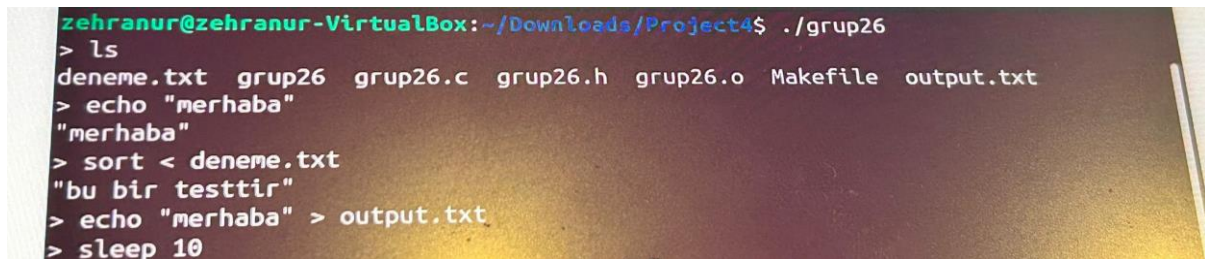
- **Prompt Gösterimi:** Kullanıcıya komut istemi **-prompt-** gösterilir ve yeni komutların girilmesi beklenir.
- **Built-in Komut:** **-quit-** komutuyla kabuk sonlandırılabilir.
- **Tekli Komutların İcrası:** Kullanıcıdan alınan komutlar bir alt süreçte çalıştırılır.
- **Giriş/Çıkış Yönlendirme:** Komutun standart girdisi bir dosyadan alınabilir veya standart çıktısı bir dosyaya yazılabilir.
- **Arka Plan İşlem Desteği:** Komutlar arka planda çalıştırılabilir.
- **Boru (Pipe):** Birden fazla komut borularla birbirine bağlanabilir.

- **Teknik Tasarım**

- **Genel Akış:**
  - Kullanıcıdan komut girişi alınır.
  - Girilen komut, parse edilerek ayrıştırılır.
  - Giriş/çıkış yönlendirme, arka plan veya boru durumları kontrol edilir.
  - Komut, bir alt süreçte çalıştırılır.
- **Alt Süreç Yönetimi:**
  - **Fork:** Ana süreç, komutu çalıştırmak için bir alt süreç oluşturur.
  - **Execvp:** Alt süreç, verilen komutu çalıştırmak için kendini yeni bir programa dönüştürür.

- Wait/Waitpid: Alt sürecin tamamlanması beklenir (Arka plan işlemleri hariç).
- **Giriş/Çıkış Yönlendirme**
  - Dup2: Standart girdiyi veya çıktıyı, dosya betiğine yeniden yönlendirir.
  - Eksik dosya durumunda, hata mesajı ekrana yazdırılır.
- **Arka Plan İşlemleri:**
  - Komut sonundaki & karakteri kontrol edilerek, komut arka planda çalıştırılır.
  - Arka plan işlemi tamamlandığında, proses kimliği (PID) ve geri dönüş değeri ekrana yazdırılır.
- **Boru (Pipe):**
  - Pipe Sistem Çağrılar: Borularda komutlar birbirine bağlanır. İlk komutun çıktısı ikinci komutun girdisi olarak kullanılır.

## • **Kullanım Örnekleri:**



```
zehranur@zehranur-VirtualBox:~/Downloads/Project4$ ./grup26
> ls
deneme.txt grup26 grup26.c grup26.h grup26.o Makefile output.txt
> echo "merhaba"
"merhaba"
> sort < deneme.txt
"bu bir testtir"
> echo "merhaba" > output.txt
> sleep 10
```

> cat < deneme.txt > output.txt

>cikis