



**Öğrencinin;**

**ADI: Selin**

**SOYADI: Daldaban**

**NO: 1421221014**

**BÖLÜM: Bilgisayar Mühendisliği**

**Projenin;**

**KONUSU: Linux Shell Yazımı**

**Dersin;**

**ADI: İşletim Sistemleri**

**EĞİTMEN: Prof. Dr. A. Yılmaz ÇAMURCU**

**Ar. Gör. Samet KAYA**

**Ar. Gör. Okan KARA**

## **İçindekiler**

<b>1. Özet</b>	<b>3</b>
<b>2. Proje Konusu</b>	<b>3</b>
<b>3. Proje Çıktıları ve Başarı Ölçütleri</b>	<b>3</b>
<b>4. Proje Süresince Yapılanlar</b>	<b>5</b>
<b>5. Kaynakça</b>	<b>5</b>

# 1. Özet

Shell nedir?

Linux'un türetildiği UNIX sistemlerinde komutları yorumlamak ve yönetmek için kullanılan programa kabuk denir. Bütün LINUX dağıtımlarında bir kabuk bulunur. Bir başka deyişle Kabuk bir LINUX sistemin olmazsa olmazıdır. [1]

Kullanıcıların program başlatmasına, dosyaları ve Linux sisteminde çalışan işlemleri yönetmesine olanak tanır. Kabuğun kaynağı **komut istemidir**. Komut istemi kabuğun etkileşimli kısmıdır. Metin olarak girilen komutları yorumlar ve çekirdekte (**kernel**) yürütür.[2] Yani **Linux kabuğu, kullanıcı ile çekirdek arasında yer alır**.

Neden kullanılır?

Programları, Dosya sistemlerini çalıştırabilir, Bilgisayar kodunu derleyebilir, Bir sistemi işletebiliriz. Kısacası bilgisayarımızı en etkili şekilde yönetebiliriz.

## 2. Proje Konusu

Proje konusu; adı Myshell olan kendi Linux Shell programını yazmamızdır. İçerisinde mul, div, ve rakam adlı alt programlar barındırmalıdır. Ana program olan Myshell'den, hangi komut çağırılıyorsa alt programa gidip işlemi yapmalıdır. Alt programlara veri taşıma işlemlerinin execv(), wait() ve fork() fonksiyonları ile yapılması beklenir. Bununla birlikte cat, exit, clear, yanlış işlem gibi kontrollerin ve komutların yapılması beklenir.

## 3. Proje Çıktıları ve Başarı Ölçütleri

Projede mul.c, div.c, rakam.c dosyaları myshell.c'den parametre olarak çalışmaktadır. Clear, exit, cat ve & operatörü ile olan işlemler ile yapılması istenilen sonuç alınmaktadır. Wait() fonksiyonu ile child processin işi bitene kadar ana process beklemesi sağlandı.

```
e used.
selin@selin-X555UB:~/Masaüstü/Selin_Daldaban_1421221014_opsis_proje1$ ./myshell

myshell >>div 3 2
3.00 / 2.00 = 1.50

myshell >>mul 4 5
4.00 * 5.00 = 20.00

myshell >>rakam 8
8 >> sekiz

myshell >>div 4 3 & div 3 1
4.00 / 3.00 = 1.33
3.00 / 1.00 = 3.00

myshell >>clear
```

```
selin@selin-X555UB: ~/Masaüstü/Selin_Daldaban_1421221014_opsis_proje1 - x
Dosya Düzenle Görünüm Ara Uçbirim Yardım

myshell >>cat selin
selin
myshell >>sel 5
yanlis komut girdiniz

myshell >>exit
selin@selin-X555UB: Masaüstü/Selin_Daldaban_1421221014_opsis_proje1$
```

```
seda@seda-LINUX ~/Desktop/Selin_Daldaban_1421221014_opsis_proje1 (copy) - +
File Edit View Search Terminal Help

be used.
gcc mul.c -o mul
gcc rakam.c -o rakam
seda@seda-LINUX ~/Desktop/Selin_Daldaban_1421221014_opsis_proje1 (copy) $ ./mysh
ell

myshell >>div 4 5 & mul 4 3
4.00 * 3.00 = 12.00
4.00 / 5.00 = 0.80

myshell >>div 5 5 & div 4 6
4.00 / 6.00 = 0.67
5.00 / 5.00 = 1.00

myshell >>cat selin
selin
myshell >>rakam 5
5 >> bes

myshell >>ssss
yanlis komut girdiniz

myshell >>exit
seda@seda-LINUX ~/Desktop/Selin_Daldaban_1421221014_opsis_proje1 (copy) $
```

```
myshell >>div 4 5 & rakam 4
4 >> dort
4.00 / 5.00 = 0.80
```

```
myshell >>rakam 5
5 >> bes

myshell >>div 4 5
4.00 / 5.00 = 0.80

myshell >>div 6 7 & div 67 6
67.00 / 6.00 = 11.17
6.00 / 7.00 = 0.86

myshell >>div 4 5 & rakam 4
4 >> dort
4.00 / 5.00 = 0.80

myshell >>
```

#### 4. Proje Süresince Yapılanlar

Proje süresince, `execv()` ve `fork()` komutlarını kullanarak diğer programlara geçişi yaptım. Program adları ile çalıştırmayı öğrendim. Makefile dosyası yapımını öğrendim. `Gets()` ile string aldım ve boşluk ve `'&'` işaretine göre böldüm. Çünkü `scanf()` kullanıldığında string sonunda boşluk karakteri olmadığı için düzgün çalışmıyordu. İlk işlem komutumu `newargv[0]` da tutarak, `&` ile olan işlemlerde ise `newargv[3]` de tutarak yaptım.

Zorlandığım kısım ise, `&` komutundan sonra olan işlemleri ayırmaktı. `Div.c`, `mul.c` ve `rakam.c` de `newargv[0]` e göre işlem yaptığı için komutu yani gideceği yerin adresini verdim ve ikinci işlem için (`newargv[3]`) `newargv+3` ile adresi arttırarak ikinci işlemin elemanlarını `mul`, `div` ve `rakam`ın almasını sağladım.

#### 5. Kaynakça

[1] <https://forum.ubuntu-tr.net/index.php?topic=2819.0>

[2] <https://teknopusula.com/linux-shell-nedir-linux-komut-satiri-nasil-kullanilir-6347/>

