

İşletim Sistemleri Proje Raporu

Hazırlayan: 1821221005 Furkan Gündoğan

1. **Projenin Adı:** Linux Shell Yazımı

2. **Proje Konusu:** Linux işletim sistemindeki 'bash' programına benzeyen, kendi programlarını çalıştırabilen ve adı 'myshell' olan bir Shell programı yazmak.

3. **Tamamlanan Talimatlar:**

3.1. Temel:

- 3.1.1. Toplama, çıkarma, kelime tekrarı için birer C programı yazmak.
- 3.1.2. İşlem ve myshell için birer C programı yazmak.
- 3.1.3. Tüm programlarımızı derleyebilecek bir Makefile oluşturmak.
- 3.1.4. myshell'in sürekli komut alacak şekilde çalışmasını sağlamak.
- 3.1.5. Myshell'in 'cat' ve 'clear' programlarını çağırmasını sağlamak.
- 3.1.6. Myshell'in '|' ile sıralı yazılmış komutları sırayla gerçekleştirmesini sağlamak.
- 3.1.7. Myshell çalışırken yapılacak hatalar sonucunda programın çökmesini engellemek ve kullanıcıya uyarılar vermek.

3.2. Shell İşleyişi

Programımızın temelini fork çağrısı ve exec program çağrısı oluşturuyor. Yeni bir process başlatmak için fork() fonksiyonunu çalıştırıyoruz. Fork kendisinin bir kopyasını ve yeni bir child process oluşturuyor. Dönüş değeri child process için '0' olmaktadır. Sıfır koşulu sağlandığı durumda da farklı bir programı çalıştırmak için exec() kullanıyoruz. Exec, child process'in artık exec() komutu ile çağrılacak programdan devam etmesini sağlıyor.

Exec'in farklı tipleri vardır. Sonuna gelen harfe göre:

- 'l' harfi varsa, programa gönderilecek olan parametreleri liste şeklinde alır.
- 'v' harfi varsa, programa gönderilecek olan parametreleri dizi şeklinde alır.
- 'e' harfi varsa, exec fonksiyonu ek olarak çevre değişkeni(envp) parametresi alır.
- 'p' harfi varsa, exec fonksiyonu çalıştırılabilir dosyanın yerinin belirlenmesinde 'PATH' çevre değişkenlerine bakar.

Ben programımda derste de yaptığımız gibi parametreleri dizi olarak gönderdim yani execve()'yi kullandım. İlk parametre olarak da çalıştıracağı dosya adını verdim. Aynı zamanda 'WEXITSTATUS(t)' kullanarak çalıştırdığımız programdan dönen değeri alabiliyoruz. Bu şekilde projede istendiği üzere önce işlem.c'yi oradan da topla.c/cikar.c'yi çağırıp işlem sonucunda dönen değerleri myshell'e dönüp orada sonucu yazdırabildim.

Not: Env'p değerini farklı yerlerde gönderebilmek için global bir değişken tanımlamak durumunda kaldım. Daha sonra programı incelediğimde Env'p'ye ihtiyacım olmadığını onu kaldırıp execv ile devam etsem işlemlerin daha tasarruflu olacağını farkettim.

4. Zorlandığım Kısımlar:

- Gelen komutu parse etmek için bir yöntem bulmak gerekiyordu. Bulduğum yöntemi bir metota çeviremediğim için kod tekrarına gittiğim yerler oldu bu da kodu uzattı.
- Kullanıcının yapabileceği parametre hatalarını kontrol edip uyarılar verdirdim ve bu da kodu biraz uzattı.

5. Gerçekleştirilemeyenler ve Sınırlandırmalar:

- Negatif sayılarla yapılacak işlemlerde doğru çözümü döndüremedim.
- Toplama ya da çıkarmada sonucun 0 olması durumunu kontrol edemedim.
- Yukarıdaki işlemleri myshell'den bir uyarı verdirerek çözüme gittim.
- Komut parse etme yöntemimden kaynaklı olarak çoklu komutları 10'da sınırladım.
- Komut satırından gets() ile bir char array'e okuma yapıyorum ve myshell'de hiçbir şey yazmadan enter'a basıp göndermek çekirdek hatasına sebep oluyor.

6. Yardım Aldığım Yerler

- Laboratuvar kodlarımız.
- Makefile uzantılı bir dosyayı internetten indirmek.
- String.h kütüphanesi.