

# SİSTEM PROGRAMLAMA FİNAL RAPORU

(29.05.2016)

Her iki dosyada(client-server) da öncelikle socket oluşturuyorum.Bu iki socket haberleşme için aynı isimde olmalı.Programın çalışması için önce server.c dosyası çalıştırılmalı.Client connect() fonksiyonu ile server a bağlanma isteği yolluyor.Bu sırada server ise listen() fonksiyonu ile client bağlantısını bekliyor.Client bağlantı isteği gönderince server while(1) içerisinde accept() fonksiyonu ile gelen client bağlantılarını kabul edip thread ile doChild fonksiyonuna gönderiyor.doChild fonksiyonu içerisinde de while(1) ile sonsuz döngü oluşturup burada client in socket e yazdığı mesajı okuyup listServer,listLocal,sendFile,listClients,help parametresi mi diye karşılaştırıp her biri için ayrı bir thread ile gerekli fonksiyonlarını çağırıyorum.Eğer bu parametrelerden biri girilmediyse server socket e boş mesaj yazıp gönderiyor.Bu arada thread ler i çağtırmadan önce de mutex le lock-unlock yapıyorum.Daha sonra bu fonksiyonların sonuçlarını socket'e yazıyorum.Client ise bu değerleri okuyup ekrana veriyor ve kullanıcının tekrar parametre girmesi için bilgilendirme mesajı verip yeni bir parametre alarak bu işlemleri tekrar gerçekleştiriyor.

**NOT→**Client ilk başta(while(1) den önce)server a bağlanınca ilk olarak kendi pid değerini socket e yazıyor ve server bu değerleri bir array de tutuyor.Server a ctrl-c sinyali gelince server bu array de tuttuğu tüm pid değerlerini kill ediyor.

while(1){

**Server**

**Client**

(2.)Okuyor

(1.)Yazıyor

(3.)Yazıyor

(4.)Okuyor

}

## Client.c

### Yardımcı fonksiyonlarım:

**void \*listLocal (void \*parg)→** client eğer mesaj olarak listLocal girerse server a bu mesajı gönderiyorum ancak client kendi içerisinde bu fonksiyon ile bulunduğu klasördeki dosyaları ekrana yazıyor.

**void \*help(void \*parg)→** client eğer mesaj olarak help girerse server a bu mesajı gönderiyorum ancak client kendi içerisinde bazı parametrelerin ne anlama geldiğini açıklama olarak ekrana veriyor.

**void \*send\_file(void \*parg)→** client eğer mesaj olarak sendFile girerse server a bu mesajı gönderiyorum ve global değişken olarak tanımladığım dosyanın kaç byte olduğunu okuyup servera gönderiyorum. Sonra dosya içeriğini servera gönderiyorum.

**void Ctrlc\_Sinyali(int signo)→** CTRL-C sinyalini yakalayan fonksiyonum.

### Parametrelerim:

*client <ip address>*

### Yapılamayan Kısımlar:

**\*sendFile,** gönderilecek dosyayı parametre olarak alması gerekirken dosya ismini global değişken olarak isimlendirip verip o şekilde server a gönderiyorum. Ve client-client transferi yok.

**\*client a ctrl-c sinyali gelince server da bitiyor.**

## Server.c

### Yardımcı fonksiyonlarım:

**void \*doChild(void \*parg)→** Bir client server a bağlanınca ilk olarak bu fonksiyon çağrılıyor ve içerisinde client'tan gelen mesaja göre aşağıdaki fonksiyonları çağırıyor.

**void \*listServer(void \*parg)→ )→** Client eğer listServer mesajı girdiyse server listServer parametresini okuyup socket'e server ın bulunduğu klasördeki dosyaları yazıp client'a gönderiyor.

**void \*IsClients(void \*parg)→** Client eğer IsClients mesajı girdiyse server IsClients parametresini okuyup kendisine bağlı olan clientların pid değerlerini sockete yazıp client a gönderiyor.

**void \*sendFile(void \*parg)→** Client eğer sendFile mesajı girdiyse server sendFile parametresini okuyup parametreyi aldım diyor, sonra clienttan dosyanın kaç byte olduğunu alıyor ve son olarak'ta dosyanın içeriğini alıp yeni bir dosya oluşturup içine bu bilgileri dolduruyor.

**void Ctrlc\_Sinyali(int signo)→** CTRL-C sinyali yakalayan fonksiyonum

### Parametrelerim:

Server<port number>

### Yapılamayan Kısımlar:

\*client server'a dosya gönderme yaparken, server client a dosya gönderimi yapmıyor.

**Muhammet Tayyip ÇANKAYA**

**131044054**