

# SİSTEM PROGRAMLAMA FİNAL PROJESİ RAPORU

## 1) Tek Client-Server Bağlantı

```
Uçbirim bayram@bayram-SATELLITE-P855: ~/Masaüstü/ServerDir
bayram@bayram-SATELLITE-P855:~$ cd Masaüstü/ServerDir
bayram@bayram-SATELLITE-P855:~/Masaüstü/ServerDir$ make
gcc -o BibakBOXServer BibakBOXServer.o -lpthread
bayram@bayram-SATELLITE-P855:~/Masaüstü/ServerDir$ make clean
rm *.o BibakBOXServer
bayram@bayram-SATELLITE-P855:~/Masaüstü/ServerDir$ make
gcc -c BibakBOXServer.c
bayram@bayram-SATELLITE-P855:~/Masaüstü/ServerDir$ ./BibakBOXServer A 10 18232
system path error:: No such file or directory
bayram@bayram-SATELLITE-P855:~/Masaüstü/ServerDir$ ./BibakBOXServer ../Server/A 10 18232
Multi-threading.
Multi-threading.
Multi-threading.
Multi-threading.
Multi-threading.
Multi-threading.
Multi-threading.
Multi-threading.
Multi-threading.
Multi-threading.
```

### Serverımız hakkında,

Bu program için, ../Server/A klasörü altında client directorylerini tutmalıyız ve 10 tane thread ile çalışacaktır. Program run edildikten sonra 10 tane “**Multi-threading**” çıktısı ile 10 threadimizin çalışması gösterilir. Client’in server’a bağlantısı gerçekleştiğinde uygun olan bir thread kurduğumuz semaphore özelliği ile seçilir. Serverımız bir client bekler.

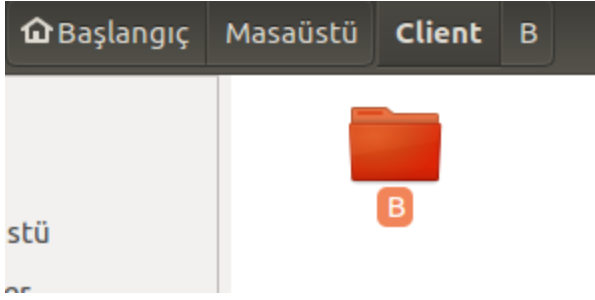


Server Directory Konumu

### Client Hakkında,

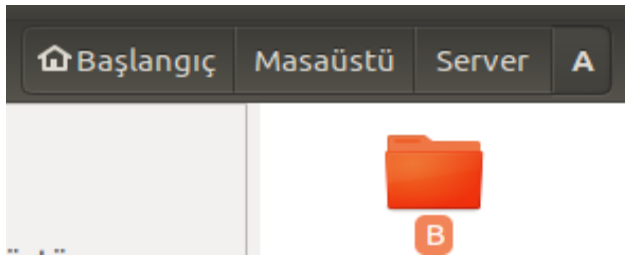
```
bayram@bayram-SATELLITE-P855: ~/Masaüstü/ClientDir
bayram@bayram-SATELLITE-P855:~$ cd Masaüstü/ClientDir
bayram@bayram-SATELLITE-P855:~/Masaüstü/ClientDir$ make
gcc -c BibakBOXClient.c
gcc -o BibakBOXClient BibakBOXClient.o -lpthread
bayram@bayram-SATELLITE-P855:~/Masaüstü/ClientDir$ ./BibakBOXClient ../Client/B localhost 18232
bayram@bayram-SATELLITE-P855:~/Masaüstü/ClientDir$
```

Client localhosttan bağlantı yapıp, ../Client/B dizinini seçiyor. Bu dizin ve subdizinleri server'a socket aracılığıyla iletilir.



Client'ın seçtiği directory

Bu işlemin sonucunda B directorysi ve bütün subdirectoryleri ile birlikte Server directorysi olan A directorysi içine kopyalanır. Ayrıca eklenen her bir directory'nin altındada file bilgileri ve güncelleme işlemlerini tutan bir log dosyası bulunmaktadır. Log dosyaları hakkındada bilgi vereceğim.



Serverdaki A directorysine kopyalandı.

## 2) İki Client-Server Bağlantı Aynı Directory Adı İle

Aynı klasör için iki farklı client bağlantı gerçekleştirir ise son güncellemeye göre filelar update edilir. Her iki clientta olan filelar server klasörümüzde bulunur. Burada dikkat edilmesi gereken nokta socket'ten okuma ve log dosyasına bilgileri yazma gibi operasyonların birbirleriyle karışmaması gerekir. Bunun için mutex ve condition variablelar kullandım.

### Client1

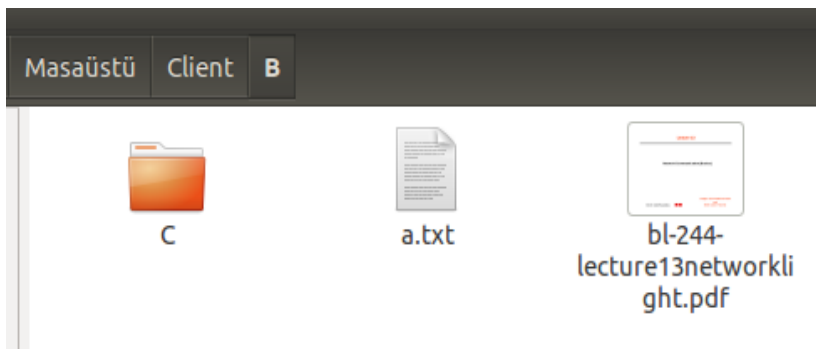
```
bayram@bayram-SATELLITE-P855: ~/Masaüstü/ClientDir
bayram@bayram-SATELLITE-P855:~$ cd Masaüstü/ClientDir
bayram@bayram-SATELLITE-P855:~/Masaüstü/ClientDir$ make
gcc -c BibakBOXClient.c
gcc -o BibakBOXClient BibakBOXClient.o -lpthread
bayram@bayram-SATELLITE-P855:~/Masaüstü/ClientDir$ ./BibakBOXClient
../Client/B localhost 18232
```

../Client/B directory

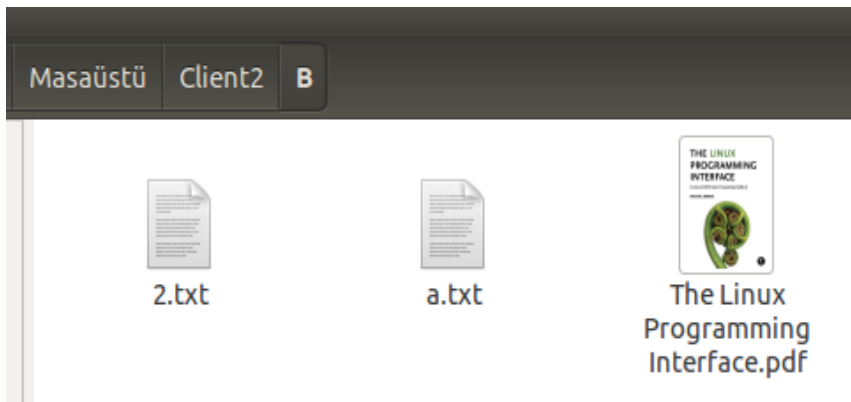
## Client2

```
bayram@bayram-SATELLITE-P855: ~/Masaüstü/ClientDir
bayram@bayram-SATELLITE-P855:~$ cd Masaüstü/ClientDir
bayram@bayram-SATELLITE-P855:~/Masaüstü/ClientDir$ make
gcc -o BibakBOXClient BibakBOXClient.o -lpthread
bayram@bayram-SATELLITE-P855:~/Masaüstü/ClientDir$ ./BibakBOXClient ../
Client2/B localhost 18232
```

../Client2/B directory

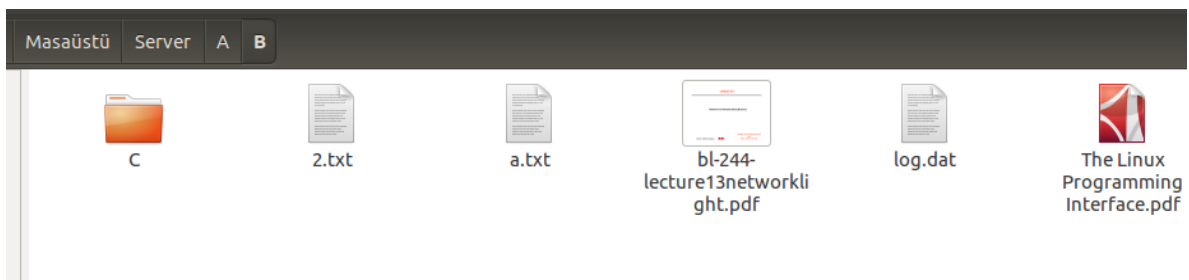


Client1 B dizini içeriği.



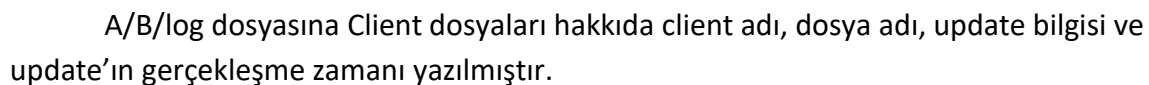
Client2 B dizini içeriği

## Server



Serverdaki A/B dizinin içeriği

Log dosyamızın iç görüntüsü ise;



Serverimiz sürekli bir client beklemesi gerektiğinden sonsuz döngüde bekler. SIGINT (CTRL C) sinyali gelene kadar döngü devam eder. Clientlarda offline olmuş olur.

CTRL-C ile Server sonlandırılır.

Multithreading, mutex ve condition variable, semaphore, sinyal handler, stream socket