

GEBZE TECHNICAL UNIVERSITY  
COMPUTER ENGINEERING

SYSTEM PROGRAMMING LECTURE

FINAL

FINAL REPORT

AHMET FURKAN KURBAN

1801042674

JUNE 17

## Design Decisions and Problem Solved Part:

### SERVER

After the threads are created in the server part, the server opens the socket and waits for the coming accepts connections. Threads are waiting at the synchronization barrier, just like the client. In order not to lose each incoming connection, we queue it. Since the queue object is global, we ensure synchronization with the mutex. After it is sent to the queue, we send a signal (not broadcast) to enable one of the threads to pull this connection from the thread and take action. With the received request, we look at which servant is working in which interval and if there is city information, we connect to the relevant servant and send the request. If there is no city information, we send the request to all servants and forward the incoming data to the client.

### SERVANT

According to the information given from the console in the server, we learn which city range to look at and read from the file. Then we save this data to the directory structure and hashmap. We get the city names as hashmap keys and the structure of the relevant city as value. First of all, the Servant connects to the server and sends its IP address, port number, information about which cities it is

responsible for, and its id. Then it opens a socket and waits for the connections to come from the server. It creates a new thread according to the incoming connects and handles that request. The necessary information is obtained by comparing the information of the cities where the incoming request and the servant are related, and it returns the answer to the server. The server takes the last result and sends it to the client.

## CLIENT

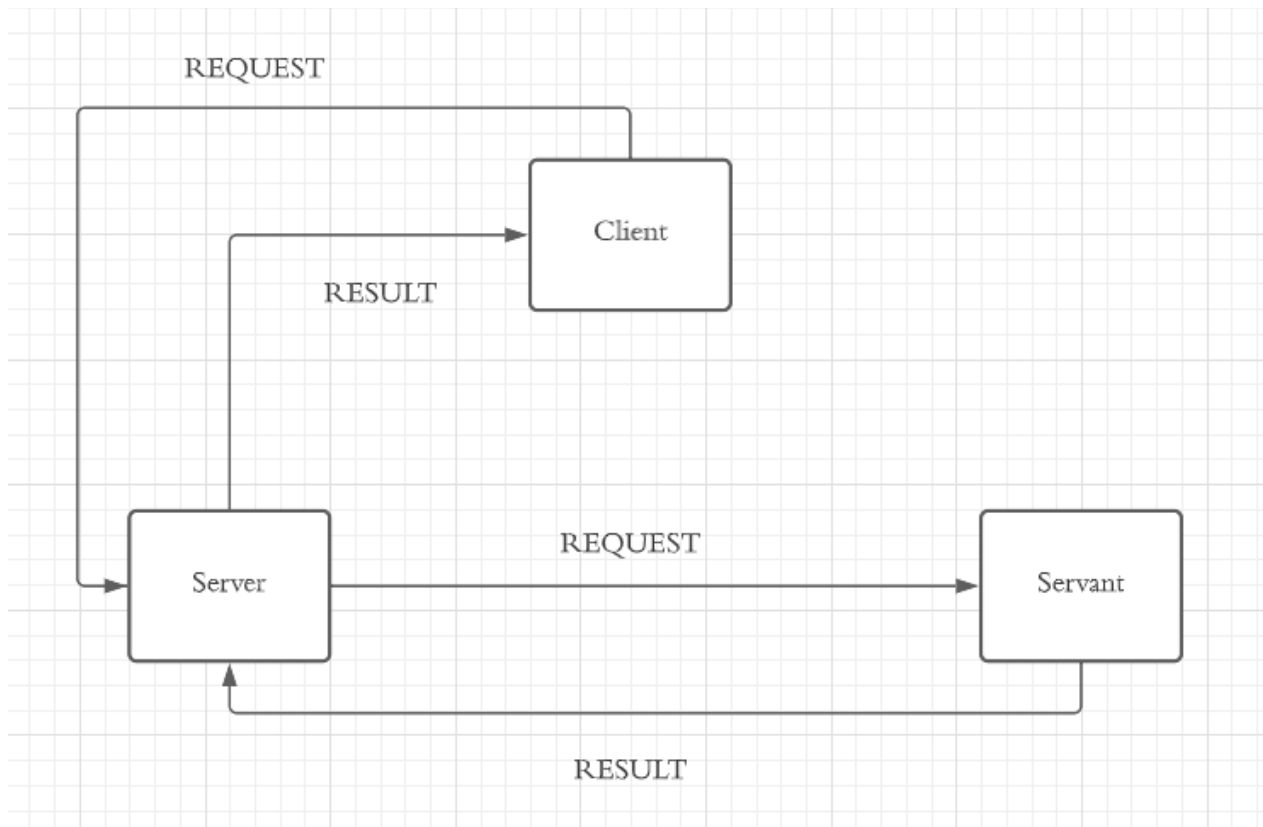
First of all, the request file was read in the client.c file and the data was saved. Then threads are created. The threads were kept in the synchronization barrier. After all the threads reached the barrier, they all worked at once. Then they made a connection with the server and forwarded the requests to the server and began to wait for the response.

### Which requirements I achieved:

I followed the instructions in the pdf. I learned sockets, I gained practicality in thread and mutex.

### Which requirements I failed:

There is memory leak because of the scandir function in servant.c file.



## RESULTS

```
Servant 384: loaded dataset, cities ADANA-ARDAHAN
Time 02:49:36:846 :Servant 384 present at port 46823 handling cities ADANA-ARDAHAN
Servant 384: listening at port 46823
Servant 387: loaded dataset, cities EDIRNE-HAKKARI
Servant 385: loaded dataset, cities ARTVIN-BITLIS
Servant 385: listening at port 37535
Time 02:49:36:846 :Servant 387 present at port 33301 handling cities EDIRNE-HAKKARI
Time 02:49:36:846 :Servant 385 present at port 37535 handling cities ARTVIN-BITLIS
Servant 387: listening at port 33301
Servant 390: loaded dataset, cities MALATYA-ORDU
Time 02:49:36:849 :Servant 390 present at port 49029 handling cities MALATYA-ORDU
Servant 390: listening at port 49029
Servant 389: loaded dataset, cities KASTAMONU-KUTAHYA
Servant 391: loaded dataset, cities OSMANIYE-SIVAS
Servant 389: listening at port 48075
Time 02:49:36:849 :Servant 389 present at port 48075 handling cities KASTAMONU-KUTAHYA
Servant 391: listening at port 50917
Time 02:49:36:849 :Servant 391 present at port 50917 handling cities OSMANIYE-SIVAS
Servant 392: loaded dataset, cities TEKIRDAG-ZONGULDAK
Servant 392: listening at port 39539
Time 02:49:36:852 :Servant 392 present at port 39539 handling cities TEKIRDAG-ZONGULDAK
Servant 388: loaded dataset, cities HATAY-KARS
Servant 388: listening at port 48273
Time 02:49:36:853 :Servant 388 present at port 48273 handling cities HATAY-KARS
Servant 386: loaded dataset, cities BOLU-DUZCE
Servant 386: listening at port 36623
Time 02:49:36:855 :Servant 386 present at port 36623 handling cities BOLU-DUZCE
Client: I have loaded 10 requests and I'm creating 10 threads.
Client-Thread-0: Thread-0 has been created
Client-Thread-0: I am requesting "transactionCount TARLA 01-01-2073 30-12-2074 ADANA"
Client-Thread-2: Thread-2 has been created
Client-Thread-3: Thread-3 has been created
Client-Thread-1: Thread-1 has been created
Client-Thread-4: Thread-4 has been created
Client-Thread-4: I am requesting "transactionCount FIDANLIK 02-09-2016 12-09-2081 BALIKESIR"
Client-Thread-5: Thread-5 has been created
```

