CSE 344 – SYSTEM PROGRAMMING

Final Project

151044072 – ALPER YAŞAR

Purpose of this Project?

The idea is creating one server and multiple servant, connecting them using socket.

Client sending question to server and server connecting with servant which has it. Sending question to servant from server and servant return the response. Server return all response to client end of the task.

Arguments

Each file starting with checking arguments. If arguments are not correct then giving error.

"./server -p 33000 -t 11"

"./servant -d ../dataset -c 1-9 -r 127.0.0.1 -p 33000"

"./client -r ./requestFile -q 33000 -s 127.0.0.1"

Server

Server starting with creating port for listening it.

After the creation of port then create threads as giving. Thread call thread function and wait mutex. If anything came port then add info to queue. If there is available thread for run then unlock the mutex and thread start to run. If there is not available thread waiting in the queue. Thread taking it from queue and call new function.

void initializeQueue()

Initialization of queue. When new stream come to port adding it to queue and If thread available then take it from queue and run it.

void initilializeMutexesAndConds()

Initializing mutex and conditions for thread synchronization.

static void *threadFunc(void *arg)

When mutex unlock then take queue and start new function. Argument for thread number.

void func(int sockfd,int tempPort,int index)

sockfd: socket number for reading and writing.

tempPort: created new port for servant. First thread start 16000 and go on each thread as incremented.

Index: thread number

Firstly function checking the arrived request is came from servant or client.

If servant then create new port for it.

If client connecting with right servant for response.

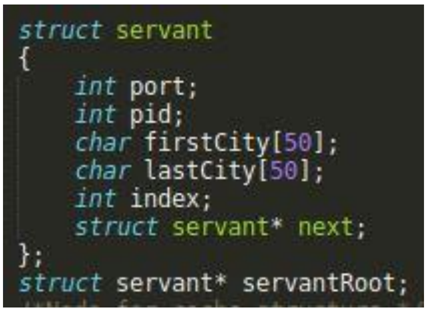
int createServant(int index,char* token,int tempPort)

tempPort: created new port for servant. First thread start 16000 and go on each thread as incremented.

Index: thread number

token: stream of coming from servant. This is stream is information about servant (pid, first city and last city).

Taking all this information to struct servant.



```
struct servant
{
    int port;
    int pid;
    char firstCity[50];
    char lastCity[50];
    int index;
    struct servant* next;
};
struct servant* servantRoot;
```

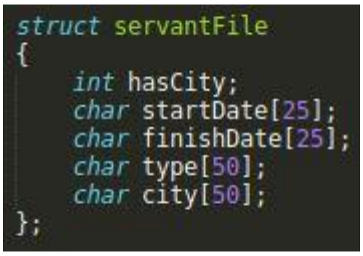
All servant creating and adding to list.

I used linked list data structure for hold the information.

int connectServant(char* str)

str: stream of coming from client. This stream is include of request. (first date, last date, type and city).

Tokenize it into servantFile structure.



```
struct servantFile
{
    int hasCity;
    char startDate[25];
    char finishDate[25];
    char type[50];
    char city[50];
};
```

After tokenization then checking stream has city or not. If has city then searching range of city. If it has city then sending special servant or sending all servant.

int comp2City(char* str1,char* str2)

Compare 2 city for finding range.

int sendData(struct servantFile srvF,struct servant srv)

This function take 2 structure. Connecting to servant port and sending servant file to it. Then waiting it's respond.

Then return the respond.

SIGINT

When SIGINT come then closing all files and freeing all memory. Then sending kill to servant pids.

Servant

Servant starting read directories and files as giving.

And communication with server it is like created in server.

void readFiles(char *path)

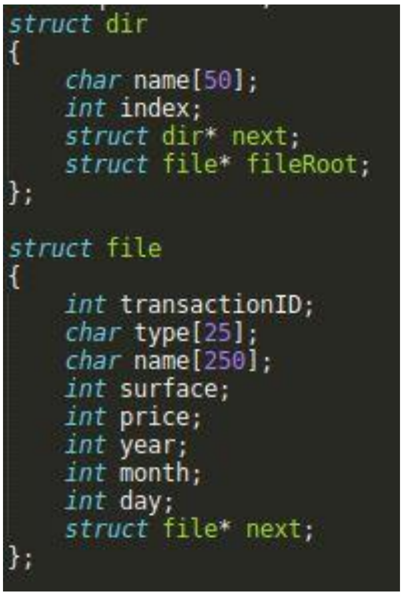
This function reading directories giving path in arguments.

void takeToStruct()

Reading all directories to dir struct as linked list. All dir struct holding files in list.

struct file *files(char* path)

This firstly opening file and reading each row to new list. Year, month, day coming from file name.



```
struct dir
{
    char name[50];
    int index;
    struct dir* next;
    struct file* fileRoot;
};

struct file
{
    int transactionID;
    char type[25];
    char name[250];
    int surface;
    int price;
    int year;
    int month;
    int day;
    struct file* next;
};
```

void listening(int index)

After the read main function communicate with server and taking info for creating new port. After the creation new port servant working like new server. Waiting information come. This system work like server mentioned.

System working like server. Initialize queues, mutex and conditions. Waiting port to read and call the func().

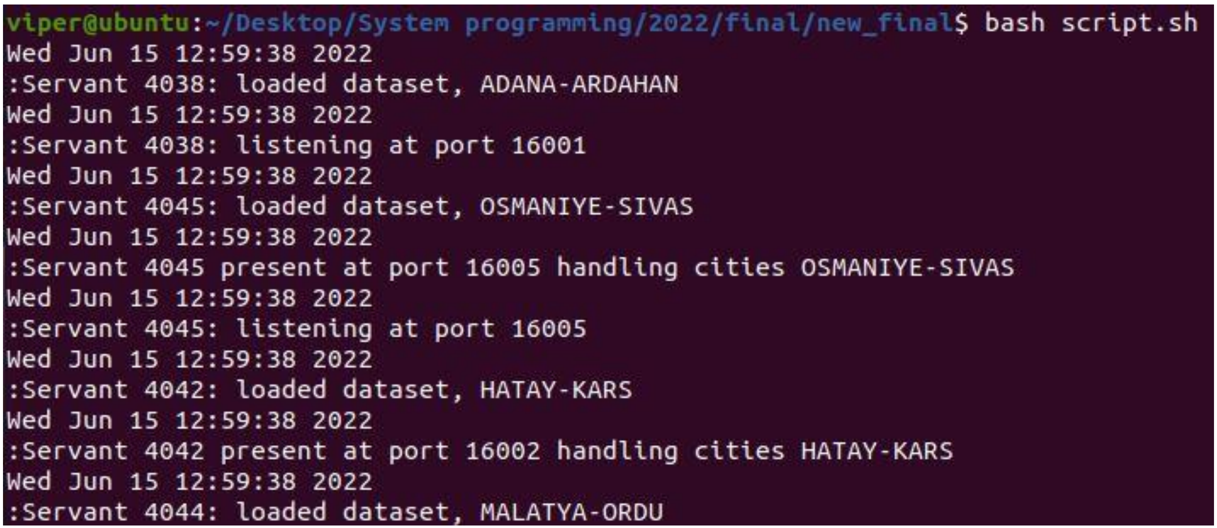
This function searching all list and checking types. If types matching then increasing the counter and return end of the list.

When SIGINT come from server closing all files and freeing all memory.

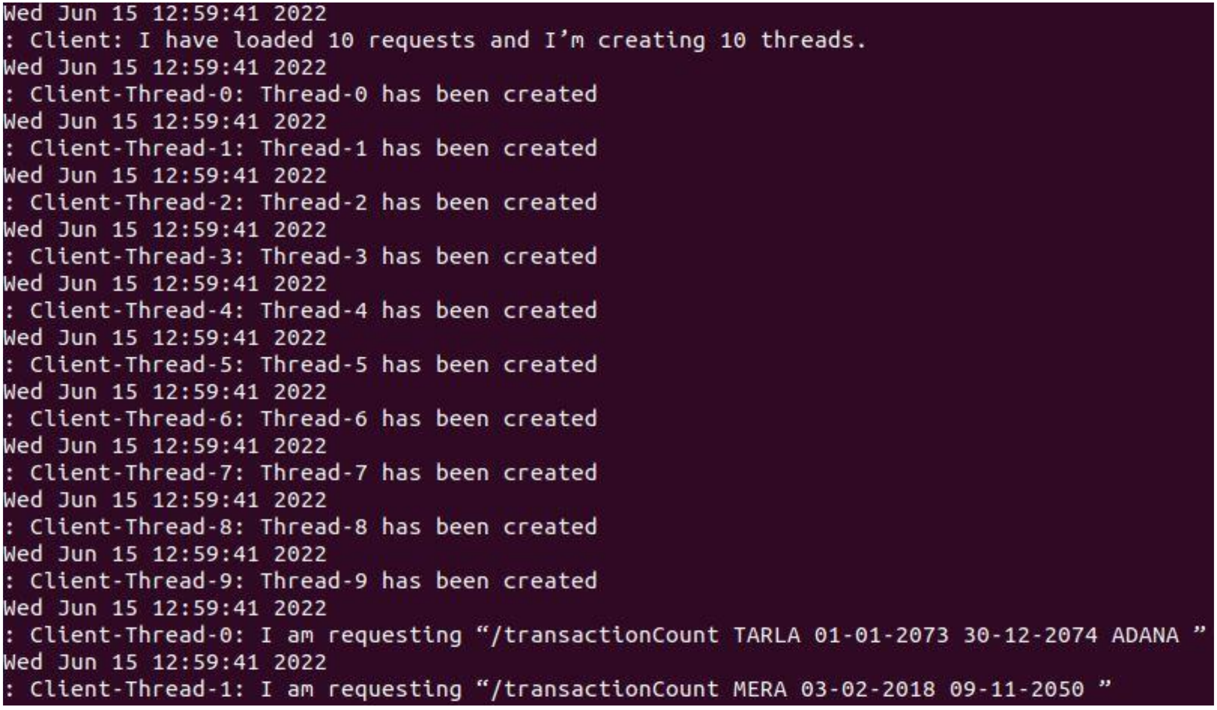
Client

Reading each row of file to string. Sending to server and waiting the respond.

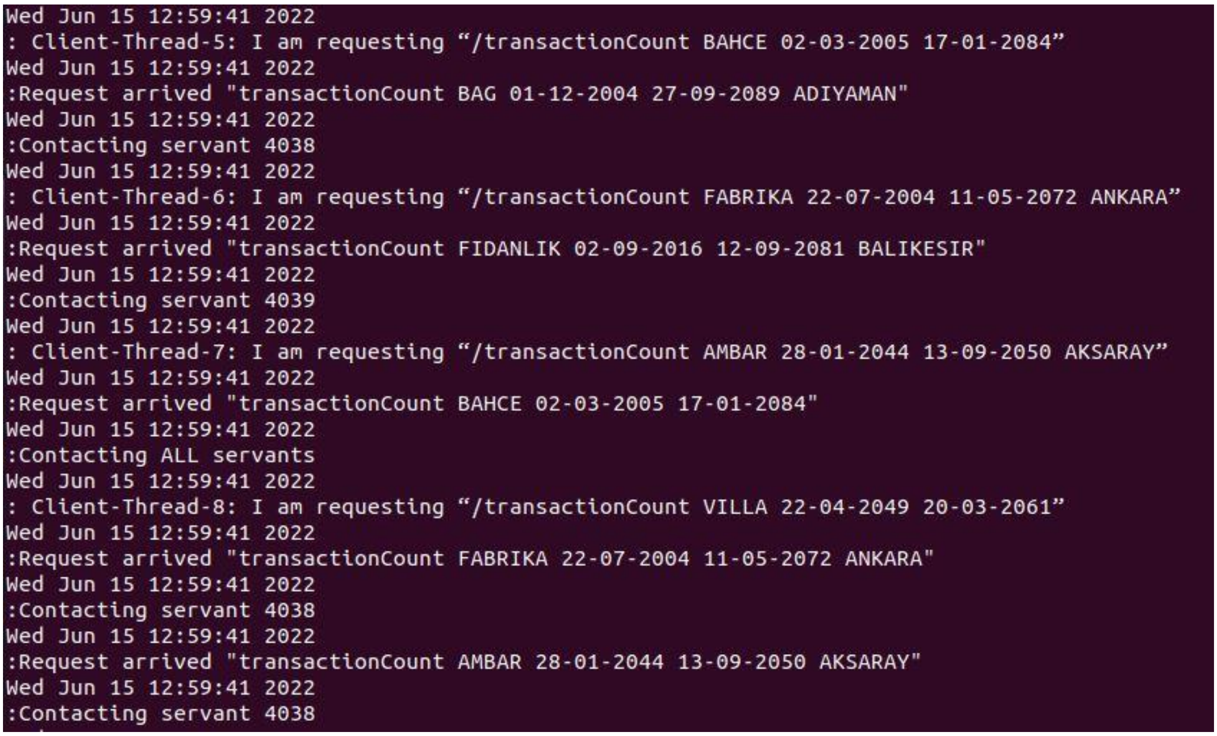
OUTPUTS



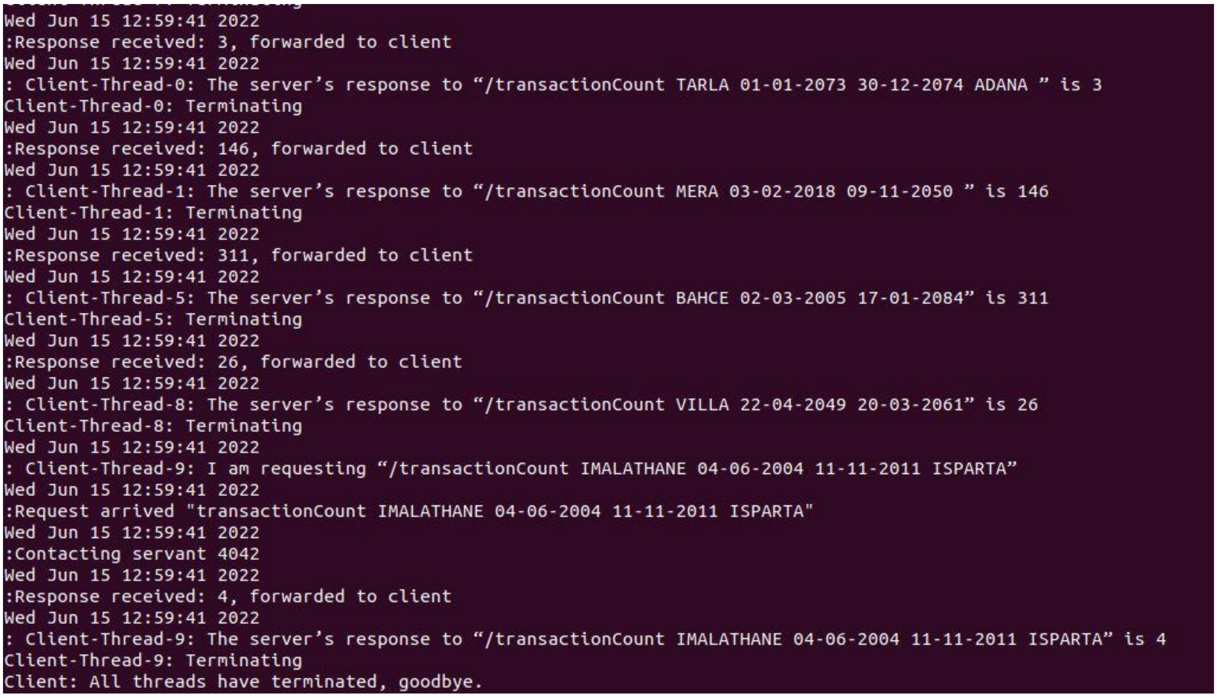
```
viper@ubuntu:~/Desktop/System programming/2022/final/new_final$ bash script.sh
Wed Jun 15 12:59:38 2022
:Servant 4038: loaded dataset, ADANA-ARDAHAN
Wed Jun 15 12:59:38 2022
:Servant 4038: listening at port 16001
Wed Jun 15 12:59:38 2022
:Servant 4045: loaded dataset, OSMANIYE-SIVAS
Wed Jun 15 12:59:38 2022
:Servant 4045 present at port 16005 handling cities OSMANIYE-SIVAS
Wed Jun 15 12:59:38 2022
:Servant 4045: listening at port 16005
Wed Jun 15 12:59:38 2022
:Servant 4042: loaded dataset, HATAY-KARS
Wed Jun 15 12:59:38 2022
:Servant 4042 present at port 16002 handling cities HATAY-KARS
Wed Jun 15 12:59:38 2022
:Servant 4044: loaded dataset, MALATYA-ORDU
```



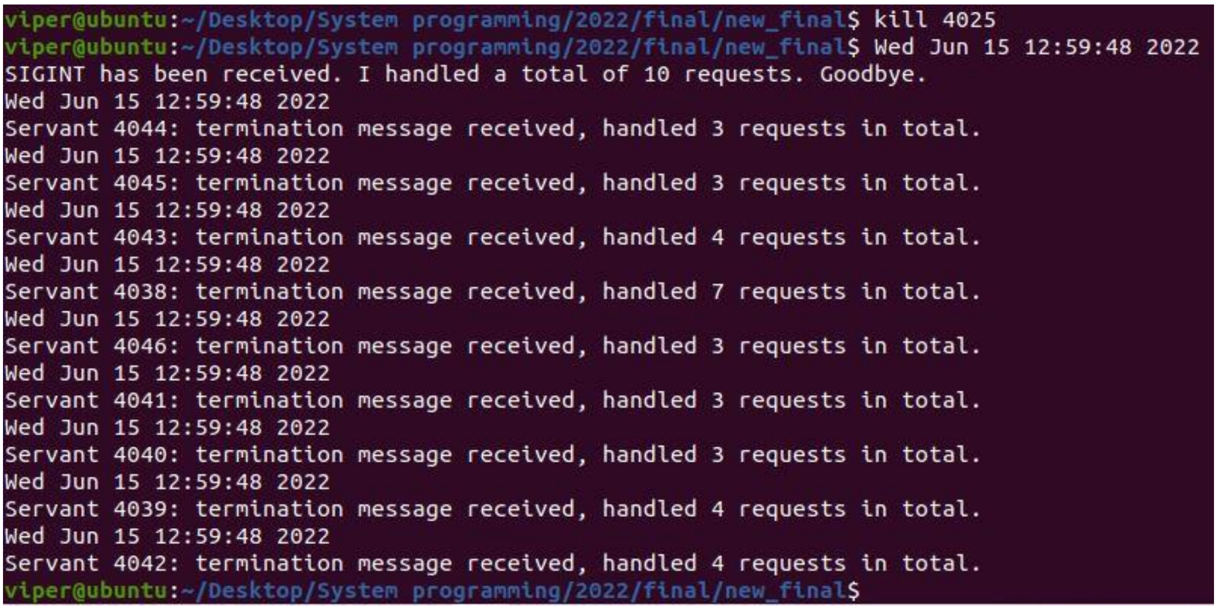
```
Wed Jun 15 12:59:41 2022
: Client: I have loaded 10 requests and I’m creating 10 threads.
Wed Jun 15 12:59:41 2022
: Client-Thread-0: Thread-0 has been created
Wed Jun 15 12:59:41 2022
: Client-Thread-1: Thread-1 has been created
Wed Jun 15 12:59:41 2022
: Client-Thread-2: Thread-2 has been created
Wed Jun 15 12:59:41 2022
: Client-Thread-3: Thread-3 has been created
Wed Jun 15 12:59:41 2022
: Client-Thread-4: Thread-4 has been created
Wed Jun 15 12:59:41 2022
: Client-Thread-5: Thread-5 has been created
Wed Jun 15 12:59:41 2022
: Client-Thread-6: Thread-6 has been created
Wed Jun 15 12:59:41 2022
: Client-Thread-7: Thread-7 has been created
Wed Jun 15 12:59:41 2022
: Client-Thread-8: Thread-8 has been created
Wed Jun 15 12:59:41 2022
: Client-Thread-9: Thread-9 has been created
Wed Jun 15 12:59:41 2022
: Client-Thread-0: I am requesting “/transactionCount TARLA 01-01-2073 30-12-2074 ADANA ”
Wed Jun 15 12:59:41 2022
: Client-Thread-1: I am requesting “/transactionCount MERA 03-02-2018 09-11-2050 ”
```



```
Wed Jun 15 12:59:41 2022
: Client-Thread-5: I am requesting “/transactionCount BAHCE 02-03-2005 17-01-2084”
Wed Jun 15 12:59:41 2022
:Request arrived "transactionCount BAG 01-12-2004 27-09-2089 ADIYAMAN"
Wed Jun 15 12:59:41 2022
:Contacting servant 4038
Wed Jun 15 12:59:41 2022
: Client-Thread-6: I am requesting “/transactionCount FABRIKA 22-07-2004 11-05-2072 ANKARA”
Wed Jun 15 12:59:41 2022
:Request arrived "transactionCount FIDANLIK 02-09-2016 12-09-2081 BALIKESIR"
Wed Jun 15 12:59:41 2022
:Contacting servant 4039
Wed Jun 15 12:59:41 2022
: Client-Thread-7: I am requesting “/transactionCount AMBAR 28-01-2044 13-09-2050 AKSARAY”
Wed Jun 15 12:59:41 2022
:Request arrived "transactionCount BAHCE 02-03-2005 17-01-2084"
Wed Jun 15 12:59:41 2022
:Contacting ALL servants
Wed Jun 15 12:59:41 2022
: Client-Thread-8: I am requesting “/transactionCount VILLA 22-04-2049 20-03-2061”
Wed Jun 15 12:59:41 2022
:Request arrived "transactionCount FABRIKA 22-07-2004 11-05-2072 ANKARA"
Wed Jun 15 12:59:41 2022
:Contacting servant 4038
Wed Jun 15 12:59:41 2022
:Request arrived "transactionCount AMBAR 28-01-2044 13-09-2050 AKSARAY"
Wed Jun 15 12:59:41 2022
:Contacting servant 4038
```



```
Wed Jun 15 12:59:41 2022
:Response received: 3, forwarded to client
Wed Jun 15 12:59:41 2022
: Client-Thread-0: The server’s response to “/transactionCount TARLA 01-01-2073 30-12-2074 ADANA ” is 3
Client-Thread-0: Terminating
Wed Jun 15 12:59:41 2022
:Response received: 146, forwarded to client
Wed Jun 15 12:59:41 2022
: Client-Thread-1: The server’s response to “/transactionCount MERA 03-02-2018 09-11-2050 ” is 146
Client-Thread-1: Terminating
Wed Jun 15 12:59:41 2022
:Response received: 311, forwarded to client
Wed Jun 15 12:59:41 2022
: Client-Thread-5: The server’s response to “/transactionCount BAHCE 02-03-2005 17-01-2084” is 311
Client-Thread-5: Terminating
Wed Jun 15 12:59:41 2022
:Response received: 26, forwarded to client
Wed Jun 15 12:59:41 2022
: Client-Thread-8: The server’s response to “/transactionCount VILLA 22-04-2049 20-03-2061” is 26
Client-Thread-8: Terminating
Wed Jun 15 12:59:41 2022
: Client-Thread-9: I am requesting “/transactionCount IMALATHANE 04-06-2004 11-11-2011 ISPARTA”
Wed Jun 15 12:59:41 2022
:Request arrived "transactionCount IMALATHANE 04-06-2004 11-11-2011 ISPARTA"
Wed Jun 15 12:59:41 2022
:Contacting servant 4042
Wed Jun 15 12:59:41 2022
:Response received: 4, forwarded to client
Wed Jun 15 12:59:41 2022
: Client-Thread-9: The server’s response to “/transactionCount IMALATHANE 04-06-2004 11-11-2011 ISPARTA” is 4
Client-Thread-9: Terminating
Client: All threads have terminated, goodbye.
```



```
viper@ubuntu:~/Desktop/System programming/2022/final/new_final$ kill 4025
viper@ubuntu:~/Desktop/System programming/2022/final/new_final$ Wed Jun 15 12:59:48 2022
SIGINT has been received. I handled a total of 10 requests. Goodbye.
Wed Jun 15 12:59:48 2022
Servant 4044: termination message received, handled 3 requests in total.
Wed Jun 15 12:59:48 2022
Servant 4045: termination message received, handled 3 requests in total.
Wed Jun 15 12:59:48 2022
Servant 4043: termination message received, handled 4 requests in total.
Wed Jun 15 12:59:48 2022
Servant 4038: termination message received, handled 7 requests in total.
Wed Jun 15 12:59:48 2022
Servant 4046: termination message received, handled 3 requests in total.
Wed Jun 15 12:59:48 2022
Servant 4041: termination message received, handled 3 requests in total.
Wed Jun 15 12:59:48 2022
Servant 4040: termination message received, handled 3 requests in total.
Wed Jun 15 12:59:48 2022
Servant 4039: termination message received, handled 4 requests in total.
Wed Jun 15 12:59:48 2022
Servant 4042: termination message received, handled 4 requests in total.
viper@ubuntu:~/Desktop/System programming/2022/final/new_final$
```