

시스템 프로그래밍 실습

# [FTP 2-3]

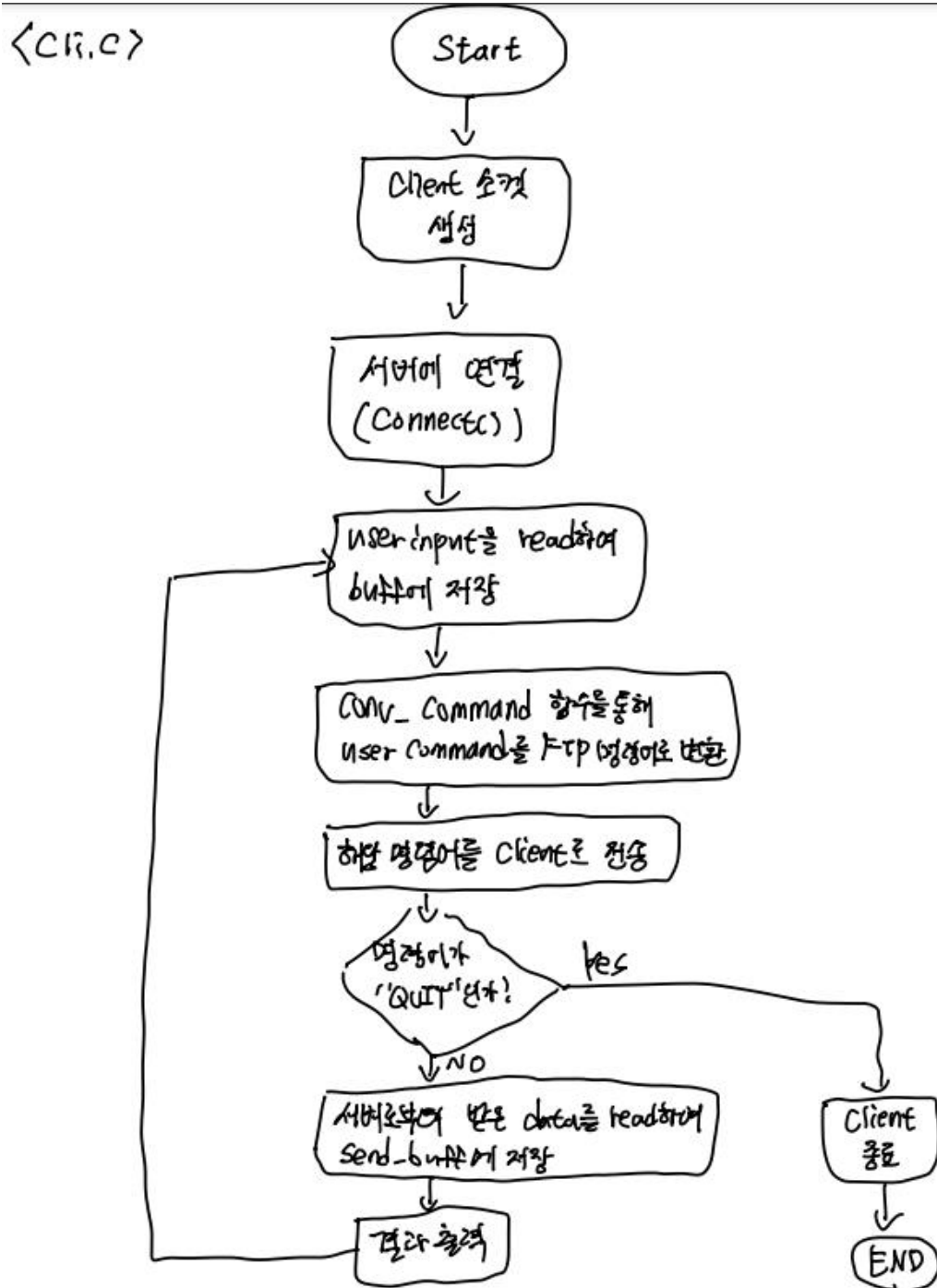
Class : D  
Professor : 최상호 교수님  
Student ID : 2020202092  
Name : 강현민

# Introduction

이번 [FTP 2-3](#) 과제는 FTP 서버 구현을 최종 목표로 하는 프로젝트의 두번째 단계인 소켓 프로그래밍으로, 이전까지의 클라이언트-서버 소켓 프로그래밍 및 명령어에 대한 동작 구현, 그리고 signal 과 관련된 구현들을 모두 합쳐 최종적으로 서버-클라이언트 구조의 FTP 서버의 동작을 구현하는 것을 목적으로 한다. 먼저 서버가 실행된 후, 클라이언트는 동일한 포트번호를 인자로하여 해당 서버에 접속한다. 그 후 user command 가 입력되면 해당 명령어가 FTP 명령어로 변환되어 서버로 전송되고, 서버에서는 이에 맞는 동작을 수행하여 해당 명령어에 대한 결과를 클라이언트로 전송하여 클라이언트에서 출력된다. 또한, 서버에서는 현재 연결되어 있는 클라이언트의 정보를 확인할 수 있으며, 클라이언트가 종료된 경우 연결 해제된 클라이언트의 정보는 서버에서 제거되고 이에 대한 정보를 확인할 수 있다.

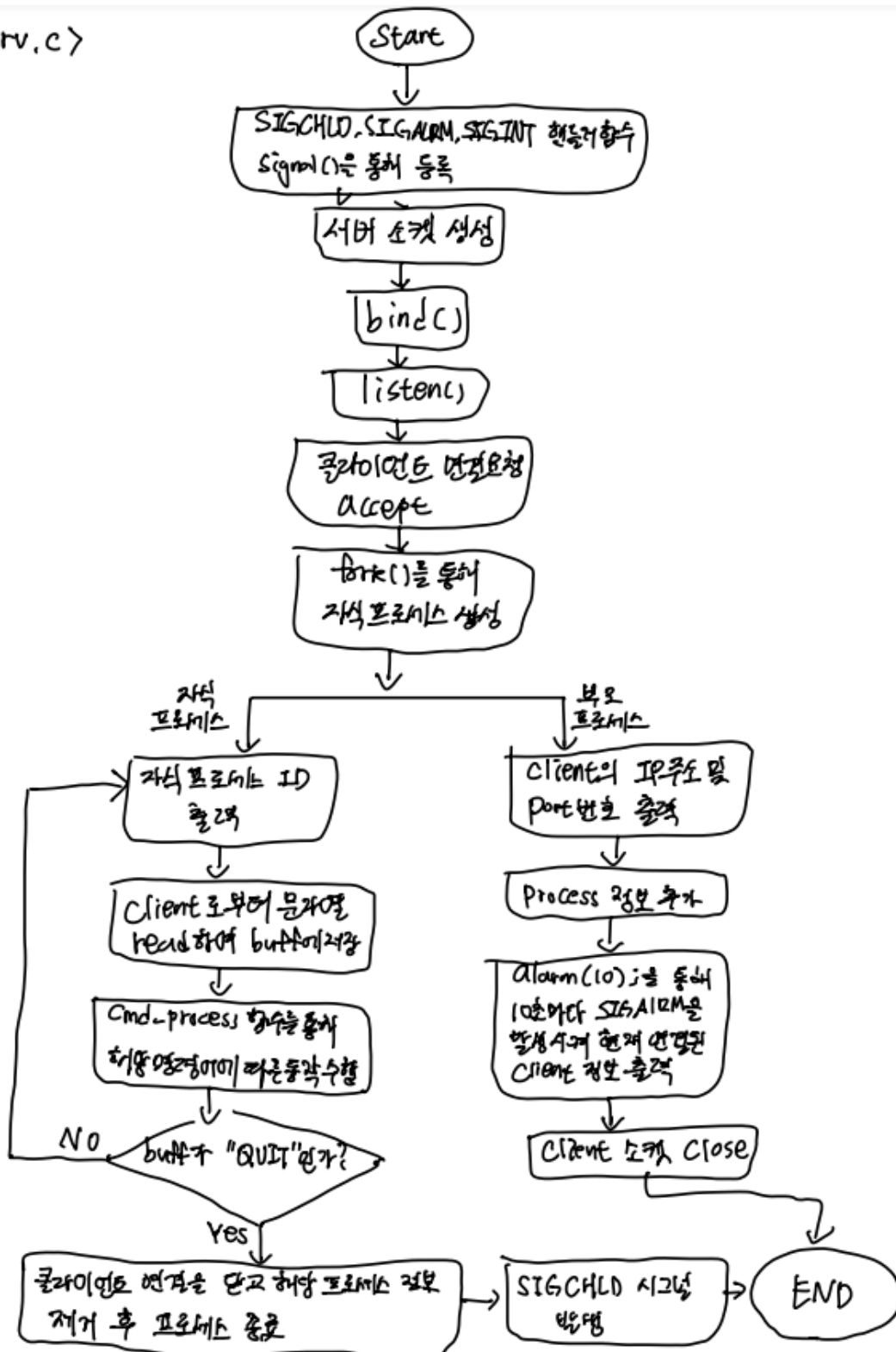
# Flow chart

<cli.c>



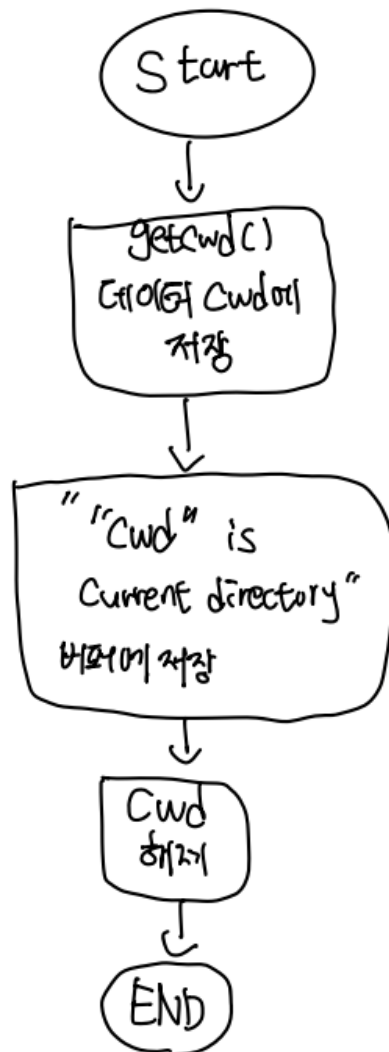
<srv.c>

<srv.c>

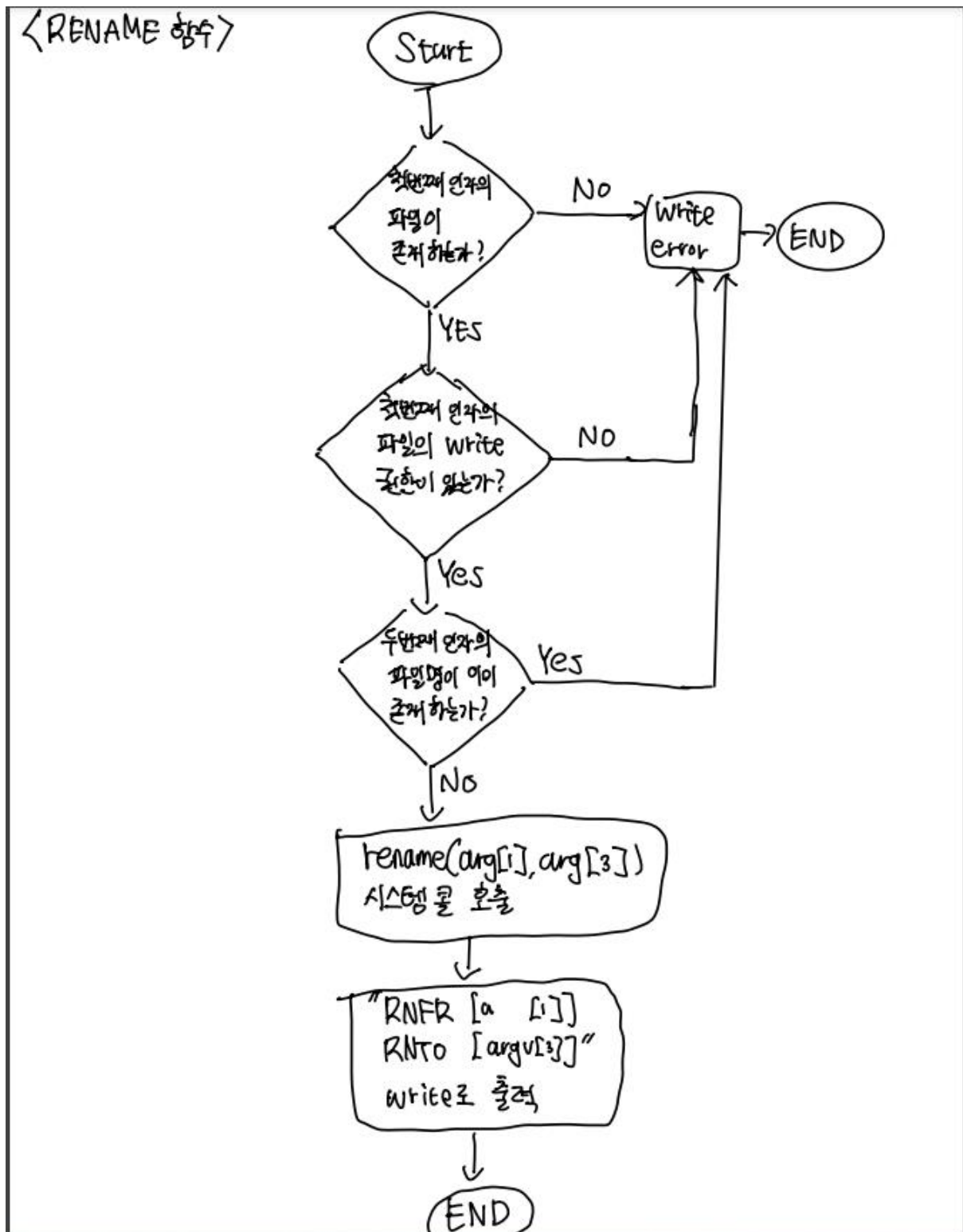


<PWD 함수>

<PWD 함수>

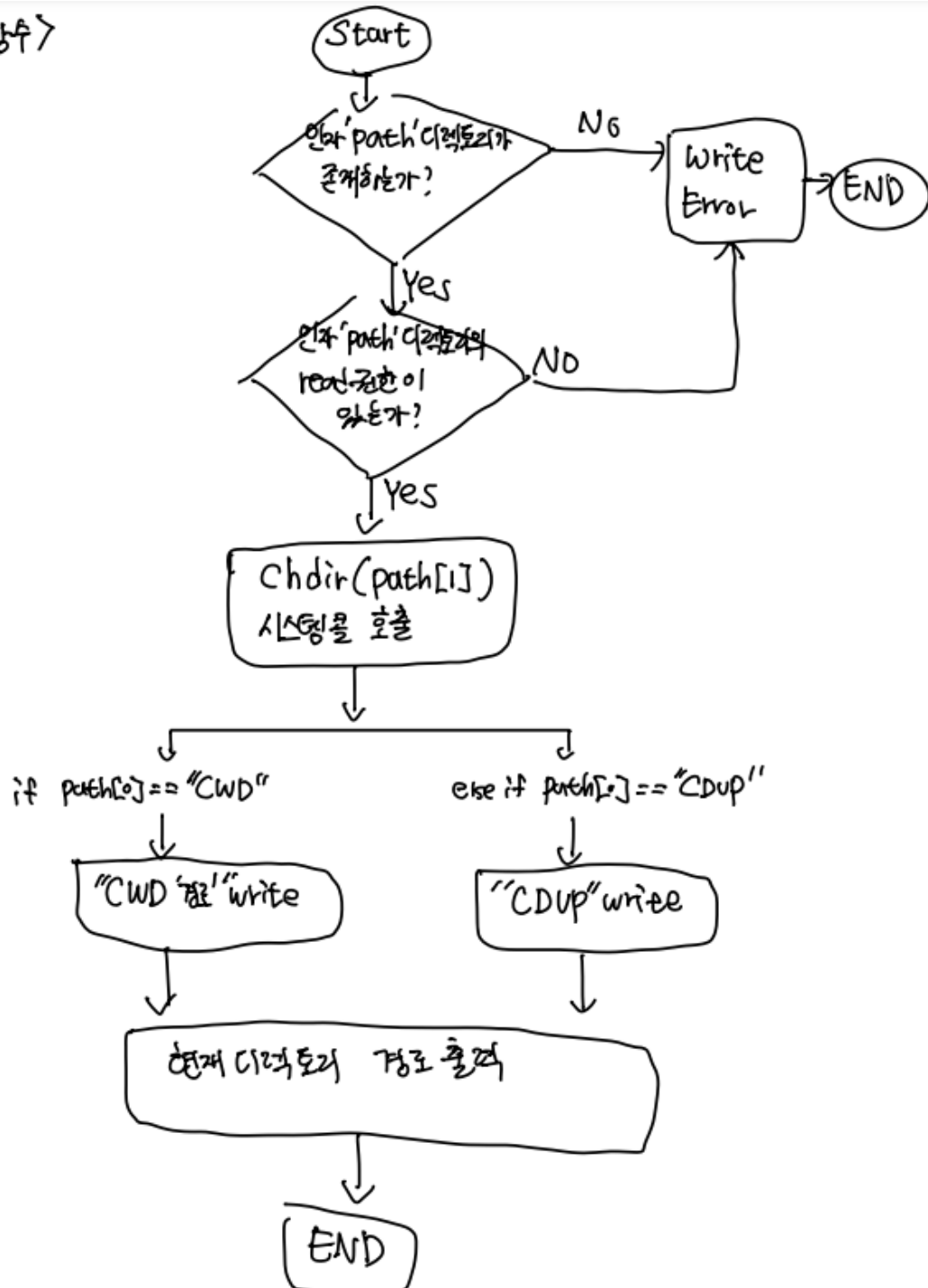


<RENAME 함수>



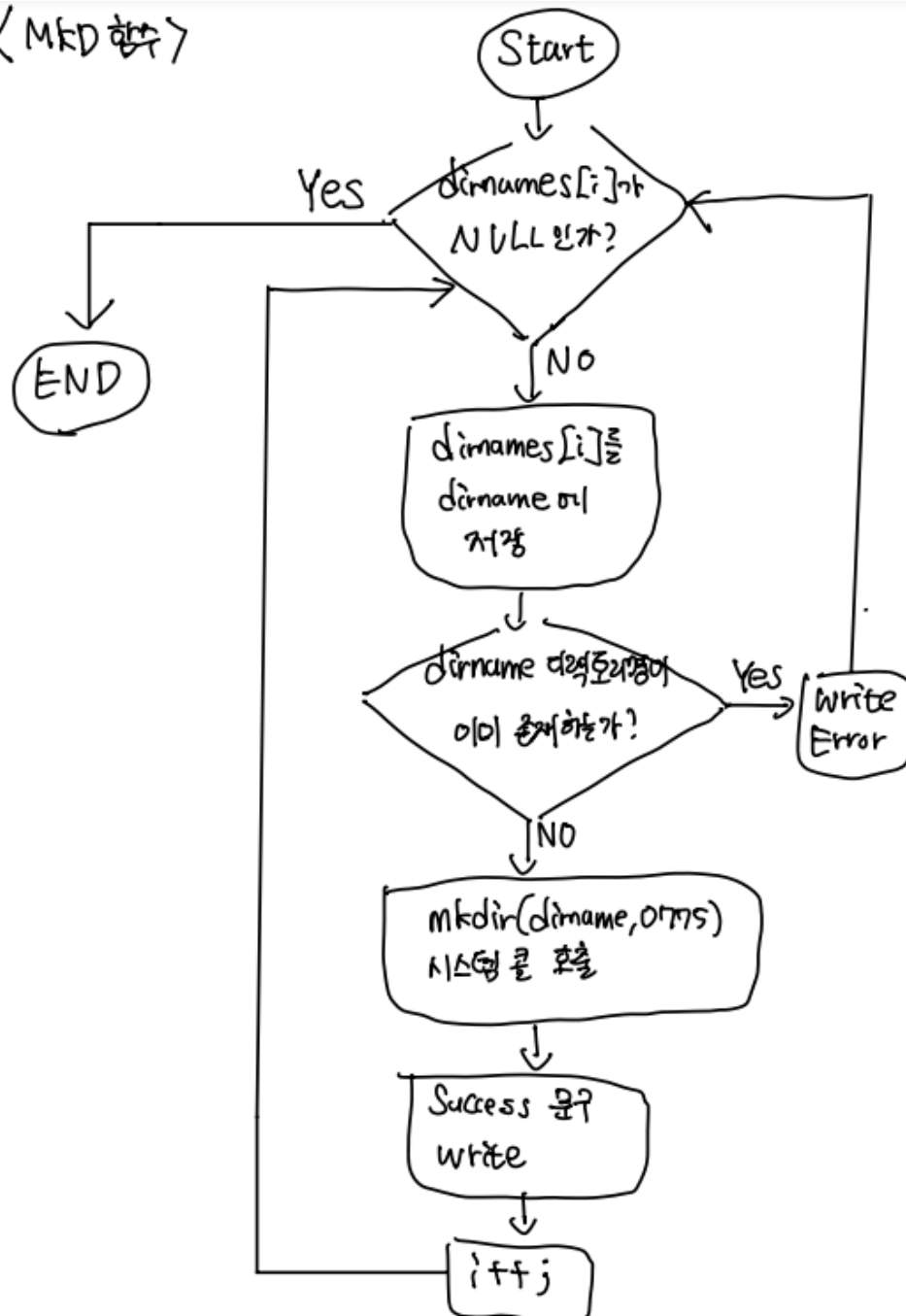
<CD 함수>

<CD 함수>



<MKD 함수>

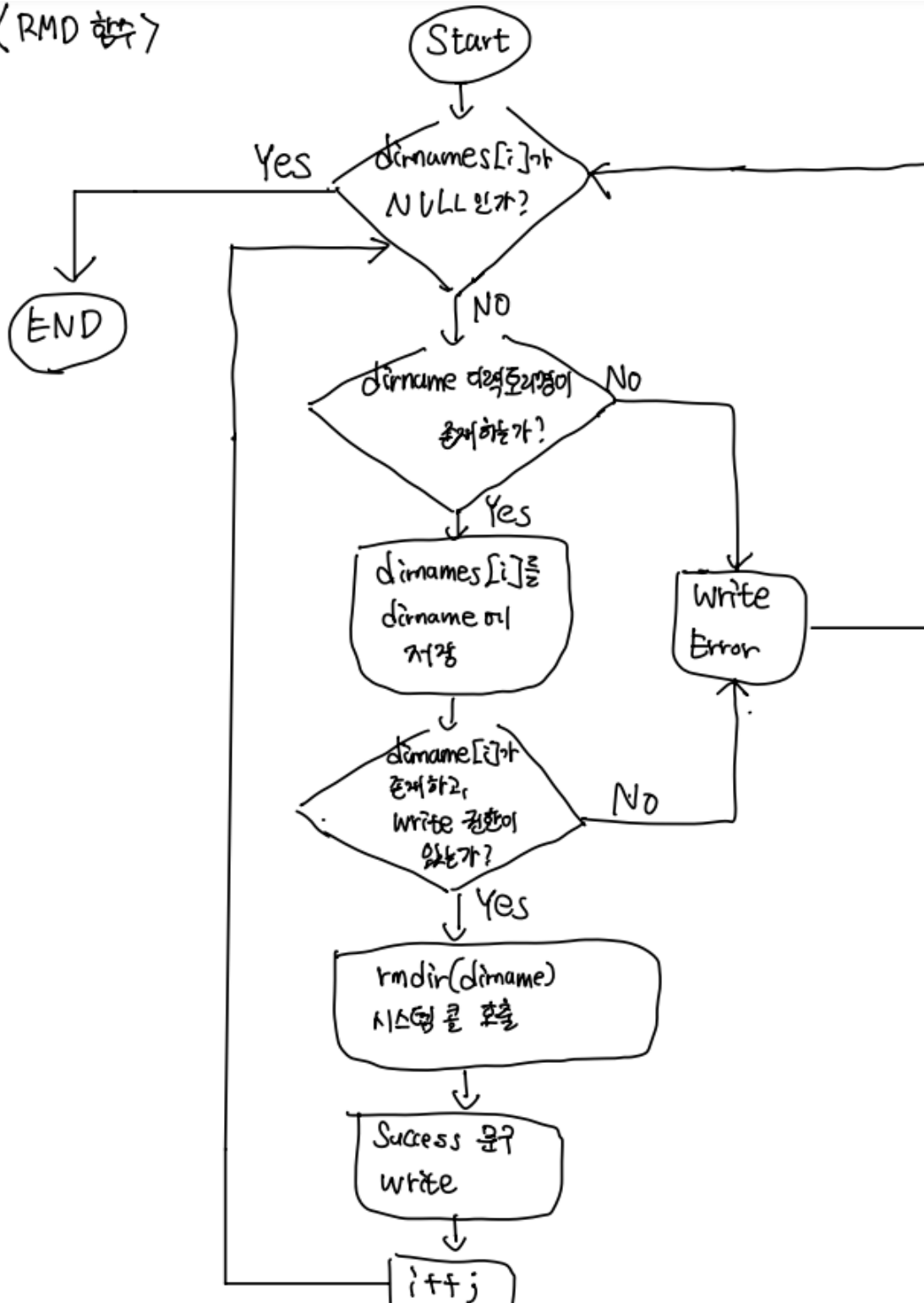
< MKD 함수 >





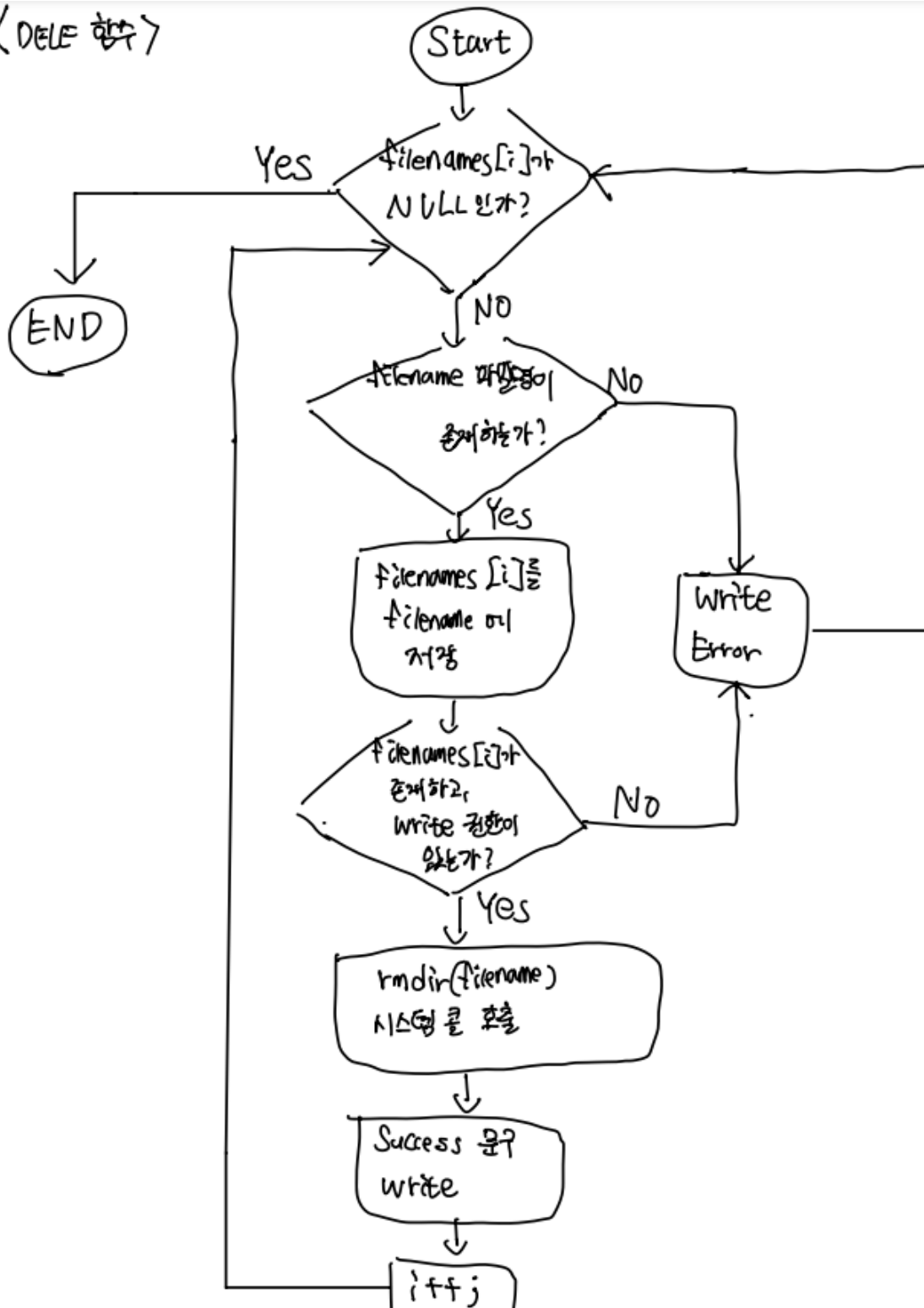
<RMD 함수>

<RMD 함수>



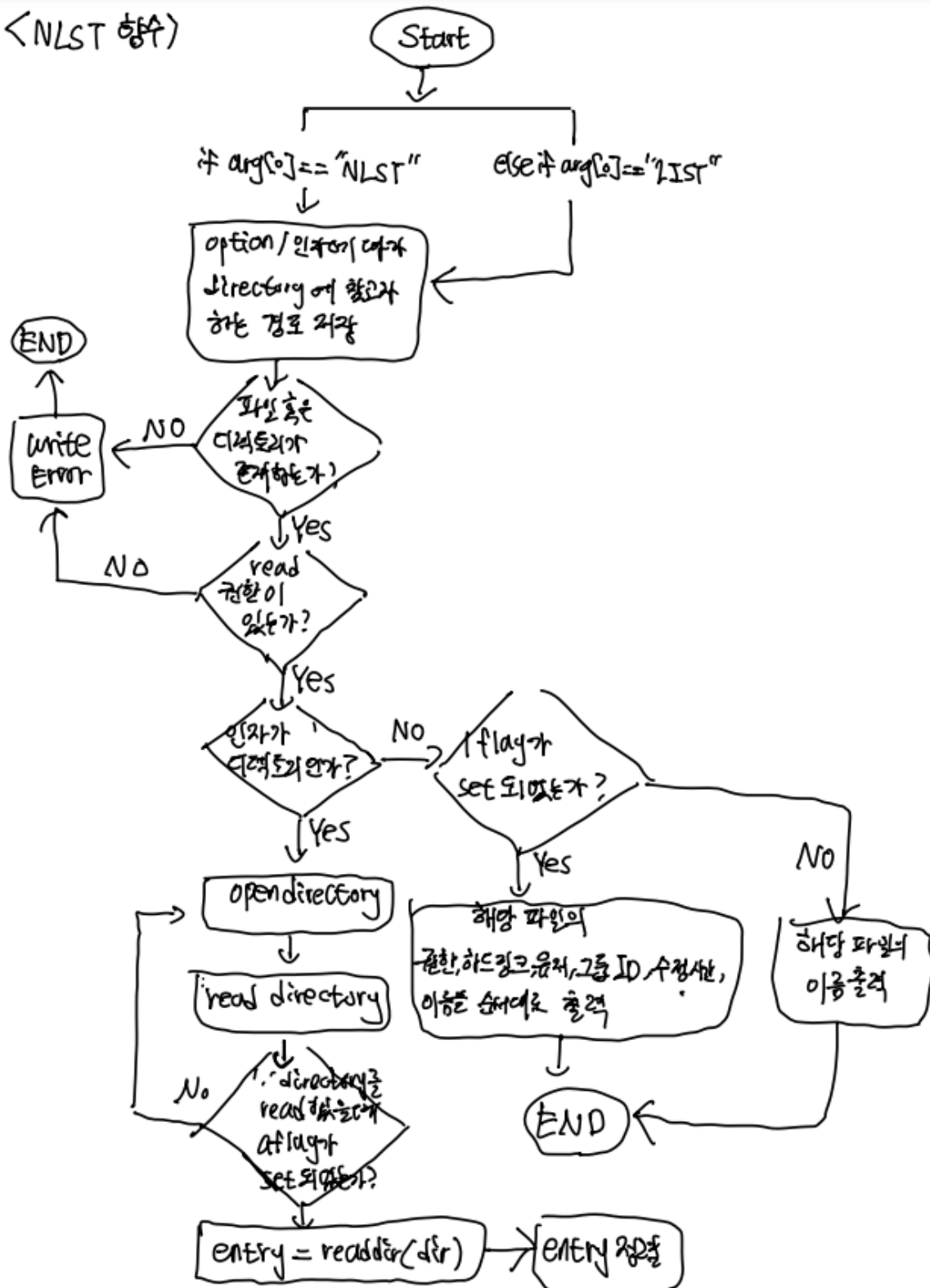
<DELE 함수>

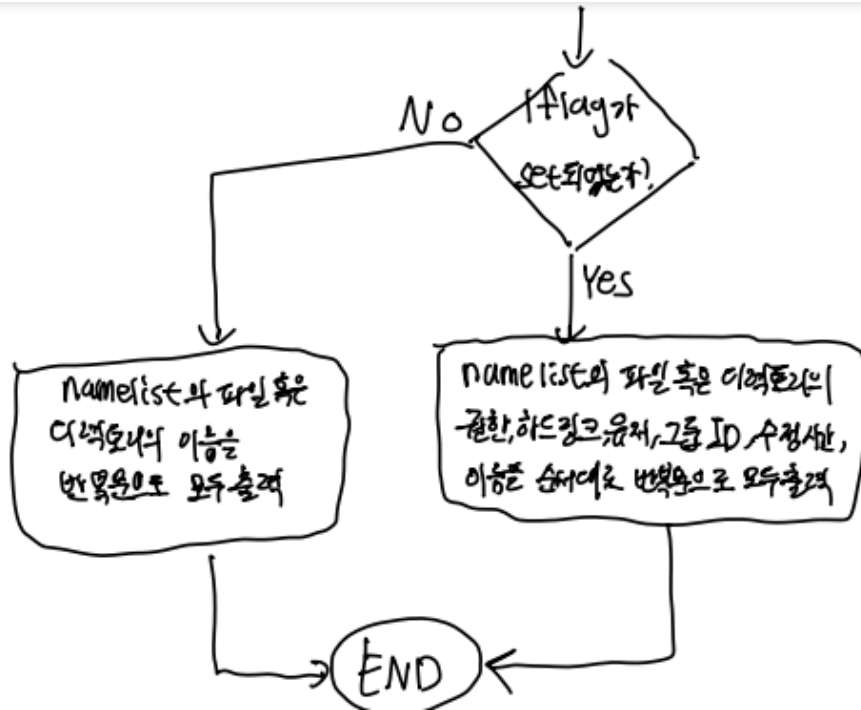
<DELE 함수>



<NLST 함수>

<NLST 함수>





# Pseudo code

<cli.c>

Int conv\_command(char\* buff, char\* send\_buff);

Int main(int argc, char \*\*argv)

    Socket(AF\_INET, SOCK\_STREAM, 0);

    Store information of server information to sockaddr\_in struct

    Connect to server

    While(1)

        Read data from server and store it in buff

        Conv\_command;

        Write FTP command to server

        Read the result from server and print it

        Close the socket

        Return 0;

Int conv\_command (char \*buff, char\* send\_buff)

    Tokenize input data

    If token == ls

        Convert "ls" to "NLST"

    Else if token == "dir"

        Conver "dir" to LIST

    Else if token == cd

        If argument is ".."

Conver cd to CDUP

Else

Convert cd to CWD

Else if token == quit

Convert quit to QUIT

Else if token == pwd

Convert pwd to PWD

Else if token == mkdir

Convert mkdir to MKD

Else if token == rmdir

Convert rmdir to RMD

Else if token == delete

Convert delete to DELE

Else if token == rename a b

Convert rename a b to "RNFR a RNT0 b"

<srv.c>

Int main(int argc, char\*\* argv)

    Declare server and client addr struct

    Apply SIGCHLD, SIGALRM and SIGINT signal handlers

    Store information of server into structure

    Bind the socket

    Listen for incoming connections

    Alarm(10);

    While(1)

        Pid\_t pid = fork();

        If child process

            Read command from current client

            If fail

                Print "Client(pid)'s Release"

        If command is "QUIT"

            Print "Client(pid)'s Release"

            Remove process information

            Close client\_fd

            Exit(0);

        Cmd\_process(buff, result\_buff);

        Write result to client

        Clear result\_buff

Close client\_fd

Exit(0);

Else if parent process

Print client information

Add process information

Alarm(10);

Close client\_fd

Void add\_process\_info(pid\_t pid, int port)

Store process information to process\_list[]

Void remove\_process\_info(pid\_t pid)

Remove process information from process\_list[]

void sh\_chld(int signum)

remove\_process\_info(terminated pid);

void sh\_alrm(int signum)

print\_process\_info();

alarm(10);

void cmd\_process(const char\* buff, char result\_buff[BUFSIZE])

if command is PWD

if there is argument

if option is inputted

write error

else write error



```
        else call PWD()

    if arg[0] == RNFR and arg[1] == RNTD
        write error else

    else if arg[0] == RNFR and arg[2] == RNTD
        call RENAME(arg)

else if command is CWD
    if there is argument
        if option is inputted
            write error

        else
            call CD(arg);

    else write error

if command is CDUP
    if there is argument
        if option is inputted
            write error

        else call CD(arg);

    else write error

if command is MKD
    if there is argument
        if option is inputted
            write error

        else call MKD(arg);

    else write error
```

if command is RMD

if there is no argument

if option is inputted

write error

else call RMD(arg)

else write error

if command is DELE

if there is no argument

if option is inputted

write error

else call DELE(arg);

else write error

if command is QUIT

if there is no argument

if option is inputted

write error

else call CD(arg);

else write error

if command is NLST

declare int aflag, lflag = 0;

if(there is no argument)

write NLST

call NLST(arg, 0, 0)

else if there is option

if option is -a

    aflag++;

    write NLST -a

    call NLST(arg, aflag, lflag);

else if option is -l

    lflag++;

    write NLST -l

    call NLST(arg, aflag, lflag);

else if option is -al or -la

    aflag++;

    lflag++;

    write NLST -al

    call NLST(arg, aflag, lflag);

else write error

    exit(0);

else if command is LIST

    if there is argument

        if option is inputted

            write error

        else call NLST(arg, 1, 1);

    else call NLST(arg, 1, 1);

Void PWD()

Cwd = current working directory

Write cwd to standard output

Deallocate cwd

Void RENAME(char\*\* arg)

If FR file is exist and not have write permission

write error (no access)

If FR file is not exist

write error

If TO file already exists

write error

Rename(FR, TO)

write "RNFR [FR] RNT0 [TO]" to standard output

void CD(char\*\* path)

if path is exist and not have read permission

write error

if (chdir(path[1]) is failed

write error

else

if command is CWD

write "CWD path[1]" to standard output

```

        else if command is CDUP
            write "CDUP" to standard output

void MKD(char** dirnames)
    while dirnames[i] is NULL
        if there is no arguments
            write error
        else
            if dirnames[i] already exists
                write error
            write MKD success message
        i++

void RMD(char** dirnames)
    while dirnames[i] is NULL
        if there is no arguments
            write error
        else
            if dirnames[i] does not exists
                write error
            write RMD success message
        i++

void DELE(char** filenames)
    while filenames [i] is NULL
        if there is no arguments

```

```

        write error
    else
        if filenames [i] does not exists
            write error
        write DELE success message
        i++
void NLST(char** arg, int aflag, int lflag)
    if command is NLST
        if there is no argument
            directory = current directory
        else if option is inputted
            directory = arg[2];
        else if option is not inputted
            directory = arg[1];
    else if command is LIST
        if there is no argument
            directory = current directory
        else if argument is inputted
            directory = arg[1];
    if arg is directory
        opendir(directory)

```

```
while readdir(dir) != NULL

    readdir(dir)

    store datas into namelist


    if aflag == 1

        write file type / permission

        write number of hard links

        write owner name

        write group name

        write file size

        write file modification time

        write file or directory name

    else write file name

else

    readdir(dir)

    store datas into namelist


    if aflag == 1

        write file type / permission

        write number of hard links

        write owner name

        write group name

        write file size
```

write file modification time

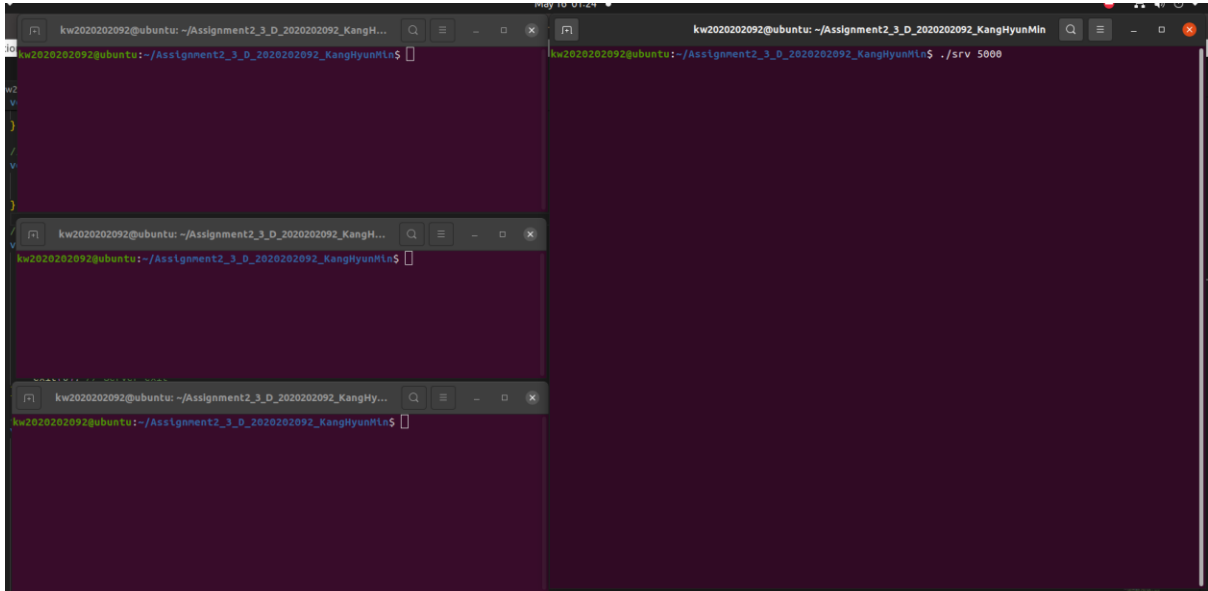
write file or directory name

else write file name

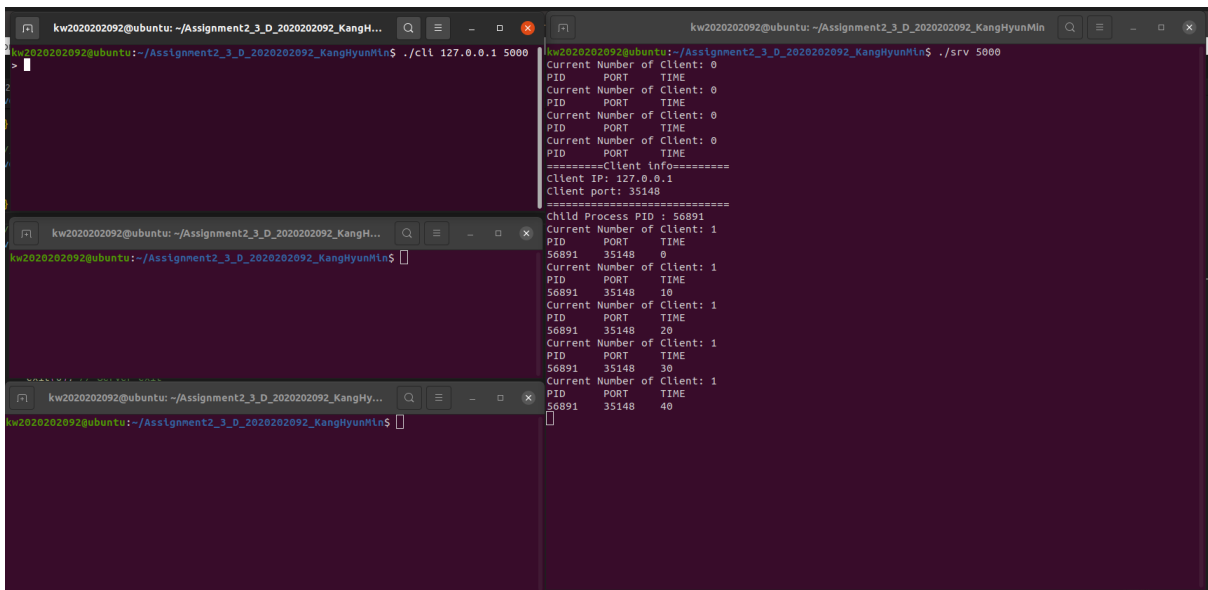


# 결과화면

<클라이언트-서버 연결>



먼저 위와 같이 포트번호를 5000 으로 하여 서버를 실행한다.



그 후 IP 주소 127.0.0.1, 포트번호를 5000 으로하여 클라이언트를 서버에 연결하고, 그 결과 해당 클라이언트의 정보가 서버에 출력되며, 10 초에 한번씩 현재 연결중인

클라이언트 수인 1 과 해당 클라이언트의 정보가 출력되는 것을 확인할 수 있다.

The screenshot shows a terminal window with the following output:

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
>
=====
Child Process PID : 56891
Current Number of Client: 1
PID    PORT    TIME
56891  35148  0
Current Number of Client: 1
PID    PORT    TIME
56891  35148  10
Current Number of Client: 1
PID    PORT    TIME
56891  35148  20
Current Number of Client: 1
PID    PORT    TIME
56891  35148  30
Current Number of Client: 1
PID    PORT    TIME
56891  35148  40
Current Number of Client: 1
PID    PORT    TIME
56891  35148  50
Current Number of Client: 1
PID    PORT    TIME
56891  35148  60
Current Number of Client: 1
PID    PORT    TIME
56891  35148  70
Current Number of Client: 1
PID    PORT    TIME
56891  35148  80
=====Client Info=====
Client IP: 127.0.0.1
Client port: 54586
=====
Child Process PID : 56894
Current Number of Client: 2
PID    PORT    TIME
56891  35148  80
56894  54586  0
Current Number of Client: 2
PID    PORT    TIME
56891  35148  90
56894  54586  10
>
```

그 후 두번째 클라이언트를 연결했을 때, 해당 클라이언트의 정보가 서버에 출력되고 10 초 후 현재 연결 중인 클라이언트들의 정보가 서버에 출력된다.

The screenshot shows a terminal window with the following output:

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
>
PID    PORT    TIME
56891  35148  98
56894  54586  10
Current Number of Client: 2
PID    PORT    TIME
56891  35148  108
56894  54586  20
Current Number of Client: 2
PID    PORT    TIME
56891  35148  118
56894  54586  30
Current Number of Client: 2
PID    PORT    TIME
56891  35148  128
56894  54586  40
Current Number of Client: 2
PID    PORT    TIME
56891  35148  138
56894  54586  50
Current Number of Client: 2
PID    PORT    TIME
56891  35148  148
56894  54586  60
Current Number of Client: 2
PID    PORT    TIME
56891  35148  158
56894  54586  70
=====Client Info=====
Client IP: 127.0.0.1
Client port: 60658
=====
Child Process PID : 56900
Current Number of Client: 3
PID    PORT    TIME
56891  35148  161
56894  54586  73
56900  60658  0
Current Number of Client: 3
PID    PORT    TIME
56891  35148  171
56894  54586  83
56900  60658  10
>
```

그리고 세번째 클라이언트까지 연결했을 때, 위와 동일한 결과가 서버에 출력되고, 최종적으로 3 개의 클라이언트가 연결되어 10 초에 한번씩 연결된 클라이언트 수 3 과 각

```

kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
> quit
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin$

kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
>

kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
>

kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
>

while (token != NULL && argc < sizeof(argv) / sizeof(argv[0])) {

```

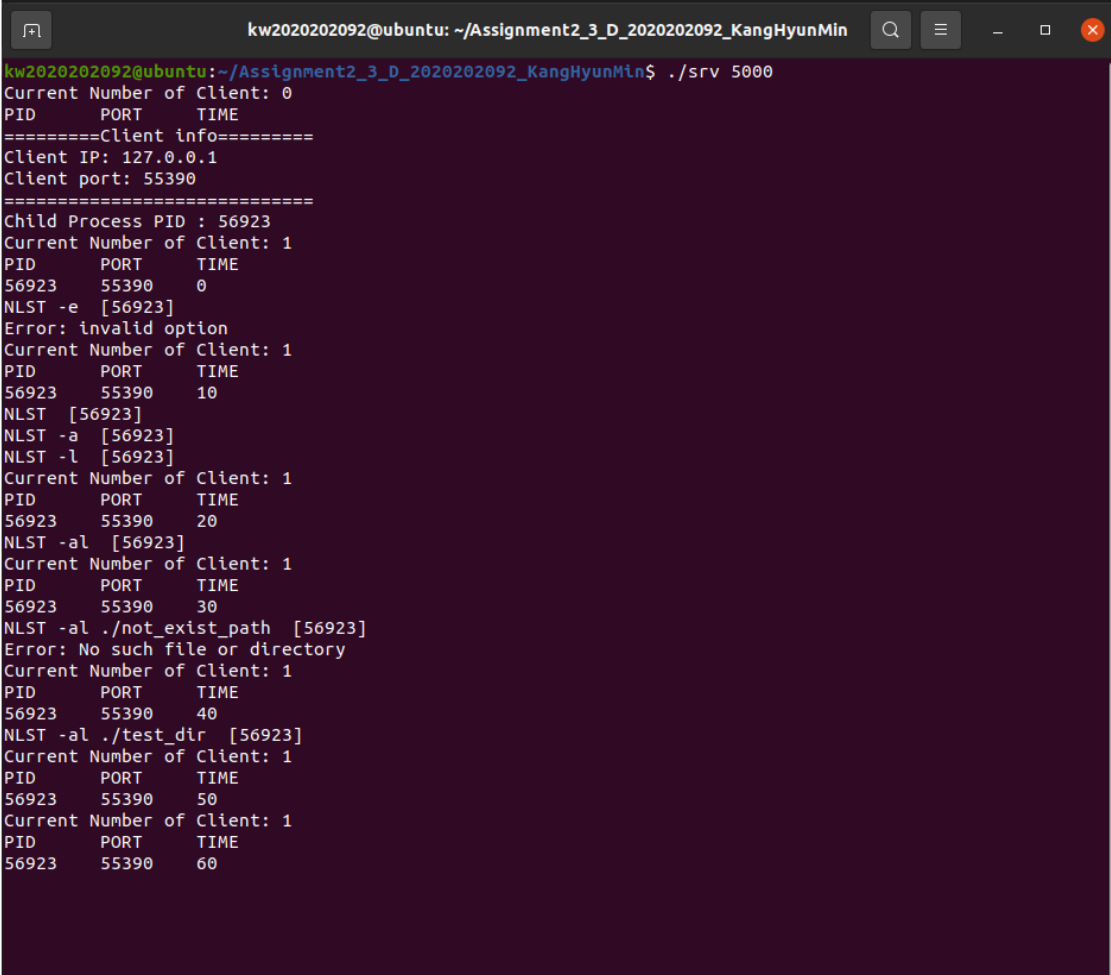
The image shows two terminal windows side-by-side, both running on a Kali Linux system. The left window is titled 'kw2020202092@ubuntu: ~/Assignment2\_3\_D\_2020202092\_KangHyunMin\$' and shows the client's perspective. The right window is titled 'kw2020202092@ubuntu: ~/Assignment2\_3\_D\_2020202092\_KangHyunMin\$' and shows the server's perspective. Both windows are running a C++ program that uses a shared library 'libnet.so' to simulate network connections. The client sends a request to the server, and the server responds with a list of client information, including PID, PORT, and TIME. The client then releases the client information, and the server responds with a list of client information, including PID, PORT, and TIME. The client then releases the client information, and the server responds with a list of client information, including PID, PORT, and TIME.

그리고 나서 두번째로 연결한 클라이언트인 pid: 56894 와 마지막으로 연결한 클라이언트인 pid: 56900 을 CTRL+C 로 종료한 경우 위와 동일하게 각 종료에 대해 종료 메시지와 해당 PID 가 출력되며, 현재 연결중인 클라이언트 정보에서 제거되어 더 이상 해당 프로세스의 정보가 출력되지 않는 것을 확인할 수 있다.



<각 명령어의 동작 화면>

## 1. NLST



```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin$ ./srv 5000
Current Number of Client: 0
PID      PORT    TIME
=====Client info=====
Client IP: 127.0.0.1
Client port: 55390
=====
Child Process PID : 56923
Current Number of Client: 1
PID      PORT    TIME
56923    55390    0
NLST -e [56923]
Error: invalid option
Current Number of Client: 1
PID      PORT    TIME
56923    55390    10
NLST [56923]
NLST -a [56923]
NLST -l [56923]
Current Number of Client: 1
PID      PORT    TIME
56923    55390    20
NLST -al [56923]
Current Number of Client: 1
PID      PORT    TIME
56923    55390    30
NLST -al ./not_exist_path [56923]
Error: No such file or directory
Current Number of Client: 1
PID      PORT    TIME
56923    55390    40
NLST -al ./test_dir [56923]
Current Number of Client: 1
PID      PORT    TIME
56923    55390    50
Current Number of Client: 1
PID      PORT    TIME
56923    55390    60
```

(Server)

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
> ls -e
Error: invalid option
> ls
Makefile      b      c      cli      cli.c
srv      srv.c  test_dir/
> ls -a
./      ../      Makefile      b      c
cli      cli.c  srv      srv.c  test_dir/

> ls -l
-rw-rw-r-- 1 kw2020202092 kw2020202092 54 May 06 21:20 Makefile
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 10 03:33 b
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 10 03:33 c
-rwxrwxr-x 1 kw2020202092 kw2020202092 17400 May 16 01:23 cli
-rwxrw-rw- 1 kw2020202092 kw2020202092 6974 May 15 23:34 cli.c
-rwxrwxr-x 1 kw2020202092 kw2020202092 36392 May 16 01:23 srv
-rwxrw-rw- 1 kw2020202092 kw2020202092 41481 May 16 01:17 srv.c
drwxrwxr-x 2 kw2020202092 kw2020202092 4096 May 16 01:22 test_dir/

> ls -al
drwxrwxr-x 3 kw2020202092 kw2020202092 4096 May 16 01:23 ./
drwxr-xr-x 28 kw2020202092 kw2020202092 4096 May 16 01:37 ../
-rw-rw-r-- 1 kw2020202092 kw2020202092 54 May 06 21:20 Makefile
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 10 03:33 b
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 10 03:33 c
-rwxrwxr-x 1 kw2020202092 kw2020202092 17400 May 16 01:23 cli
-rwxrw-rw- 1 kw2020202092 kw2020202092 6974 May 15 23:34 cli.c
-rwxrwxr-x 1 kw2020202092 kw2020202092 36392 May 16 01:23 srv
-rwxrw-rw- 1 kw2020202092 kw2020202092 41481 May 16 01:17 srv.c
drwxrwxr-x 2 kw2020202092 kw2020202092 4096 May 16 01:22 test_dir/

> ls -al ./not_exist_path
Error: No such file or directory
> ls -al ./test_dir
drwxrwxr-x 2 kw2020202092 kw2020202092 4096 May 16 01:22 ./
drwxrwxr-x 3 kw2020202092 kw2020202092 4096 May 16 01:23 ../
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 16 01:22 fileA
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 16 01:22 fileB

> █
```

(Client)

위와 같이 먼저 ls -e 를 입력한 경우 정의되지 않은 옵션이므로 에러메세지가 출력되는데, 이 때 서버와 클라이언트 둘 다 에러메세지가 출력되는 것을 확인할 수 있다. 또한 ls, ls -a, ls -al 에 대한 결과가 클라이언트에 출력되며, ls -al ./not\_exist\_path 를 입력한 경우 존재하지 않는 경로이기 때문에 이에 해당하는 에러메세지가 서버와 클라이언트에 출력되는 것을 확인할 수 있다. 또한 ls -al ./test\_dir 을 입력한 경우 test\_dir 디렉토리 내의 리스트가 클라이언트에 정상적으로 출력되는 것을 확인할 수 있다.

## 2. LIST

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./srv 5000
=====Client info=====
Client IP: 127.0.0.1
Client port: 41930
=====
Child Process PID : 56986
Current Number of Client: 1
PID      PORT    TIME
56986    41930    0
LIST [56986]
LIST -e [56986]
Error: invalid option
Current Number of Client: 1
PID      PORT    TIME
56986    41930    10
LIST ./not_exist_path [56986]
Error: No such file or directory
Current Number of Client: 1
PID      PORT    TIME
56986    41930    20
LIST ./test_dir [56986]
Current Number of Client: 1
PID      PORT    TIME
56986    41930    30
Current Number of Client: 1
PID      PORT    TIME
56986    41930    40
□
```

(Server)

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
> dir
drwxrwxr-x 3 kw2020202092 kw2020202092 4096 May 16 01:23 ./
drwxr-xr-x 28 kw2020202092 kw2020202092 4096 May 16 01:37 ../
-rw-rw-r-- 1 kw2020202092 kw2020202092 54 May 06 21:20 Makefile
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 10 03:33 b
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 10 03:33 c
-rwxrwxr-x 1 kw2020202092 kw2020202092 17400 May 16 01:23 cli
-rwxrw-rw- 1 kw2020202092 kw2020202092 6974 May 15 23:34 cli.c
-rwxrwxr-x 1 kw2020202092 kw2020202092 36392 May 16 01:23 srv
-rwxrw-rw- 1 kw2020202092 kw2020202092 41481 May 16 01:17 srv.c
drwxrwxr-x 2 kw2020202092 kw2020202092 4096 May 16 01:22 test_dir/

> dir -e
Error: invalid option
> dir ./not_exist_path
Error: No such file or directory
> dir ./test_dir
drwxrwxr-x 3 kw2020202092 kw2020202092 4096 May 16 01:22 ./
drwxrwxr-x 3 kw2020202092 kw2020202092 4096 May 16 01:23 ../
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 16 01:22 fileA
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 16 01:22 fileB

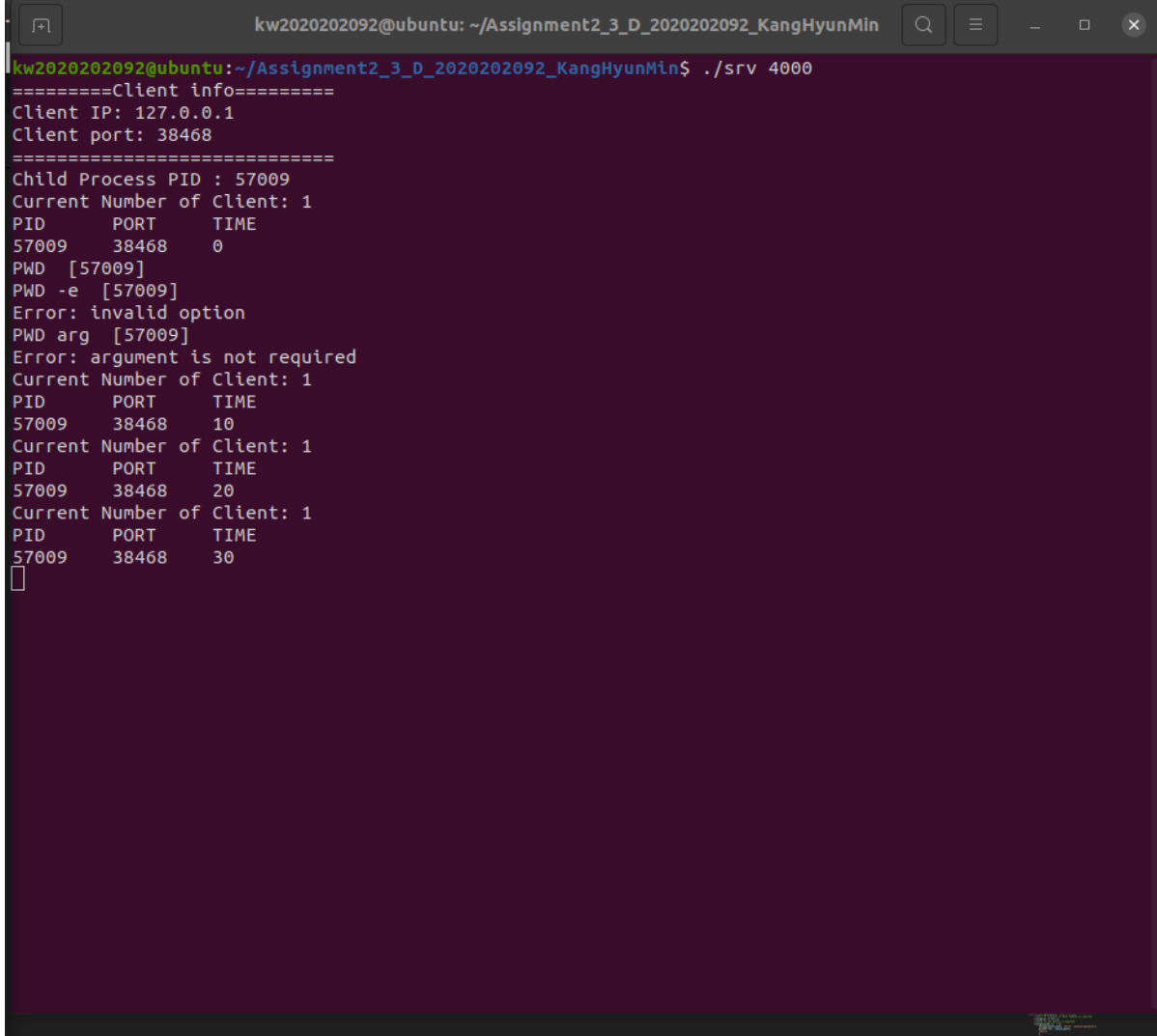
> █
```

(Client)

먼저 서버와 클라이언트를 연결한 후 dir 을 입력했을 때 해당 결과가 클라이언트에 출력된다. 그리고 dir -e 를 입력한 경우 정의되지 않은 옵션이므로 에러메세지가 서버와 클라이언트에 출력되는 것을 확인할 수 있고, dir ./not\_exist\_dir 에 대해서도 동일하게 서버와 클라이언트에 에러메세지가 출력된다. 그리고 dir test\_dir 을 입력한 경우 test\_dir 디렉토리 내의 결과가 클라이언트에 출력되는 것을 확인할 수 있다.



### 3. PWD



```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./srv 4000
=====Client info=====
Client IP: 127.0.0.1
Client port: 38468
=====
Child Process PID : 57009
Current Number of Client: 1
PID      PORT      TIME
57009    38468     0
PWD [57009]
PWD -e [57009]
Error: invalid option
PWD arg [57009]
Error: argument is not required
Current Number of Client: 1
PID      PORT      TIME
57009    38468     10
Current Number of Client: 1
PID      PORT      TIME
57009    38468     20
Current Number of Client: 1
PID      PORT      TIME
57009    38468     30
█
```

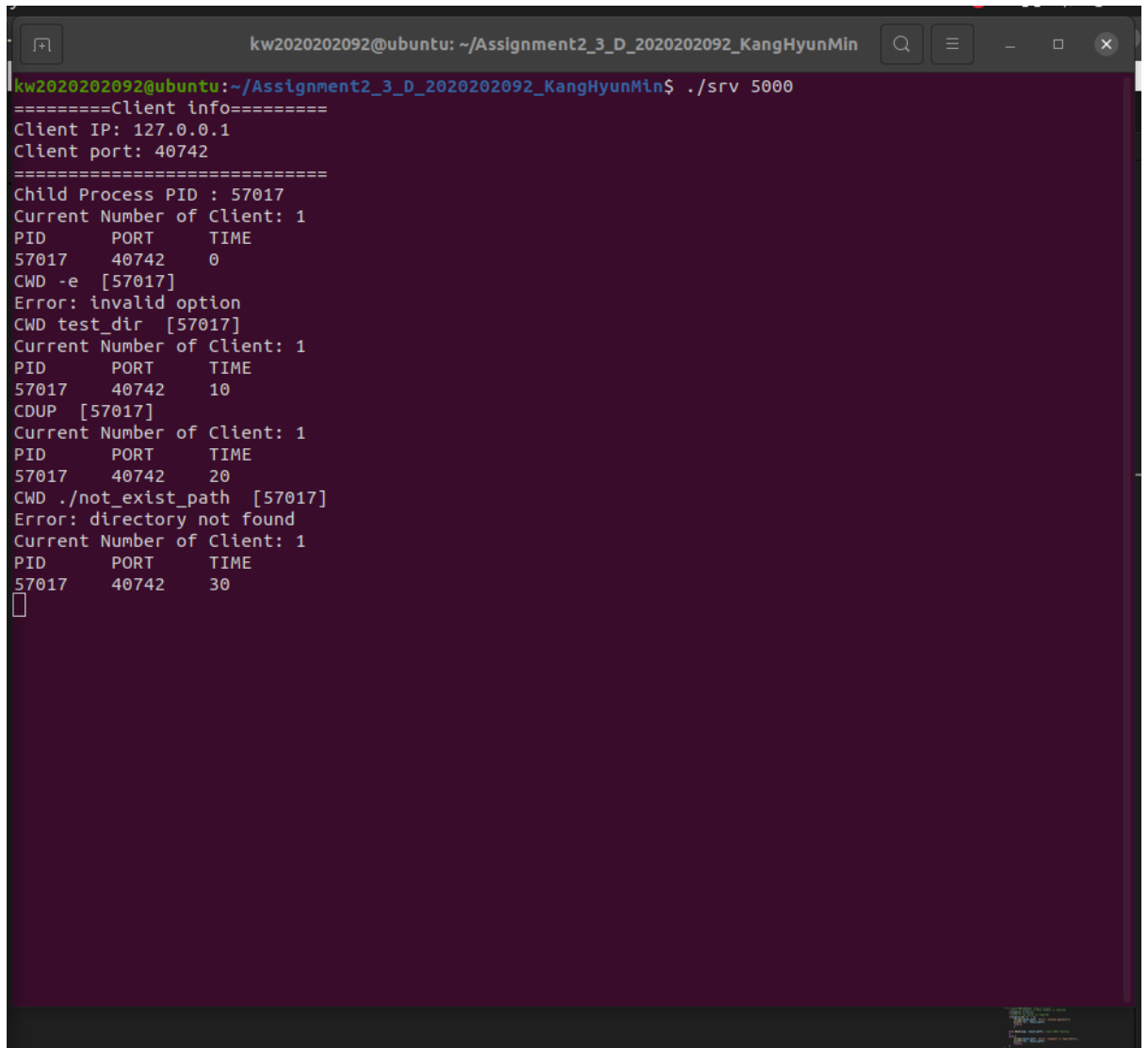
(Server)

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 4000
> pwd
"/home/kw2020202092/Assignment2_3_D_2020202092_KangHyunMin" is current directory
> pwd -e
Error: invalid option
> pwd arg
Error: argument is not required
> 
```

(Client)

서버와 클라이언트 연결 후 `pwd` 를 입력했을 때, 현재 디렉토리의 경로가 클라이언트에 출력되는 것을 확인할 수 있다. 또한, `pwd -e`, `pwd arg` 와 같이 정의되지 않은 옵션을 입력하거나 인자를 입력하는 경우 각 경우에 해당하는 에러메세지가 서버와 클라이언트에 출력되는 것을 확인할 수 있다.

#### 4. CWD & CDUP



A terminal window titled 'kw2020202092@ubuntu: ~/Assignment2\_3\_D\_2020202092\_KangHyunMin' showing the output of a server program. The program is run with './srv 5000'. The output shows client information (IP: 127.0.0.1, port: 40742) and a table of client activity. Three red arrows point to specific lines in the log: 'CWD -e [57017]', 'CWD test\_dir [57017]', and 'CWD ./not\_exist\_path [57017]'. The log shows that the first two commands succeed, while the third fails with 'Error: directory not found'.

```
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./srv 5000
=====Client info=====
Client IP: 127.0.0.1
Client port: 40742
=====
Child Process PID : 57017
Current Number of Client: 1
PID      PORT      TIME
57017    40742     0
CWD -e [57017]
Error: invalid option
CWD test_dir [57017]
Current Number of Client: 1
PID      PORT      TIME
57017    40742     10
CDUP [57017]
Current Number of Client: 1
PID      PORT      TIME
57017    40742     20
CWD ./not_exist_path [57017]
Error: directory not found
Current Number of Client: 1
PID      PORT      TIME
57017    40742     30
□
```

(Server)

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
> cd -e ..
Error: invalid option
> cd test_dir
CWD test_dir
"/home/kw2020202092/Assignment2_3_D_2020202092_KangHyunMin/test_dir" is current directory
> cd ..
CDUP
"/home/kw2020202092/Assignment2_3_D_2020202092_KangHyunMin" is current directory
> cd ./not_exist_path
Error: directory not found
> 
```

(Client)

서버와 클라이언트 연결 후 `cd -e ..`을 입력했을 때, 정의되지 않은 옵션을 사용했으므로 에러 메시지가 서버와 클라이언트에 출력되는 것을 확인할 수 있다. 그리고 `cd test_dir`을 입력한 경우 해당 디렉토리로 현재 디렉토리를 변경하여 결과가 클라이언트에 출력되며, `cd ..`을 입력한 경우 이전 디렉토리로 현재 디렉토리를 변경하여 결과가 클라이언트에 출력되는 것을 확인할 수 있다. 마지막으로 `cd./not_exist_path`를 입력한 경우 존재하지 않는 디렉토리이므로 서버와 클라이언트에 에러 메시지가 출력되는 것을 확인할 수 있다.

## 5. MKD

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./srv 4000
Current Number of Client: 0
PID      PORT    TIME
=====Client info=====
Client IP: 127.0.0.1
Client port: 59680
=====
Child Process PID : 61013
Current Number of Client: 1
PID      PORT    TIME
61013    59680    0
MKD -e new_dir [61013]
Error: invalid option
MKD new_dir1 [61013]
Current Number of Client: 1
PID      PORT    TIME
61013    59680    10
MKD new_dir2 new_dir3 [61013]
Current Number of Client: 1
PID      PORT    TIME
61013    59680    20
MKD new_dir1 [61013]
Error: cannot create directory 'new_dir1': File exists
MKD [61013]
Error: argument is required
Current Number of Client: 1
PID      PORT    TIME
61013    59680    30
█
```

(Server)

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 4000
> mkdir -e new_dir
Error: invalid option
> mkdir new_dir1
MKD new_dir1
> mkdir new_dir2 new_dir3
MKD new_dir2
MKD new_dir3
> mkdir new_dir1
Error: cannot create directory 'new_dir1': File exists
> mkdir
Error: argument is required
> █
```

(Client)

서버와 클라이언트 연결 후 `mkdir -e new_dir` 을 입력한 경우 정의되지 않은 옵션이 사용되었으므로 에러메세지가 서버와 클라이언트에 출력된다. 그리고 `mkdir new_dir1` 을 입력한 경우 정상적으로 디렉토리가 생성되고, 성공 메시지가 출력되는 것을 확인할 수 있다. `Mkdir new_dir2 new_dir3` 과 같이 인자를 여러 개 입력하는 경우에도 해당 이름의 디렉토리를 모두 생성한 후 성공 메시지가 출력되는 것을 확인할 수 있고, `mkdir new_dir1` 을 입력하는 경우 이미 존재하는 파일명이기 때문에 에러메세지가 서버와 클라이언트에 출력된다. 마지막으로 인자 없이 `mkdir` 을 입력하는 경우 또한 에러메세지가 서버와 클라이언트에 출력되는 것을 확인할 수 있다.

## 6. DELE

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./srv 5000
=====Client info=====
Client IP: 127.0.0.1
Client port: 43242
=====
Child Process PID : 61046
Current Number of Client: 1
PID    PORT    TIME
61046  43242    0
Current Number of Client: 1
PID    PORT    TIME
61046  43242    10
DELE test1.dat [61046]
Current Number of Client: 1
PID    PORT    TIME
61046  43242    20
NLST [61046]
Current Number of Client: 1
PID    PORT    TIME
61046  43242    30
Current Number of Client: 1
PID    PORT    TIME
61046  43242    40
NLST test1.dat [61046]
Error: No such file or directory
DELE test2.dat test3.dat [61046]
Current Number of Client: 1
PID    PORT    TIME
61046  43242    50
NLST test2.dat [61046]
Error: No such file or directory
NLST test3.dat [61046]
Error: No such file or directory
Current Number of Client: 1
PID    PORT    TIME
61046  43242    60
DELE -e test.dat [61046]
Error: invalid option
Current Number of Client: 1
PID    PORT    TIME
61046  43242    70
DELE not_exist_file [61046]
Error: failed to delete 'not_exist_file'
```

(Server)

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ touch test1.dat
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ touch test2.dat
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ touch test3.dat
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ls -al *.dat
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 16 03:29 test1.dat
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 16 03:29 test2.dat
-rw-rw-r-- 1 kw2020202092 kw2020202092 0 May 16 03:29 test3.dat
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
> delete test1.dat
DELE test1.dat
> ls
Makefile      b      c      cli      cli.c
new_dir1/     new_dir2/  new_dir3/  srv      srv.c
test2.dat     test3.dat  test_dir/
> ls test1.dat
Error: No such file or directory
> delete test2.dat test3.dat
DELE test2.dat
DELE test3.dat
> ls test2.dat
Error: No such file or directory
> ls test3.dat
Error: No such file or directory
> delete -e test.dat
Error: invalid option
> delete not_exist_file
Error: failed to delete 'not_exist_file'
> 
```

(Client)

위와 같이 먼저 test1.dat test2.dat test3.dat 파일을 생성한 후 서버와 클라이언트를 연결했을 때 delete test1.dat 을 입력하고 ls 를 입력한 결과, test1.dat 파일이 삭제된 것을 확인할 수 있다. 그 후에 delete test2.dat test3.dat 과 같이 여러 개의 인자를 입력한 경우에도 해당 파일들이 모두 삭제되는 것을 확인할 수 있고, delete -e 그리고 delete not\_exist\_file 을 입력한 경우 각각 정의되지 않은 옵션을 사용했고, 존재하지 않는 파일을 인자로 입력했기 때문에 에러메세지가 클라이언트와 서버에 출력되는 것을 확인할 수 있다.



## 7. RMD

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./srv 5000
=====Client info=====
Client IP: 127.0.0.1
Client port: 60080
=====
Child Process PID : 62225
Current Number of Client: 1
PID      PORT    TIME
62225    60080    0
MKD dir1 [62225]
MKD dir2 dir3 [62225]
RMD dir1 [62225]
Current Number of Client: 1
PID      PORT    TIME
62225    60080    10
NLST [62225]
RMD dir2 dir3 [62225]
NLST [62225]
MKD dir1 [62225]
Current Number of Client: 1
PID      PORT    TIME
62225    60080    20
RMD not_exist_dir dir1 [62225]
Error: failed to remove 'not_exist_dir'
RMD [62225]
Error: argument is required
Current Number of Client: 1
PID      PORT    TIME
62225    60080    30
█
```

(Server)

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
> mkdir dir1
MKD dir1
> mkdir dir2 dir3
MKD dir2
MKD dir3
> rmdir dir1
RMD dir1
> ls
Makefile      b      c      cli      cli.c
dir2/  dir3/  new_dir2/  new_dir3/  srv
srv.c  test_dir/
> rmdir dir2 dir3
RMD dir2
RMD dir3
> ls
Makefile      b      c      cli      cli.c
new_dir2/     new_dir3/  srv      srv.c  test_dir/

> mkdir dir1
MKD dir1
> rmdir not_exist_dir dir1
Error: failed to remove 'not_exist_dir'
RMD dir1
> rmdir
Error: argument is required
> 
```

(Client)

서버와 클라이언트를 연결한 후 `mkdir dir1 dir2 dir3` 을 통해 세개의 디렉토리를 생성하고 `rmdir dir1` 을 입력했을 때 `ls` 를 입력한 결과 정상적으로 제거된 것을 확인할 수 있다. 그리고 `rmdir dir2 dir3` 과 같이 여러 개의 인자를 입력한 경우에도 모두 제거되는 것을 확인할 수 있다. 그리고나서 `mkdir dir1` 을 통해 디렉토리를 생성한 후 `rmdir not_exist_dir dir1` 과 같이 존재하지 않는 디렉토리와 존재하는 디렉토리를 인자로 입력한 경우 존재하지 않는 디렉토리에 대해서는 에러 메시지가 서버와 클라이언트에 출력되고, 존재하는 디렉토리는 제거되는 것을 확인할 수 있다. 마지막으로 인자 없이 `rmdir` 을 입력한 경우 에러메세지가 서버와 클라이언트에 출력된다.

## 8. RNFR & RENTO

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./srv 5000
=====Client info=====
Client IP: 127.0.0.1
Client port: 41186
=====
Child Process PID : 63831
Current Number of Client: 1
PID      PORT    TIME
63831    41186    0
NLST [63831]
RNFR a RNT0 b [63831]
Error: name to change already exists
RNFR a RNT0 c [63831]
Current Number of Client: 1
PID      PORT    TIME
63831    41186    10
NLST [63831]
Current Number of Client: 1
PID      PORT    TIME
63831    41186    20
RNFR RNT0 [63831]
Error: two arguments are required
RNFR a RNT0 [63831]
Error: two arguments are required
Current Number of Client: 1
PID      PORT    TIME
63831    41186    30
█
```

(Server)

```
kw2020202092@ubuntu: ~/Assignment2_3_D_2020202092_KangHyunMin
kw2020202092@ubuntu:~/Assignment2_3_D_2020202092_KangHyunMin$ ./cli 127.0.0.1 5000
> ls
Makefile      a      b      cli      cli.c
new_dir2/     new_dir3/  srv    srv.c    test_dir/

> rename a b
Error: name to change already exists
> rename a c
RNFR a
RNT0 c
> ls
Makefile      b      c      cli      cli.c
new_dir2/     new_dir3/  srv    srv.c    test_dir/

> rename
Error: two arguments are required
> rename a
Error: two arguments are required
> 
```

(Client)

먼저 ls 를 통해 a b c 파일이 존재하는 것을 확인한 후 rename a b 를 입력한 결과 이미 b 파일이 존재하기 때문에 rename 할 수 없으므로 에러 메시지가 서버와 클라이언트에 출력된다. 그리고 rename a c 를 입력하는 경우 c 파일이 존재하지 않기 때문에 rename 이 가능하므로 동작을 수행하고, ls 를 입력한 결과 의도대로 동작했음을 확인할 수 있다. 그리고 rename 의 인자가 없거나 한 개인 경우에는 서버와 클라이언트에 에러메세지가 출력되는 것을 확인할 수 있다.

## 9. QUIT

QUIT 명령어의 동작은 앞의 클라이언트-서버 연결 부분에서 설명했으므로 생략한다.

## 고찰

이번 [FTP 2-3](#) 과제를 통해 지금까지의 서버-클라이언트 구조 구현, 각 명령어의 동작 구현, 소켓 프로그래밍 등의 모든 부분을 합쳐 최종적으로 클라이언트에서 사용자 명령어를 입력하면, FTP 명령어로 변환되어 서버로 전송되고, 서버에서 처리되어 결과가 클라이언트로 전송되는 과정을 구현했다. 또한, 이번에 새롭게 학습한 SIGINT 시그널에 대해 익힐 수 있었고, 이를 응용할 수 있는 기회가 되었다. 또한, 새로운 클라이언트가 접속할 때마다 새로 fork 를 통해 프로세스를 생성하여 서버-클라이언트 구조를 이루는 concurrent server 에 대해 학습하고 이를 응용하여 다중접속이 가능한 서버-클라이언트를 코드로 구현할 수 있는 좋은 기회가 되었다.

## Reference