시스템 프로그래밍 실습

[Assignment1-3]

Class : **[A]**

Professor : **[김태석 교수님]**

Student ID : **[2019202032]**

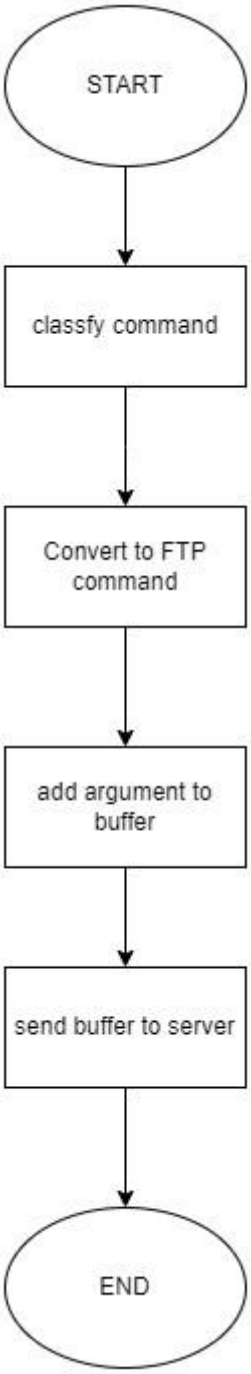
Name : **[이상현]**

Introduction

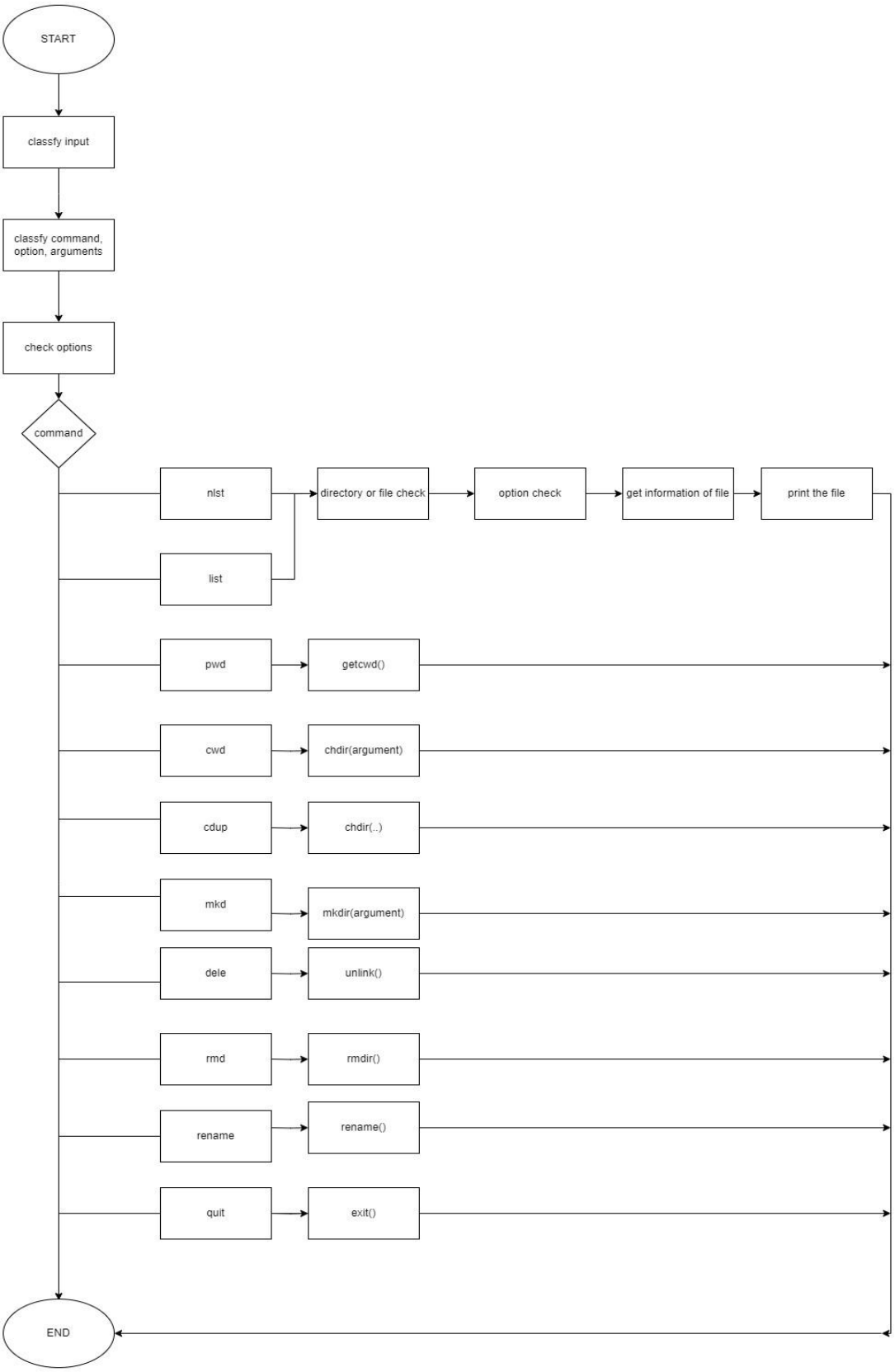
해당 과제는 cli.c 파일과 srv.c 파일의 코드를 구현하는 과제이다. cli.c 파일은 사용자로부터 명령어, 옵션, 인자 등을 입력 받고, 입력 받은 명령어이를 FTP 명령어로 변환한다. 변환된 결과를 결과를 pipe 와 tee command 를 이용하여 srv.c 의 입력으로 전달한다. 이때 입력과 출력에 대해서는 read()함수와 write()함수를 사용한 standard input & output 으로 진행하도록 한다. 각각의 명령어에 대해서 invalid 한 옵션이 입력되거나 잘못된 파일이름이나 경로가 주어지면 예외처리를 한다.

Flow chart

[CLI]



[SRV]


START
classfy input
classfy command, option, arguments
check options
command
nlst
directory or file check
option check
get information of file
print the file
list
pwd
getcwd()
cwd
chdir(argument)
cdup
chdir(..)
mkd
mkdir(argument)
dele
unlink()
rmd
rmdir()
rename
rename()
quit
exit()
END

Input 에 대해서 command 와 option, arguments 를 구분하고, 각각의 command 에 맞는 동작을 하도록 적절한 함수를 사용한다.

Pseudo code

[cli.c]

```
#define MAX_SIZE 4096

int main(int argc, char **argv) {
	char buffer[MAX_SIZE];
	int cmd_num = 0;
	////////////////	command check start	//////////////
	if(command == ls)	// case of ls
	{
		strcpy(buffer, "NLST");
	}
	else if(command == dir))	//case of dir
		strcpy(buffer, "LIST");
	else if(command == pwd)	//case of pwd
		strcpy(buffer, "PWD");
	else if(command == cd)	//case of cd
	{	if(argc >= 3 && !(strcmp(argv[2], "..")))	//convert to cdup
			strcpy(buffer, "CDUP");
		else strcpy(buffer, "CWD");			//conver to cwd
	}
	else if(command == mkdir)	//case of mkdir
		strcpy(buffer, "MKD");
	else if(!strcmp("delete", argv[1])) //case of delete
		strcpy(buffer, "DELE");
```

```
        else if(command == rmdir)   //case of rmdir
                strcpy(buffer, "RMD");
        else if(command == rename){      //case of rename
                strcpy(buffer, "RNFR");
        }
        else if(command == quit)    //case of quit
                strcpy(buffer, "QUIT");
        else
                return 0;
        /////////////////    end of checking command ////////////////


        ////////////////      add argument to buffer   ////////////////
        While(No argument){
            Copy argument to buffer }
        ////////////////      end of adding argument   ////////////////
            Send buffer;
}
```

[srv]

- error_handling

```
void error_handling(int num){
	//////////////	start checking error type	//////////////
	if( invalid option
	{
		write(1, "Error : invalid option₩n₩0", 25);
	}
	else if invalid argument
	{
		write(1, "Error : argument is not required₩n₩0", 35);
	}
	else if no argument
		write(1, "Error : argument is required₩n₩0", 31);
	}
	else if lack argument
		write(1, "Error : two arguments are required₩n₩0", 37);
	}
	else if much too many arguments
		write(1, "Error : only one argument can be processed₩n₩0", 45);
	}
	/////////////// end checking argument ////////////////////
	exit(0);
}
```

- option check

```
void option_check(char *options, int *aflag, int *lflag, int *invalid){
    ////////////// check option start /////////////////
    for(int i = 1; i<strlen(options); i++){
        if(options[i] == 'a')
            *aflag = 1;                    //a option
        else if(options[i] == 'l')
            *lflag = 1;                    //l option
        else
            *invalid = 1;                  //invalid option
    }
    ////////////// end of option check ///////////////
}
```

- getfiletype

```
char GetFiletype(struct stat file, char *filename)
{
	////////////////// get file type ////////////////////////
	stat(filename, &file);
	if (S_ISBLK(file.st_mode))   //type of b
		return 'b';
	else if (S_ISCHR(file.st_mode)) //type of c
		return 'c';
	else if (S_ISDIR(file.st_mode)) //type of directory
		return 'd';
	else if (S_ISFIFO(file.st_mode))      //type of FiFO
		return 'p';
	else if (S_ISLNK(file.st_mode)) //type of link
		return 'l';
	else if (S_ISREG(file.st_mode)) //type of file
		return '-';
	else if (S_ISSOCK(file.st_mode))      //type of l
		return 'l';
	return 'N'; // not exist file
}
```

-option_l

```
char option_l(char *argument){
	char*buf = (char*)malloc(sizeof(char) * MAX_SIZE);
	struct stat file;

	//////////////filetype//////////////////////
	char filetype = GetFiletype(file, argument);	//get filetype
	write(1, temp, strlen(temp));	//print filetype

	//////// permission //////////////////
	GetPermission(file, argument, permission);	//get permission
	write(1, permission, strlen(permission));	//write permission

	//////// nlink //////////////////////
	int filelink = file.st_nlink;		//get nlink
	write(1, temp, strlen(temp));		//write nlink

	//////////////pw_name/////////////////////////
	owner_id = file.st_uid;		//get owner of file
	write(1, temp, strlen(temp));	//write owner

	/////////////// group id ////////////////////////
	struct group *grp = getgrgid(getgid()); //get group id
	write(1, grp->gr_name, strlen(grp->gr_name));	//write group id
```

```
/////////////// filesize//////////////////////////

long long filesize = file.st_size;   //get file size

write(1, temp, strlen(temp));     //write filesize


////////////////// file time ////////////////////////

time_t t = file.st_mtime;            //get file time


char *t_ptr = strtok(asctime(time), " ");

while(t_ptr != NULL){

        write(1, t_ptr, strlen(t_ptr)); //write month

    }

    else if(td_num == 2){

        write(1, t_ptr, strlen(t_ptr)); //write day

    }

    if(td_num == 3){

        for(int i = 0; i<5; i++)

            time_data[td_num][i] = t_ptr[i];

        write(1, time_data[td_num], strlen(time_data[td_num])); //write time

    }

    td_num++;

    t_ptr = strtok(NULL, " ");

}


write(1, argument, strlen(argument));     //write file name
```

```
        if(filetype == 'd')
        {
                write(1, "/", 1);      //if it is directory, write /
        }

        write(1, "₩n", 1);
}
```

- main

```
nt main(int argc, char **argv){
        classfy input
    ////////////// start input parsing ////////////////////////
while(ptr){
        get command;
        get argument
    /////////////// end parsing input ////////////////////////

        ////////////////////get option ////////////////////////////
        option_check(argument[0], &aflag, &lflag, &invalid);
    }
    //////////////     end option check ////////////////////////

////////////////////// start processing command ///////////////
    if(!strcmp(command, "NLST") || !strcmp(command, "LIST")){   //case of NLST and LIST
    error_handling;
////////////////// if the argument is file ////////////////////////
         if(access(argument[argument_starting_number], F_OK)== -1 ){
              correct_argument(argument[argument_starting_number],          command);
//check if i can open the directory

        }
```

```
        ////////////////////// case of no read permission     ////////////////////
        struct stat file;
        char per[100];
        GetPermission(file, argument[argument_starting_number], per);
        if(per[0] == '-')
        {
            write(1, "cannot access₩n₩0", 16);
            exit(0);
        }
        char filetype = GetFiletype(file, argument[argument_starting_number]);

        ////////////////// case of file ////////////////////////////
        if(filetype == '-' && arg_cnt != 0)
        {
            option_l(argument[argument_starting_number]);
       }

////////// get the files of directory ////////////////////
        dirp = opendir(argument[argument_starting_number]);
        while((dir = readdir(dirp)) != NULL){
            if(aflag)           //if a option
                strcpy(filenames[filecnt++], dir->d_name);
            else{
                if(dir->d_name[0] != '.')     //if not a option
```

```
                    strcpy(filenames[filecnt++], dir->d_name);
            }
        }
        closedir(dirp);
        ////////////change directory////////////////////
        chdir(argument[argument_starting_number]);   //change directory
        //////// arrange files ////////////////////////////////////
        ArrangeFilenames(filenames, temp_filenames, 0, filecnt-1);

        if(aflag != 0 && lflag != 0){           //case of al option
          option_l(filenames[i]);
            exit(0);
        }
        else if(aflag){
       write(1, filenames[i], strlen(filenames[i]));
exit(0);
        }
        else if(lflag){        //case of l option
                option_l(filenames[i]);
            exit(0);
        }
        else{                       //case of no option

          write(1, filenames[i], strlen(filenames[i]));
```

```
exit(0);
            }


    }
    else if(!strcmp(command, "PWD")){     //print working directory
          error handling;
         print(pwd());
    }
    else if(!strcmp(command, "CWD")){     //change directory
          error handling;
         chdir();
    }
    else if(!strcmp(command, "CDUP")){    //cd ..
          error handling;
         chdir(..);
    }
    else if(!strcmp(command, "MKD")){     //make directory
         error handling;
         int mkdir = mkdir();
    }
    else if(!strcmp(command, "DELE")){    //delete
         error handling;
         int unlink = unlink();
    }
```

```
        else if(!strcmp(command, "RNFR")){    //rename
            int rename = rename()
        }
        else if(!strcmp(command, "QUIT")){    //quit
            error handling
             write(1, "QUIT success₩n₩0", 15);     //quit the program
             exit(0);
        }
        else{
             write(1, "ERROR : NO COMMAND₩n₩0", 21);
        }
}
```

결과화면

- CLI & NLST

```
kw2019202032@ubuntu:~/Assignment1_3$ ./cli ls -e | tee cli.out | ./srv
Error : invalid option
kw2019202032@ubuntu:~/Assignment1_3$ ./cli ls | tee cli.out | ./srv
NLST
a.txt b c cli cli.c
cli.out copy_srv.c srv srv.c srv_strcpy.c
test1 test11 test111 test_dir_1
kw2019202032@ubuntu:~/Assignment1_3$ ./cli ls -a | tee cli.out | ./srv
NLST -a
./ ../ .vscode/ a.txt b
c cli cli.c cli.out copy_srv.c
srv srv.c srv_strcpy.c test1/ test11/
test111/ test_dir_1/
kw2019202032@ubuntu:~/Assignment1_3$ ./cli ls -l | tee cli.out | ./srv
NLST -l
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:52 a.txt
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:12 b
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:12 c
-rwxrwxr-x 1 kw2019202032 kw2019202032 16880 Apr 16 18:10 cli
-rw-rw-r-- 1 kw2019202032 kw2019202032 1229 Apr 17 01:56 cli.c
-rw-rw-r-- 1 kw2019202032 kw2019202032 9 Apr 17 05:40 cli.out
-rw-rw-r-- 1 kw2019202032 kw2019202032 19418 Apr 17 04:23 copy_srv.c
-rwxrwxr-x 1 kw2019202032 kw2019202032 30664 Apr 17 05:39 srv
-rw-rw-r-- 1 kw2019202032 kw2019202032 19866 Apr 17 05:39 srv.c
-rw-rw-r-- 1 kw2019202032 kw2019202032 18496 Apr 17 04:41 srv_strcpy.c
d--------- 2 kw2019202032 kw2019202032 4096 Apr 17 01:54 test1/
drwxrwxr-x 2 kw2019202032 kw2019202032 4096 Apr 17 01:55 test11/
drwxrwxr-x 2 kw2019202032 kw2019202032 4096 Apr 17 01:55 test111/
drwxrwxr-x 2 kw2019202032 kw2019202032 4096 Apr 16 08:35 test_dir_1/
```

```
kw2019202032@ubuntu:~/Assignment1_3$ ./cli ls -al | tee cli.out | ./srv
NLST -al
drwxrwxr-x 7 kw2019202032 kw2019202032 4096 Apr 17 05:39 ./
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 ../
drwxrwxr-x 2 kw2019202032 kw2019202032 4096 Apr 15 03:26 .vscode/
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:52 a.txt
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:12 b
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:12 c
-rwxrwxr-x 1 kw2019202032 kw2019202032 16880 Apr 16 18:10 cli
-rw-rw-r-- 1 kw2019202032 kw2019202032 1229 Apr 17 01:56 cli.c
-rw-rw-r-- 1 kw2019202032 kw2019202032 10 Apr 17 05:40 cli.out
-rw-rw-r-- 1 kw2019202032 kw2019202032 19418 Apr 17 04:23 copy_srv.c
-rwxrwxr-x 1 kw2019202032 kw2019202032 30664 Apr 17 05:39 srv
-rw-rw-r-- 1 kw2019202032 kw2019202032 19866 Apr 17 05:39 srv.c
-rw-rw-r-- 1 kw2019202032 kw2019202032 18496 Apr 17 04:41 srv_strcpy.c
d--------- 2 kw2019202032 kw2019202032 4096 Apr 17 01:54 test1/
drwxrwxr-x 2 kw2019202032 kw2019202032 4096 Apr 17 01:55 test11/
drwxrwxr-x 2 kw2019202032 kw2019202032 4096 Apr 17 01:55 test111/
drwxrwxr-x 2 kw2019202032 kw2019202032 4096 Apr 16 08:35 test_dir_1/
kw2019202032@ubuntu:~/Assignment1_3$ ./cli ls -al ./not_exist_path | tee cli.out | ./srv
Error : No such file or directory
kw2019202032@ubuntu:~/Assignment1_3$ ./cli ls -al ./test_1 | tee cli.out | ./srv
Error : No such file or directory
kw2019202032@ubuntu:~/Assignment1_3$ ./cli ls -al ./test1 | tee cli.out | ./srv
cannot access
kw2019202032@ubuntu:~/Assignment1_3$ ./cli ls -al /dev | tee cli.out | ./srv
NLST -al
drwxrwxr-x 7 kw2019202032 kw2019202032 4096 Apr 17 05:39 ./
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 ../
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 autofs/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 block/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 bsg/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 btrfs-control/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 bus/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 cdrom/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 cdrw/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 char/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 console/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 core/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 cpu/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 cpu_dma_latency/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 cuse/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 disk/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 dma_heap/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 dmmidi/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 dri/
drwxr-xr-x 24 kw2019202032 kw2019202032 4096 Apr 15 13:16 dvd/
```

Invalid option 과 존재하지 않는 파일에 대해 오류처리를 하였고, -a 옵션이 존재하는 경우에만 숨긴 파일을 출력할 수 있도록 하였다. 또한 directory 뒤에는 /을 붙여서 출력하였고, -l 옵션의 경우 파일의 정보를 출력하고 있다.

- DIR & LIST

```
kw2019202032@ubuntu:~/Assignment1_3$ ./cli dir | tee cli.out | ./srv
LIST
 -al-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:52 a.txt
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:12 b
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:12 c
-rwxrwxr-x 1 kw2019202032 kw2019202032 16880 Apr 16 18:10 cli
-rw-rw-r-- 1 kw2019202032 kw2019202032 1229 Apr 17 01:56 cli.c
-rw-rw-r-- 1 kw2019202032 kw2019202032 6 Apr 17 05:50 cli.out
-rw-rw-r-- 1 kw2019202032 kw2019202032 19418 Apr 17 04:23 copy_srv.c
-rwxrwxr-x 1 kw2019202032 kw2019202032 30664 Apr 17 05:49 srv
-rw-rw-r-- 1 kw2019202032 kw2019202032 20078 Apr 17 05:49 srv.c
-rw-rw-r-- 1 kw2019202032 kw2019202032 18496 Apr 17 04:41 srv_strcpy.c
d--------- 2 kw2019202032 kw2019202032 4096 Apr 17 01:54 test1/
drwxrwxr-x 2 kw2019202032 kw2019202032 4096 Apr 17 01:55 test11/
drwxrwxr-x 2 kw2019202032 kw2019202032 4096 Apr 17 01:55 test111/
drwxrwxr-x 2 kw2019202032 kw2019202032 4096 Apr 16 08:35 test_dir_1/
kw2019202032@ubuntu:~/Assignment1_3$ ./cli dir -e | tee cli.out | ./srv
Error : invalid option
kw2019202032@ubuntu:~/Assignment1_3$ ./cli dir ./not_exist_path | tee cli.out | ./srv
LIST
Error : No such file or directory
```

```
kw2019202032@ubuntu:~/Assignment1_3$ ./cli dir /dev | tee cli.out | ./srv
LIST
 -alN--------- 0 root kw2019202032 0 Dec 31 16:00 autofs
N--------- 0 root kw2019202032 0 Dec 31 16:00 block
N--------- 0 root kw2019202032 0 Dec 31 16:00 bsg
N--------- 0 root kw2019202032 0 Dec 31 16:00 btrfs-control
N--------- 0 root kw2019202032 0 Dec 31 16:00 bus
N--------- 0 root kw2019202032 0 Dec 31 16:00 cdrom
N--------- 0 root kw2019202032 0 Dec 31 16:00 cdrw
N--------- 0 root kw2019202032 0 Dec 31 16:00 char
N--------- 0 root kw2019202032 0 Dec 31 16:00 console
N--------- 0 root kw2019202032 0 Dec 31 16:00 core
N--------- 0 root kw2019202032 0 Dec 31 16:00 cpu
N--------- 0 root kw2019202032 0 Dec 31 16:00 cpu_dma_latency
N--------- 0 root kw2019202032 0 Dec 31 16:00 cuse
N--------- 0 root kw2019202032 0 Dec 31 16:00 disk
N--------- 0 root kw2019202032 0 Dec 31 16:00 dma_heap
N--------- 0 root kw2019202032 0 Dec 31 16:00 dmmidi
N--------- 0 root kw2019202032 0 Dec 31 16:00 dri
N--------- 0 root kw2019202032 0 Dec 31 16:00 dvd
N--------- 0 root kw2019202032 0 Dec 31 16:00 ecryptfs
N--------- 0 root kw2019202032 0 Dec 31 16:00 fb0
N--------- 0 root kw2019202032 0 Dec 31 16:00 fd
N--------- 0 root kw2019202032 0 Dec 31 16:00 full
N--------- 0 root kw2019202032 0 Dec 31 16:00 fuse
N--------- 0 root kw2019202032 0 Dec 31 16:00 hidraw0
N--------- 0 root kw2019202032 0 Dec 31 16:00 hpet
N--------- 0 root kw2019202032 0 Dec 31 16:00 hugepages
N--------- 0 root kw2019202032 0 Dec 31 16:00 hwrng
N--------- 0 root kw2019202032 0 Dec 31 16:00 initctl
N--------- 0 root kw2019202032 0 Dec 31 16:00 input
N--------- 0 root kw2019202032 0 Dec 31 16:00 kmsg
N--------- 0 root kw2019202032 0 Dec 31 16:00 log
N--------- 0 root kw2019202032 0 Dec 31 16:00 loop-control
N--------- 0 root kw2019202032 0 Dec 31 16:00 loop0
N--------- 0 root kw2019202032 0 Dec 31 16:00 loop1
N--------- 0 root kw2019202032 0 Dec 31 16:00 loop10
N--------- 0 root kw2019202032 0 Dec 31 16:00 loop11
N--------- 0 root kw2019202032 0 Dec 31 16:00 loop2
N--------- 0 root kw2019202032 0 Dec 31 16:00 loop3
```

Dir 명령어의 경우 ls-al 과 같은 동작을 하고, 예시화면과 같은 결과를 출력하였다.

- pwd & PWD

```
kw2019202032@ubuntu:~/Assignment1_3$ ./cli pwd | tee cli.out | ./srv
"/home/kw2019202032/Assignment1_3" is current directory
kw2019202032@ubuntu:~/Assignment1_3$ ./cli pwd -e | tee cli.out | ./srv
Error : invalid option
kw2019202032@ubuntu:~/Assignment1_3$ ./cli pwd arg | tee cli.out | ./srv
Error : argument is not required
```

Working directory 를 출력하고 있고 에러를 처리하였다.

- CWD & CDUP

```
kw2019202032@ubuntu:~/Assignment1_3$ ./cli cd -e .. | tee cli.out | ./srv
Error : invalid option
kw2019202032@ubuntu:~/Assignment1_3$ ./cli cd practice | tee cli.out | ./srv
CWD practice
"/home/kw2019202032/Assignment1_3/practice" is current directory
kw2019202032@ubuntu:~/Assignment1_3$ ./cli cd .. | tee cli.out | ./srv
CDUP
"/home/kw2019202032" is current directory
kw2019202032@ubuntu:~/Assignment1_3$ ./cli cd ./not_exist_path | tee cli.out | .
/srv
Error : directory not found
```

특정 directory 로 이동하거나 이전 파일로 이동하고 있는 것을 확인할 수 있다.

- MKD

```
kw2019202032@ubuntu:~/Assignment1_3$ ./cli mkdir -e new_dir | tee cli.out | ./sr
v
Error : invalid option
kw2019202032@ubuntu:~/Assignment1_3$ ./cli mkdir new_dir | tee cli.out | ./srv
MKD new_dir
kw2019202032@ubuntu:~/Assignment1_3$ ./cli mkdir new_dir1 new_dir2 | tee cli.out
 | ./srv
MKD new_dir1
MKD new_dir2
kw2019202032@ubuntu:~/Assignment1_3$ ./cli mkdir new_dir1 | tee cli.out | ./srv
Error : cannot create directory 'new_dir1': File exists
kw2019202032@ubuntu:~/Assignment1_3$ ./cli mkdir | tee cli.out | ./srv
Error : argument is required
```

Directory 를 생성하고, 이미 존재하는 directory 를 만들거나 옵션을 사용할 경우 예외처리를 하고 있다.

- DELE

```
kw2019202032@ubuntu:~/Assignment1_3$ touch test1.dat
kw2019202032@ubuntu:~/Assignment1_3$ touch test2.dat
kw2019202032@ubuntu:~/Assignment1_3$ touch test3.dat
kw2019202032@ubuntu:~/Assignment1_3$ ./cli delete test3.dat | tee cli.out | ./srv
DELE test3.dat
kw2019202032@ubuntu:~/Assignment1_3$ ./cli delete test2.dat | tee cli.out | ./srv
DELE test2.dat
Ckw2019202032@ubuntu:~/Assignment1_3$./cli delete test1.dat | tee cli.out | ./srv
DELE test1.dat
Ckw2019202032@ubuntu:~/Assignment1_3$
kw2019202032@ubuntu:~/Assignment1_3$ touch test1.dat
kw2019202032@ubuntu:~/Assignment1_3$ touch test2.dat
kw2019202032@ubuntu:~/Assignment1_3$ touch test3.dat
kw2019202032@ubuntu:~/Assignment1_3$ ls -al *.dat
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 00:12 test1.dat
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 00:12 test2.dat
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 00:12 test3.dat
kw2019202032@ubuntu:~/Assignment1_3$ ./cli delete test1.dat | tee cli.out | ./srv
DELE test1.dat
kw2019202032@ubuntu:~/Assignment1_3$ ./cli delete test2.dat | tee cli.out | ./srv
DELE test2.dat
kw2019202032@ubuntu:~/Assignment1_3$ ./cli delete test3.dat | tee cli.out | ./srv
DELE test3.dat
kw2019202032@ubuntu:~/Assignment1_3$ ls -al *.dat
ls: cannot access '*.dat': No such file or directory
kw2019202032@ubuntu:~/Assignment1_3$ touch test.dat
kw2019202032@ubuntu:~/Assignment1_3$ ./cli delete -e test.dat | tee cli.out | ./srv
Error : invalid option
kw2019202032@ubuntu:~/Assignment1_3$ ./cli delete not_exist_file test.dat | tee cli.out | ./srv
Error : failed to delete 'not_exist_file'
DELE test.dat
```

.dat 파일을 touch 로 만들고 dele 명령어를 통해 삭제하는 동작을 수행하고 있음을 확인할 수 있다.

- RMD

```
kw2019202032@ubuntu:~/Assignment1_3$ mkdir new_dir_1
kw2019202032@ubuntu:~/Assignment1_3$ mkdir new_dir_2
kw2019202032@ubuntu:~/Assignment1_3$ mkdir new_dir_3
kw2019202032@ubuntu:~/Assignment1_3$ ./cli rmdir new_dir_1 | tee cli.out | ./srv
RMD new_dir_1
kw2019202032@ubuntu:~/Assignment1_3$ ls -al new_dir_1
ls: cannot access 'new_dir_1': No such file or directory
kw2019202032@ubuntu:~/Assignment1_3$ ./cli rmdir new_dir_1 | tee cli.out | ./srv
Error : failed to remove 'new_dir_1'
kw2019202032@ubuntu:~/Assignment1_3$ ./cli rmdir new_dir_2 new_dir_3 | tee cli.out | ./srv
RMD new_dir_2
RMD new_dir_3
```

```
kw2019202032@ubuntu:~/Assignment1_3$ mkdir new_dir_1
kw2019202032@ubuntu:~/Assignment1_3$ mkdir new_dir_2
kw2019202032@ubuntu:~/Assignment1_3$ mkdir new_dir_3
kw2019202032@ubuntu:~/Assignment1_3$ ./cli rmdir new_dir_1 | tee cli.out | ./srv
RMD new_dir_1
kw2019202032@ubuntu:~/Assignment1_3$ ls -al new_dir_1
ls: cannot access 'new_dir_1': No such file or directory
kw2019202032@ubuntu:~/Assignment1_3$ ./cli rmdir new_dir_1 | tee cli.out | ./srv
Error : failed to remove 'new_dir_1'
kw2019202032@ubuntu:~/Assignment1_3$ ./cli rmdir new_dir_2 new_dir_3 | tee cli.out | ./srv
RMD new_dir_2
RMD new_dir_3
kw2019202032@ubuntu:~/Assignment1_3$ mkdir new_dir
kw2019202032@ubuntu:~/Assignment1_3$ ./cli rmdir not_exist_dir new_dir | tee cli.out | ./srv
Error : failed to remove 'not_exist_dir'
RMD new_dir
kw2019202032@ubuntu:~/Assignment1_3$ ./cli rmdir | tee cli.out | ./srv
Error : argument is required
```

rmdir 명령어를 통해 특정 directory 를 지우고 있는 것을 확인할 수 있다.

RNFR&RNTO

```
kw2019202032@ubuntu:~/Assignment1_3$ touch a b
kw2019202032@ubuntu:~/Assignment1_3$ ls -al a b
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:12 a
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:12 b
kw2019202032@ubuntu:~/Assignment1_3$ ./cli rename a b | tee cli.out | ./srv
Error : name to change already exists
kw2019202032@ubuntu:~/Assignment1_3$ ./cli rename a c | tee cli.out | ./srv
RNFR a
RNTO c
kw2019202032@ubuntu:~/Assignment1_3$ ls -al a b c
ls: cannot access 'a': No such file or directory
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:12 b
-rw-rw-r-- 1 kw2019202032 kw2019202032 0 Apr 16 09:12 c
kw2019202032@ubuntu:~/Assignment1_3$ ./cli rename | tee cli.out | ./srv
Error : two arguments are required
```

a 라는 이름의 파일을 b 라는 이름으로 바꾸는 것을 확인할 수 있다.

QUIT

```
kw2019202032@ubuntu:~/Assignment1_3$ ./cli quit | tee cli.out | ./srv
QUIT success
kw2019202032@ubuntu:~/Assignment1_3$ ./cli quit -e | tee cli.out | ./srv
Error : invalid option
kw2019202032@ubuntu:~/Assignment1_3$ ./cli quit arg | tee cli.out | ./srv
Error : argument is not required
```

Quit 명령어를 통해 종료하고 있음을 알 수 있다.

고찰

해당 과제를 진행하는 과정에서 read 함수과 write 를 사용한 standard input & output 을 진행하였다. 이 과정에서 strcat 와 strcpy 함수 등을 사용하여 buffer 에 output 을 적고 출력을 하였는데, NULL 문자로 인해서 출력이 정상적으로 진행되지 않는 것을 확인할 수 있었다. 특히 cli 에서 srv 로 buffer 를 전달하는 과정에서 NULL 문자가 예상하지 못한 곳에 발생하였고, 이로인해 srv 에서 입력을 처리할 때 NULL 문자로 인해 어려움을 겪었다. 이 문제를 해결하기 위해 알아본 결과 buffer 에 write 를 할 때 '₩0'을 통해 입력의 끝을 확실하게 하고, buffer 를 비워줘야 한다는 것을 알게 되었고, 이를 통해 문제를 해결할 수 있었다.

Reference

강의자료만 참고하였습니다.