

## 시스템프로그래밍 3분반 3번째 과제 32203660 이현정

### mycp 프로그램 작성

#### 1. snapshot

```
sys32203660@embedded:~$ vi mycp.c
sys32203660@embedded:~$ ls -l
total 56
-rw-rw-r-- 1 sys32203660 sys32203660 27 10월 13 23:23 alpha.txt
-rw-r--r-- 1 sys32203660 sys32203660 8980 4월 20 2016 examples.desktop
-rwxrwxr-x 1 sys32203660 sys32203660 5476 10월 14 01:48 mycp
-rw-rw-r-- 1 sys32203660 sys32203660 1011 10월 14 01:50 mycp.c
-rw-rw-r-- 1 sys32203660 sys32203660 258 9월 28 21:03 my_favorite_poem.txt
-rw-rw-r-- 1 sys32203660 sys32203660 83 9월 28 21:06 test.c
-rw-rw-r-- 1 sys32203660 sys32203660 83 9월 28 21:08 test_copy.c
-rw-rw-r-- 1 sys32203660 sys32203660 884 9월 28 21:10 test.o
-rwxrwxr-x 1 sys32203660 sys32203660 4712 9월 28 21:11 test.out
-rw-rw-r-- 1 sys32203660 sys32203660 515 9월 28 21:10 test.s
sys32203660@embedded:~$
sys32203660@embedded:~$ gcc -o mycp mycp.c
sys32203660@embedded:~$
sys32203660@embedded:~$ ./mycp
USAGE: ./mycp source_file destination_file
sys32203660@embedded:~$ ./mycp alpha alpha_new.txt
Open fail for read. No such file
sys32203660@embedded:~$ ./mycp alpha.txt alpha_new.txt
sys32203660@embedded:~$
sys32203660@embedded:~$ ls -l
total 60
-rw-r--r-- 1 sys32203660 sys32203660 27 10월 14 01:50 alpha_new.txt
-rw-rw-r-- 1 sys32203660 sys32203660 27 10월 13 23:23 alpha.txt
-rw-r--r-- 1 sys32203660 sys32203660 8980 4월 20 2016 examples.desktop
-rwxrwxr-x 1 sys32203660 sys32203660 5476 10월 14 01:50 mycp
-rw-rw-r-- 1 sys32203660 sys32203660 1011 10월 14 01:50 mycp.c
-rw-rw-r-- 1 sys32203660 sys32203660 258 9월 28 21:03 my_favorite_poem.txt
-rw-rw-r-- 1 sys32203660 sys32203660 83 9월 28 21:06 test.c
-rw-rw-r-- 1 sys32203660 sys32203660 83 9월 28 21:08 test_copy.c
-rw-rw-r-- 1 sys32203660 sys32203660 884 9월 28 21:10 test.o
-rwxrwxr-x 1 sys32203660 sys32203660 4712 9월 28 21:11 test.out
-rw-rw-r-- 1 sys32203660 sys32203660 515 9월 28 21:10 test.s
sys32203660@embedded:~$
sys32203660@embedded:~$ cat alpha.txt
abcdefghijklmnopqrstuvwxyz
sys32203660@embedded:~$ cat alpha_new.txt
abcdefghijklmnopqrstuvwxyz
sys32203660@embedded:~$ ./mycp alpha.txt alpha_new.txt
Can not create alpha_new.txt. May already exist
sys32203660@embedded:~$

sys32203660@embedded:~$ vi mycp.c
sys32203660@embedded:~$ gcc -o mycp mycp.c
sys32203660@embedded:~$ ./mycp alpha.txt alpha_attr.txt
sys32203660@embedded:~$
sys32203660@embedded:~$ ls -l
total 64
-rw-rw-r-- 1 sys32203660 sys32203660 27 10월 14 01:52 alpha_attr.txt
-rw-r--r-- 1 sys32203660 sys32203660 27 10월 14 01:50 alpha_new.txt
-rw-rw-r-- 1 sys32203660 sys32203660 27 10월 13 23:23 alpha.txt
-rw-r--r-- 1 sys32203660 sys32203660 8980 4월 20 2016 examples.desktop
-rwxrwxr-x 1 sys32203660 sys32203660 5852 10월 14 01:51 mycp
-rw-rw-r-- 1 sys32203660 sys32203660 1005 10월 14 01:51 mycp.c
-rw-rw-r-- 1 sys32203660 sys32203660 258 9월 28 21:03 my_favorite_poem.txt
-rw-rw-r-- 1 sys32203660 sys32203660 83 9월 28 21:06 test.c
-rw-rw-r-- 1 sys32203660 sys32203660 83 9월 28 21:08 test_copy.c
-rw-rw-r-- 1 sys32203660 sys32203660 884 9월 28 21:10 test.o
-rwxrwxr-x 1 sys32203660 sys32203660 4712 9월 28 21:11 test.out
-rw-rw-r-- 1 sys32203660 sys32203660 515 9월 28 21:10 test.s
sys32203660@embedded:~$ whoami
sys32203660
sys32203660@embedded:~$ date
2021. 10. 14. (수) 01:52:34 KST
sys32203660@embedded:~$ cat alpha_attr.txt
abcdefghijklmnopqrstuvwxyz
sys32203660@embedded:~$
```

## 2. mycp.c 코드

```
sys32203660@embedded: ~
/* mycopy program, by HyeonJeong.32203660@dankook.ac.kr */
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <errno.h>
#include <sys/stat.h>
#include <sys/types.h>
#define MAX_BUF 4096

int main(int argc, char *argv[])
{
    int fd, read_size, write_size, src_fd, dest_fd;
    char buf[MAX_BUF];

    if (argc != 3) {
        printf("USAGE: %s source_file_destination_file\n", argv[0]); exit(-1);
    }

    src_fd = open(argv[1], O_RDONLY);
    if (src_fd < 0) {
        printf("Open fail for read. No such file\n"); exit(-1);
    }

    dest_fd = open(argv[2], O_WRONLY|O_CREAT|O_EXCL, 0644);
    if (dest_fd < 0) {
        printf("Can not create alpha_new.txt. May already exist\n"); exit(-1);
    }

    while (1) {
        read_size = read(src_fd, buf, MAX_BUF);
        write_size = write(dest_fd, buf, read_size);
        if (write_size == 0) {
            break;
        }
    }

    struct stat sts;
    fstat(src_fd, &sts);
    chmod(argv[2], sts.st_mode);

    close(src_fd);
    close(dest_fd);
}
```

첫번째 vi mycp.c에서는 38 ~ 40줄의 내용을 제외하고 작성하였습니다.

argc으로 우선 원하는 형태로 mycp를 사용하는지 확인하고, 오류시에는 그에 대한 문장을 출력하도록 구성하였습니다.

읽을 수 있게 argv[1]을 열었고(읽을 파일을 여는 과정에서 문제 발생시, 주어진 문장을 출력하고 작업이 끝나게 만들었음), rw-r—w—를 형태를 가질 수 있게 argv[2]를 열게 만들었고(쓸 파일을 여는 과정에서 문제 발생시, 주어진 문장을 출력하고 작업이 끝나게 만들었음), 파일이 존재하면 새로 생성하지 않게 O\_EXCL을 사용하였습니다.

while문 내에서는 읽는 파일의 내용을 최대 MAX\_BUF값만큼 읽어서 buf에 저장하도록 하고, 해당 buf를 이용해서 쓰는 파일에 내용이 작성되게 하였습니다. write\_size가 0인 경우는 read\_size 또한 0이었다는 의미이며, 쓸 내용이 없다는 의미이기에 while문에서 벗어날 수 있게 구성하였습니다.

마지막으로 해당하는 파일들의 이용이 종료되었으므로 연 파일들을 닫아주었습니다.

두번째 vi mycp.c에서는 38 ~ 40줄의 내용을 넣었습니다.

해당 구조체를 통해서 읽는 파일의 접근 권한을 받았고, 해당 접근 권한을 쓰는 파일의 접근 권한으로 지정하게 구성하였습니다.

### 3. 시행 착오

#### 1) ./mycp 실행시 오류

교수님의 예시 shanpshot과는 다른 문장이 출력되어서 내부적으로 코드가 돌아가는데 문제가 있다고 생각을 하였습니다. 그러다가 해당 문장이 `arg == 3`이 아닌 경우에 출력되는 문장이라는 것을 깨닫고, 각 오류마다 정해진 문장을 출력하도록 구현해야겠다는 생각을 할 수 있게 되었습니다.

#### 2) `read(dest_fd, ~)`에서 마지막(3번째)으로 주어지는 argument

해당 자리를 빈 자리로 놓으니, 읽지도 쓰지도 못하는 파일이 되었습니다. 그 다음으로 0664를 넣으니 `rw-rw-r---`형태가 되어서 이것을 응용해서 `rw-r--w---`형태를 만들어야겠다는 생각을 했고, 수업시간에 배운 내용을 통해서 `r -> 4, w -> 2, x -> 1`이기에 0644를 통해서 해당 형태로 mycp파일의 접근 권한을 설정할 수 있었습니다.

#### 3) `write()`

처음에는 이제까지 배운 문장들이 모두 1번의 `read()`를 통해 한번에 buf에 가져와서 저 역시 buf에 모든 내용을 담았다가 전달하려는 생각도 해보았지만, buf의 용량이 부족할 것이라는 생각을 했기에 buf로 파일의 내용을 읽어왔으면 바로 해당 내용을 쉘파일에 전달하도록 하는 것이 반복되게 실행해야겠다는 생각을 하였습니다. 그래서 while문에 해당 내용을 넣고, 쓸 내용이 없는 경우에 while문 밖으로 나오게 구성하였습니다.

#### 4) `chmod()`

처음에는 어떤 것을 사용해야할지 모르다가, 복습하다보니 해당 함수가 있다는 사실을 알 수 있었습니다. 다음으로는 어떻게 읽는 파일의 접근 권한을 읽어올 수 있을지를 찾아보니, 구조체를 통해서 해당 내용을 읽어올 수 있어서 그것을 이용해서 쓰는 파일 또한 같은 접근 권한을 가지도록 구성하였습니다.