

## [506489] 시스템프로그래밍

### 실습 #05 문제 및 보고서

이름	곽영주
학번	20175105
소속 학과/대학	빅데이터
분반	01 (담당교수: 김태운)

### <주의사항>

- 개별 과제입니다. (팀으로 진행하는 과제가 아니며, 모든 학생이 보고서를 제출해야 함)
- **각각의 문제 바로 아래에 답을 작성 후 제출해 주세요.**
  - 소스코드/스크립트 등을 작성 한 경우, 해당 파일의 이름도 적어주세요.
- **SmartLEAD 제출 데드라인:**
  - **월요일 분반: 다음 실습 시간 전날까지(일요일 까지)**
  - **수요일 분반: 다음 실습 시간 전날까지 (화요일 까지)**
  - 데드라인을 지나서 제출하면 24 시간 단위로 20%씩 감점(5 일 경과 시 0 점)
  - 주말/휴일/학교행사 등 모든 날짜 카운트 함
  - 부정행위 적발 시, 원본(보여준 사람)과 복사본(베낀 사람) 모두 0 점 처리함
  - 예외 없음
- **SmartLEAD 에 아래의 파일을 제출해 주세요**
  - **보고서(PDF 파일로 변환 후 제출 권장하나, WORD 형식으로 제출도 가능)**
  - 보고서 파일명에 이름과 학번을 입력해 주세요.
  - **소스코드, 스크립트, Makefile 등을 작성해야 하는 경우, 모든 파일 제출 (zip 파일로 압축하여 하나의 파일로 제출)**

### <개요>

이번 과제는 리눅스 운영체제에서 파일 및 디렉토리를 프로그래밍 하는 내용입니다. 그리고 ncurses 라이브러리를 사용하는 과제가 포함되어 있습니다.

## &lt;실습 과제&gt;

**[Q 0] 요약 [배점: 10]**

이번 과제에서 배운 내용 또는 과제 완성을 위해서 무엇을 했는지 2~3 문장으로 요약하세요.

답변: 리눅스 명령어들의 동작을 직접 프로그래밍 해보면서 리눅스 명령어들이 어떠한 방식으로 동작하는지 배울 수 있었습니다. 마지막으로 ncurses 를 간단하게 맛보면서 ncurses 가 무엇인지 알게 되었습니다.

**[Q 1] mystat [배점: 15]**

파일의 정보를 추출하고 그 결과를 아래와 같이 터미널에 출력하는 프로그램을 작성하세요.  
정보를 조회하려는 파일의 이름은 명령 행 인자로 받습니다.

<실행 예: 파일명이 xxx.c 인 경우>

입력	\$mystat xxx.c
출력	Filename: xxx.c - inode number: 192 - File type: regular file - Permission: 644 - Number of links: 1 - Size : 24

프로그램을 예시와 같이 구동하고 터미널 출력 화면을 캡처해서 본 문서에 첨부하세요.  
소스코드도 첨부파일로 제출해야 합니다.

답변:

```

yeongju@vm-ubuntu20:~/sp2021-2/w05/q01$ ls -al test.txt
-rw-rw-r-- 1 yeongju yeongju 6  9월  28 02:39 test.txt
yeongju@vm-ubuntu20:~/sp2021-2/w05/q01$ cat test.txt
Hello
yeongju@vm-ubuntu20:~/sp2021-2/w05/q01$ ./mystat test.txt
Filename: test.txt
- inode number: 1835610
- File type: Regular File
- Permission: 664
- Number of links: 1
- Size: 6

```

### [Q 2] mychmod [배점: 15]

입력 받은 파일의 접근 권한을 출력하고, 접근 권한을 변경하는 프로그램을 작성하세요.

<실행 예 1: 파일명만 인자로 주어진 경우: 접근 권한을 아래와 같이 출력>

입력	\$mychmod xxx.c
출력	Permission : rw-r--r--

<실행 예 2: 접근 권한이 숫자로 주어진 경우: 주어진 숫자에 맞게 접근 권한을 변경>

입력	\$mychmod 777 xxx.c
출력	Permission (before) : rw-r-xr-- Permission (after) : rwxrwxrwx

프로그램을 위의 예시와 같이 구동하고 각각의 터미널 출력 화면을 캡처해서 본 문서에 첨부하세요. 소스코드도 첨부파일로 제출해야 합니다.

답변 1:

```

yeongju@vm-ubuntu20:~/sp2021-2/w05/q02$ ls -al test.txt
-rw-rw-r-- 1 yeongju yeongju 0  9월  28 02:41 test.txt
yeongju@vm-ubuntu20:~/sp2021-2/w05/q02$ ./mychmod test.txt
Permission: rw-rw-r--

```

답변 2:

```
yeongju@vm-ubuntu20:~/sp2021-2/w05/q02$ ls -al test.txt
-rw-rw-r-- 1 yeongju yeongju 0  9월 28 02:41 test.txt
yeongju@vm-ubuntu20:~/sp2021-2/w05/q02$ ./mychmod 777 test.txt
permission (before): rw-rw-r--
permission (after): rwxrwxrwx
yeongju@vm-ubuntu20:~/sp2021-2/w05/q02$ ls -al test.txt
-rwxrwxrwx 1 yeongju yeongju 0  9월 28 02:41 test.txt
```

**[Q 3] myChangeDir [배점: 15]**

디렉토리명을 명령행 인자로 입력 받아 디렉토리를 생성하고, 작업 디렉토리를 새로 생성한 디렉토리로 이동한 후, README.TXT 파일을 생성하는 myChangeDir 프로그램을 작성하세요.

프로그램 구동 전에 \$pwd 명령과 \$ls 명령을 실행하세요. 프로그램 구동 후 myChangeDir 에 의해 생성된 디렉토리로 이동하여, \$ls 명령과 \$pwd 명령을 각각 입력하세요. 터미널 출력 화면을 캡처해서 본 문서에 첨부하세요. 소스코드도 첨부파일로 제출해야 합니다.

답변:

```
yeongju@vm-ubuntu20:~/sp2021-2/w05/q03$ pwd
/home/yeongju/sp2021-2/w05/q03
yeongju@vm-ubuntu20:~/sp2021-2/w05/q03$ ls
myChangeDir myChangeDir.c
yeongju@vm-ubuntu20:~/sp2021-2/w05/q03$ ./myChangeDir test
yeongju@vm-ubuntu20:~/sp2021-2/w05/q03$ cd test
yeongju@vm-ubuntu20:~/sp2021-2/w05/q03/test$ ls
README.txt
yeongju@vm-ubuntu20:~/sp2021-2/w05/q03/test$ pwd
/home/yeongju/sp2021-2/w05/q03/test
```

**[Q 4] myls [배점: 15]**

\$ls 명령과 유사한 기능을 하는 myls 프로그램을 작성하세요. 'mysls' 명령은 현재 디렉토리 내에 있는 모든 파일 및 디렉토리를 아래와 같이 이름만 출력합니다(단, '.' 과 '..' 을 제외).

<실행 예>

입력	\$mysls
출력	<Regular Files>  a.out helloworld.c parsing.c  <Directories> subdir1 subdir2

현재 디렉토리에 일반 파일, 디렉토리를 생성하세요(위의 예제에서 출력된 목록과 같이 생성하세요). 프로그램을 예시와 같이 구동하고 터미널 출력 화면을 캡처해서 본 문서에 첨부하세요. 소스코드도 첨부파일로 제출해야 합니다.

답변:

```
yeongju@vm-ubuntu20:~/sp2021-2/w05/q04$ ./mysls
<Regular Files>
parsing.c
a.out
helloworld.c

<Directories>
subdir1
subdir2
```

[Q 5] myFileMove [배점: 15]

파일을 이동하는(또는 이름을 변경하는) myFileMove 프로그램을 작성하세요.

<실행 예> `$myFileMove src.txt dst.txt //` 동일한 디렉토리 내에서 이동

터미널에서 `src.txt` 라는 파일을 디렉토리를 생성하고(예: `$touch src.txt`), `$ls` 명령을 실행하세요. <실행 예>와 같이 실행한 뒤에 `$ls` 명령을 다시 실행하세요. 터미널 출력 화면을 캡처해서 본 문서에 첨부하세요.

소스코드도 첨부파일로 제출해야 합니다.

답변:

```
yeongju@vm-ubuntu20:~/sp2021-2/w05/q05$ touch src.txt
yeongju@vm-ubuntu20:~/sp2021-2/w05/q05$ ls
myFileMove myFileMove.c src.txt
yeongju@vm-ubuntu20:~/sp2021-2/w05/q05$ ./myFileMove src.txt dst.txt
yeongju@vm-ubuntu20:~/sp2021-2/w05/q05$ ls
dst.txt myFileMove myFileMove.c
```

#### [Q 6] ncurses 맛보기 [배점: 15]

1. ncurses 설치하기: `$sudo apt install libncurses5-dev libncursesw5-dev`

2. Hello World 프로그램 및 컴파일 하기 [hello.c]

```
#include <ncurses.h>

int main(void) {
    initscr();                /* Start curses mode          */
    printw("Hello World !!!"); /* Print Hello World          */
    refresh();                /* Print it on to the real screen */
    getch();                  /* Wait for user input */
    endwin();                  /* End curses mode            */

    return 0;
}
```

컴파일 방법: `$gcc <filename> -lncurses`

## 3. 맛보기: 키보드에서 누른 키 값을 화면에 볼드체로 출력하기 [boldPrint.c]

```

1  #include <ncurses.h>
2
3  int main()
4  {      int ch;
5
6          initscr();                /* Start curses mode          */
7          raw();                    /* Line buffering disabled    */
8          keypad(stdscr, TRUE);     /* We get F1, F2 etc..        */
9          noecho();                 /* Don't echo() while we do getch */
10
11        printw("Type any character to see it in bold\n");
12         ch = getch();              /* If raw() hadn't been called
13                                     * we have to press enter before it
14                                     * gets to the program          */
15         if(ch == KEY_F(1))         /* Without keypad enabled this will */
16             printw("F1 Key pressed"); /* not get to us either          */
17                                     /* Without noecho() some ugly escape
18                                     * characters might have been printed
19                                     * on screen                      */
20         else
21         {       printw("The pressed key is ");
22                 attron(A_BOLD);
23                 printw("%c", ch);
24                 attroff(A_BOLD);
25             }
26         refresh();                 /* Print it on to the real screen */
27         getch();                   /* Wait for user input          */
28         endwin();                  /* End curses mode              */
29
30         return 0;
31     }

```

[bolPrint.c] 파일을 컴파일 후 실행하세요. 터미널 출력 화면(입력한 글자가 볼드체로 표시되는 화면)을 캡처하여 본 문서에 첨부하세요. 소스코드도 첨부하여 제출하세요

답변 (boldPrint.c 컴파일 후 실행화면):



---

```
yeongju@vm-ubuntu20: ~/sp2021-2/w05/q06
type any character to see it in bold
The pressed key is a
```

끝! 수고하셨습니다 ☺