컴퓨터학부 20180713 김누리

실습과제 #3\_1. ‘ls’ 명령어와 동일한 기능을 수행하는 ‘myls’ 명령어를 구현하시오.

1. 구현 개요

ls -l 옵션은 교재코드를 기반으로 만들었다.

1. 소스코드
2. 실행결과(캡쳐)
3. 구현을 통해 배운 점

Ls 명령어와 구현에 필요한 여러 함수와 구조체에 대해 잘 알게 되었다

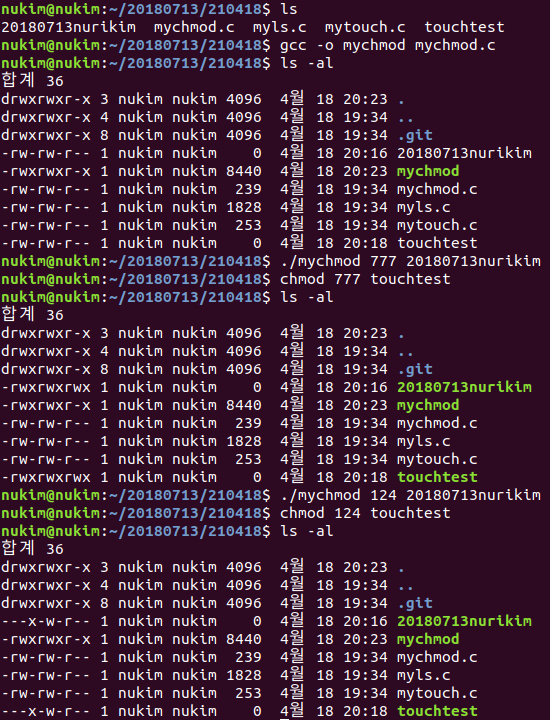
실습과제 #3\_2. ‘chmod’ 명령어와 동일한 기능을 수행하는 ‘mychmod’ 명령어를 구현하시오.

1. 구현 개요

Permission은 숫자로만 입력받음. 기호 입력은 구현 안함.

chmod() 함수를 이용하여 간단하게 구현.

1. 소스코드
2. #include <stdio.h>
3. #include <sys/stat.h>
4. void mychmod (char \*modeNumber, char \*filename)
5. {
6. int mode;
7. sscanf(modeNumber,"%o", &mode);
8. chmod(filename, mode);
9. }
10. int main(int argc, char \*argv[])
11. {
12. mychmod(argv[1], argv[2]);
13. return 0;
14. }
15. 실행결과(캡쳐)



1. 구현을 통해 배운 점

chmod 함수 사용없이 구현해보고 싶어서 공부하다가 chmod 함수에 대해 잘 알게되었다

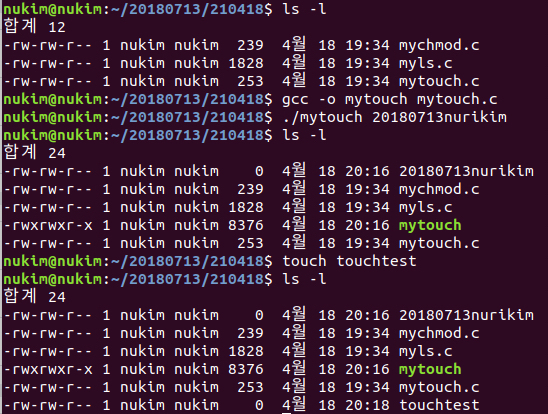
실습과제 #3\_3. ‘touch’ 명령어와 동일한 기능을 수행하는 ‘mytouch’ 명령어를 구현하시오.

1. 구현 개요

touch 함수 사용시 생성되는 함수의 파일 접근권한을 체크한 후 구현했다.

3줄로 구현했더니 타임스템프가 안 바꿔져서 다시 공부했다.

1. 소스코드
2. #include <sys/types.h>
3. #include <sys/stat.h>
4. #include <fcntl.h>
5. #include <unistd.h>
6. #include <stdio.h>
7. #include <time.h>
8. #include <utime.h>
9. void mytouch (char \*filename[])
10. {
11. int fd;
12. struct stat st;
13. struct utimbuf buf;
14. lstat(filename[1], &st);
15. fd = open(filename[1],O\_RDWR| O\_CREAT,0664);
16. close(fd);
17. buf.actime = time(NULL);
18. buf.modtime = time(NULL);
19. utime(filename[1], &buf);
20. }
21. int main(int argc, char \*argv[])
22. {
23. mytouch(argv);
24. return 0;
25. }
26. 실행결과(캡쳐)



추가로 파일의 타임스템프를 변경하는 용도 테스트 한번 더 함

텍스트이(가) 표시된 사진

자동 생성된 설명

1. 구현을 통해 배운 점

처음에 엄청 복잡하게 생각했는데 생각보다 간단해서 신기했다.