NETWORK ASSIGNMENT REPORT

NETWORK ASSIGNMENT2 - CONCURRENT FILE COPIES

2014310198

NAM JAEYOUN

DEVELOPMENT ENVIRONMENT

Operating system
: Linux version 4.4.0-31-generic
(swye cluster)

Compiler version
: gcc version 5.3.1

Programming languague
: C

HOW TO TEST PROGRAM

1. Put 'fileCopyMulti.c', 'Makefile' into the same directory.

2. Enter 'make' command to make a runnable file named 'multicopy'.

3. Enter './fileCopy' to run program.

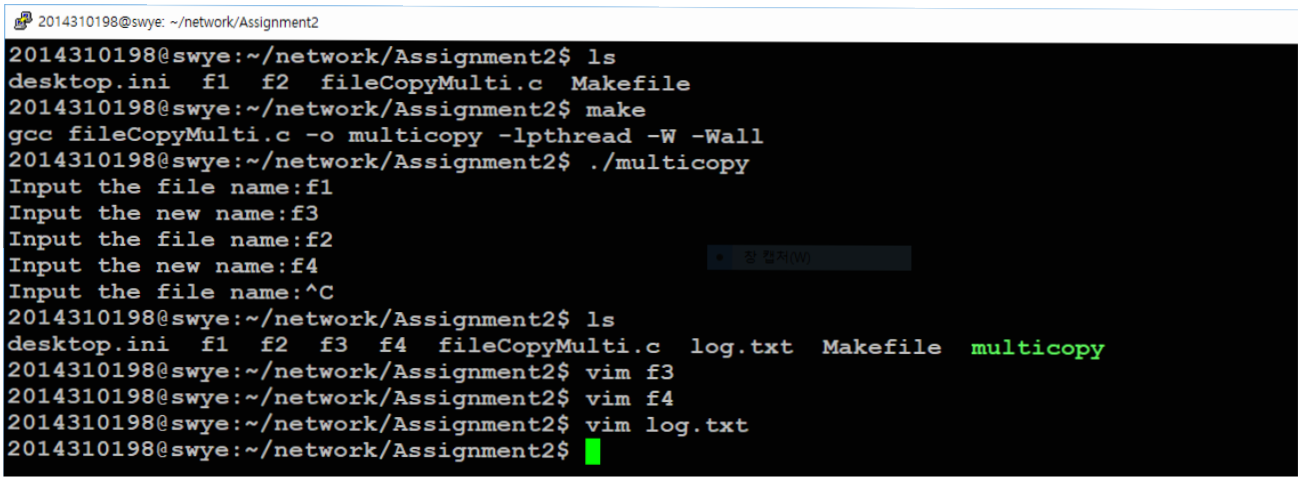

FIGURE 1. HOW TO RUN

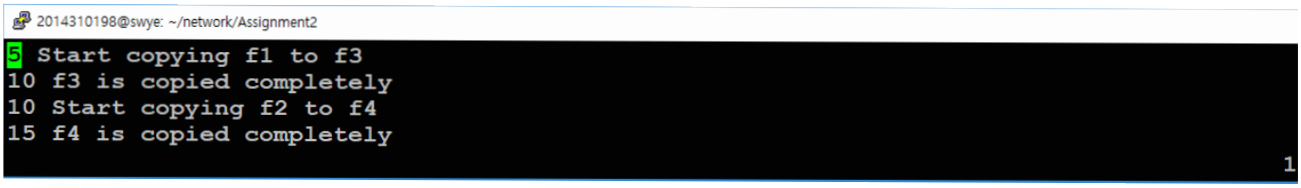

FIGURE 2. LOG.TXT

NETWORK ASSIGNMENT REPORT

DATA STRUCTURE

FNAME structure

: 입력받은 파일이름을 쓰레드로 전해주기위한 구조체

char input[100] // 복사할 파일의 이름

char output[100] // 새로 만들 파일의 이름

PTHREAD WORK FLOW

int main()

: 사용자에게 입력을 받아서 FNAME 스트럭쳐에 저장하고 work함수를 처리하는 쓰레드를 생성한다.

: 쓰레드는 join될 필요가 없으므로 detach한다.

void *work(void *fname)

: 로그에 카피를 시작했음을 기록한다.

: 파일을 카피한다.

: 카피가 끝나면 로그를 다시열어 카피가 끝났음을 기록한다.

MUTEX & SYNC

1. "log.txt"를 두개이상의 쓰레드가 동시에 작성하면 안되므로 mutex로 보호한다.

2. 새로 입력받기전에 입력받은 파일이름들이 쓰레드내의 private variable에 저장되었음을 보장하기위해 mutex_sync와 cond_sync를 사용했다.

```
69      pthread_mutex_lock(&mutex);
70      flog = fopen("log.txt", "a");
71      fprintf(flog, "%ld", time(NULL)-t);
72      fprintf(flog, " Start copying %s to %s\n", filename->input, filena
73      fclose(flog);
74      pthread_mutex_unlock(&mutex);
75
```

FIGURE 3. MUTEX

```
62
63      pthread_mutex_lock(&mutex_sync);
64      strcpy(filename->input,((FNAME *)fname)->input);
65      strcpy(filename->output,((FNAME *)fname)->output);
66      pthread_cond_signal(&cond_sync);
67      pthread_mutex_unlock(&mutex_sync);
68
```

FIGURE 4. SYNCRONIZATION CODE