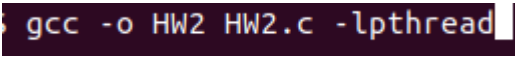


# Computer Network HW 2

학번 / 이름: 2014310407 / 이 준 혁

## 1. 개발 환경

- OS: Linux (Ubuntu 1804)
- Programming language: C
- Compiler: GCC (version 7.3.0)
- Compile option: -lpthread option

Ex) 

## 2. 구현

병렬적인 파일 복사 프로그램을 구축하기 위해, 멀티스레드를 사용하기로 했고, 이 때 pthread 라는 라이브러리를 사용하기로 했다.

이 때 사용되는 pthread\_create 라는 함수는 스레드, 스레드의 속성, 실행시킬 스레드 함수와 스레드 함수로 넘겨줄 변수(void \* type)를 받는다. 그 후 스레드 함수를 실행시키는데, 이를 병렬적으로 실행한다.

우선 파일을 복사하는 스레드 함수를 thread\_copy 라고 선언 후 다음과 같이 동작하도록 만들었다. 그 전에, 이 함수를 실행시키기 위해서는, 복사할 파일의 이름, 복사할 파일의 포인터, 복사될 파일의 이름, 복사될 파일의 포인터가 필요했다. 이 때 스레드 함수는 인자를 하나(void \* type)밖에 받을 수 없으므로 문자 배열 두개와 파일 포인터 두개를 멤버 변수로 가지는 FILE\_ptrs 라는 구조체를 선언했다. 이후 이 프로그램을 실행시킬 때마다 FILE\_ptrs\* 변수를 인자로 넘겨주도록 했다.

이 thread\_copy 함수는 입력에 따라 두 가지 다른 기능을 하도록 설계했다.

#### 1) 입력이 NULL 인 경우: 초기화

최초에 이 함수를 실행시킬 때는 인자를 NULL 로 줘서, 이 함수에서 사용하는 여러 값을 초기화 시켜줬다. 프로그램이 시작된 지 몇 초가 지났는지를 확인해야 하기 때문에, 최초에 이 함수가 실행될 때의 시간을 기록하고자 했다. 따라서 입력이 NULL 인 경우에는, clock 함수(<time.h>에 정의)를 이용해 최초 시간을 기록했다. 이 값은 매 호출 때 마다 사용하므로 static 변수에 저장했다. 또한, 기존에 있는 로그 파일을 없애기 위해 log.txt 파일을 우선 읽기 전용으로 열어서 로그 파일 역시 초기화를 시켰다.

#### 2) 입력이 NULL 이 아닌 경우: 파일 복사

복사하기 전에는, log 파일을 열어서 프로그램 최초 시작 시간과 현재 시간 차이를 기록한 후, 무슨 파일이 어디에 복사되는지에 대해 기록을 남겼다.

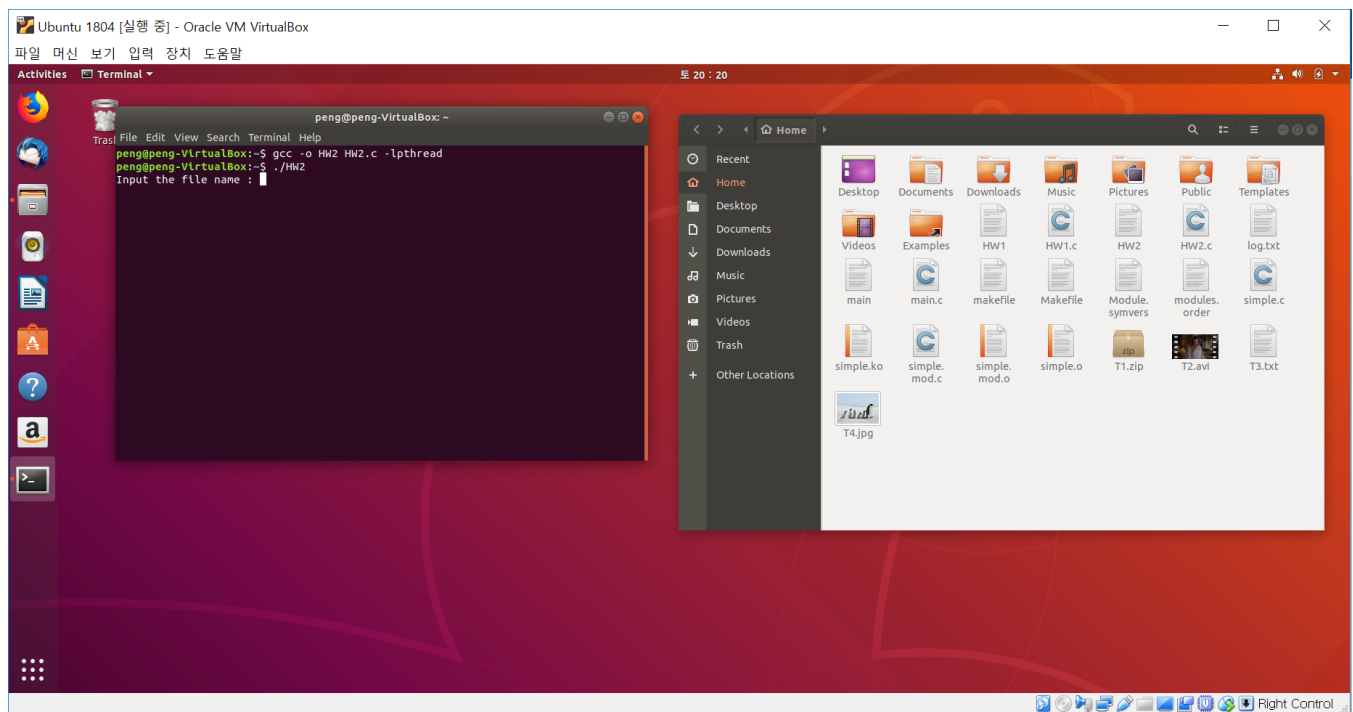
이후는 첫 번째 과제와 마찬가지로 fread 함수와 buffer 라는 문자 배열을 이용해, 원본 파일을 buffer size 만큼 읽어서, 그것을 buffer 에 저장한 후, 복사될 파일에 fwrite 함수를 이용해 썼다. 이를 파일을 끝까지 다 읽을 때까지 반복했다. 이 때, 파일을 끝부분에서는 읽은 양이 buffer size 보다 작을 수 있는데, 이 경우에는 읽은 만큼만 쓰도록 했다. (이렇게 하지 않는 경우 이상한 값이 들어갈 수 있으므로)

이렇게 복사된 후에는, 마찬가지로 log 파일을 열어서 프로그램 최초 시작 시간과 현재 시간 차이를 기록한 후, 복사된 파일이 복사되었다는 것을 기록으로 남겼다. 로그까지 모두 남겼다면 pthread\_exit 함수를 이용해 스레드를 종료시켰다.

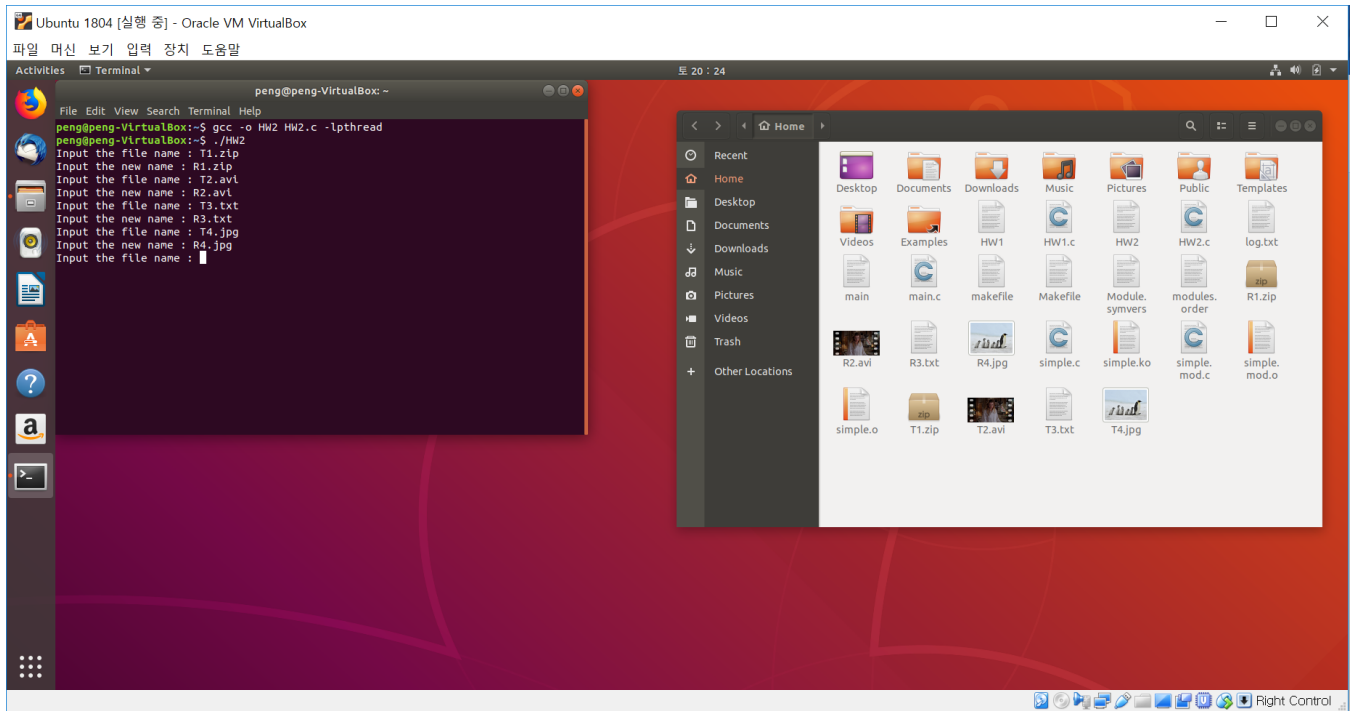
메인 함수 내에서는 가장 먼저 NULL 을 인자로 주어서 thread\_copy 함수를 초기화 시켰다. 이후 while 문을 이용해 계속 복사할 파일의 이름과 복사될 파일의 이름을 입력 받았고, 만약 이름이 잘못되었거나 기타이유로 파일을 열지 못한 경우에는 프로그램을 종료시키도록 했다.

복사할 파일과 복사될 파일의 이름을 제대로 입력 받은 후에 파일 열기까지 완료했다면, 앞에서 설명한 pthread\_create 함수를 이용해 thread\_copy 함수를 병렬적으로 실행시켰다. 이렇게 파일 복사를 병렬적으로 수행했다. while 문의 조건에는 1 을 넣어 별다른 오류가 없다면 프로그램이 종료되는 일은 없도록 구성했다.

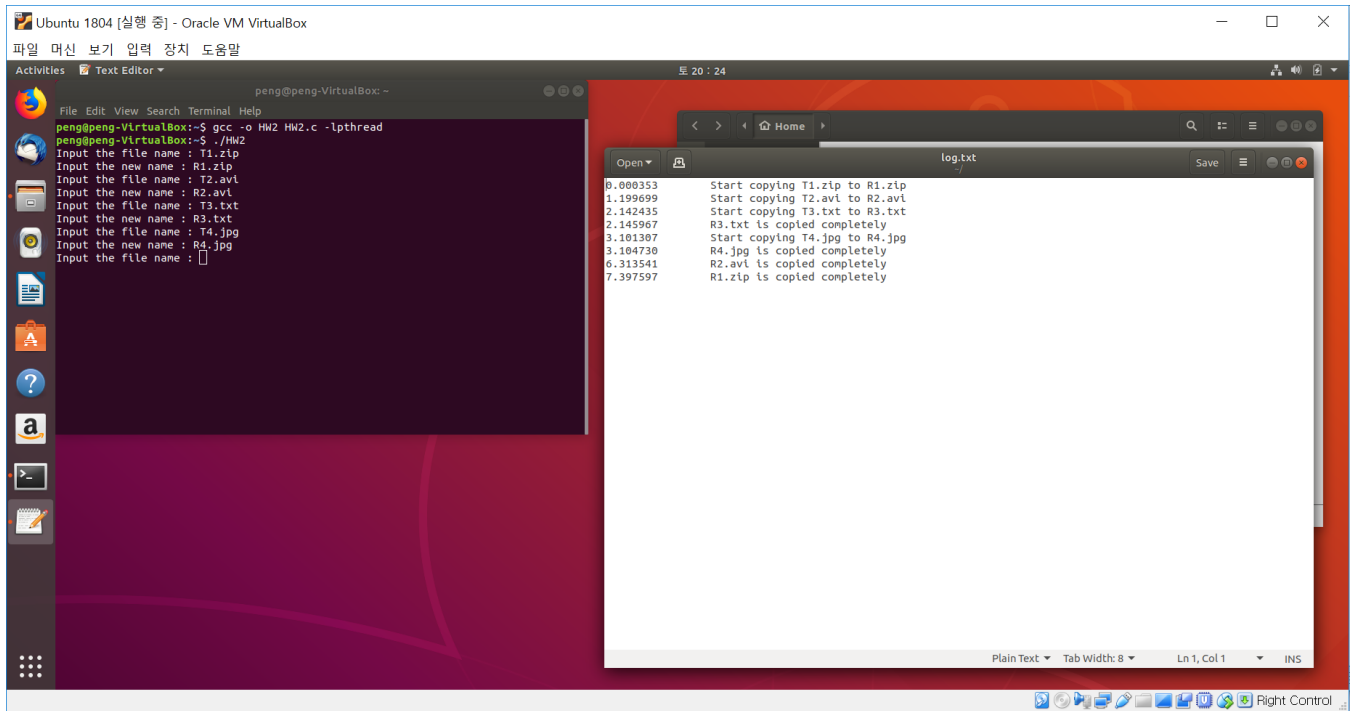
### 3. 결과



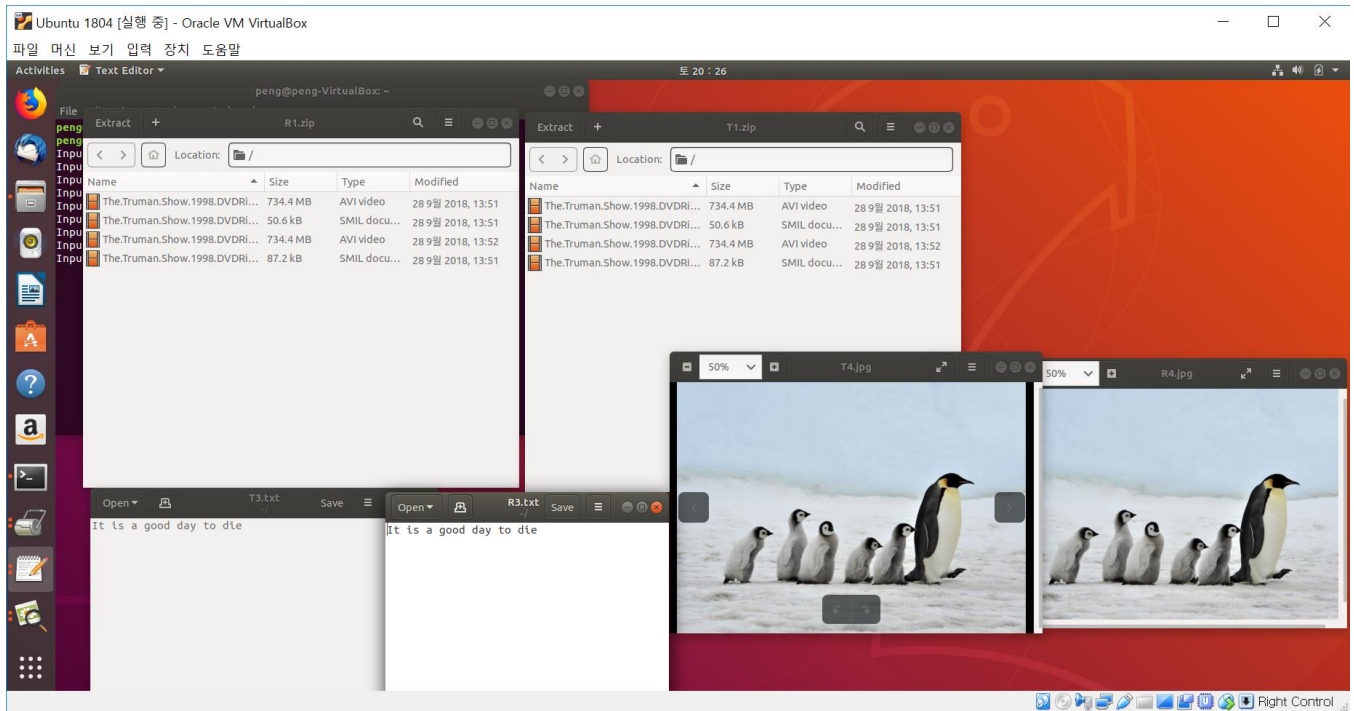
- 다음과 같이 T1.zip, T2.avi, T3.txt, T4.jpg 파일을 디렉토리에 두고 프로그램을 실행시켰다.



- 총 네 개의 파일을 다음과 같이 복사했다.



- 순서대로 파일이 복사되는 것이 아니라 위와 같이 T1, T2 가 복사되는 도중에도 T3, T4 가 복사되는 것을 알 수 있다.



- 파일 복사가 성공적으로 이루어진 것 역시 확인 가능하다.

(avi 파일은 생략했으나 역시 정상적으로 복사되었습니다.)