

Assignment#1

2015311901

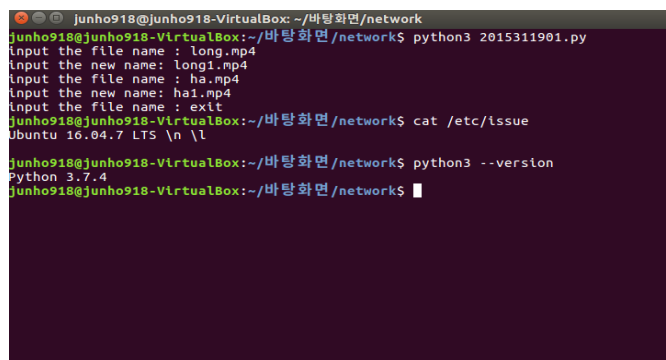
Junho Bae

Computer Networks

1) Development environments

A. OS

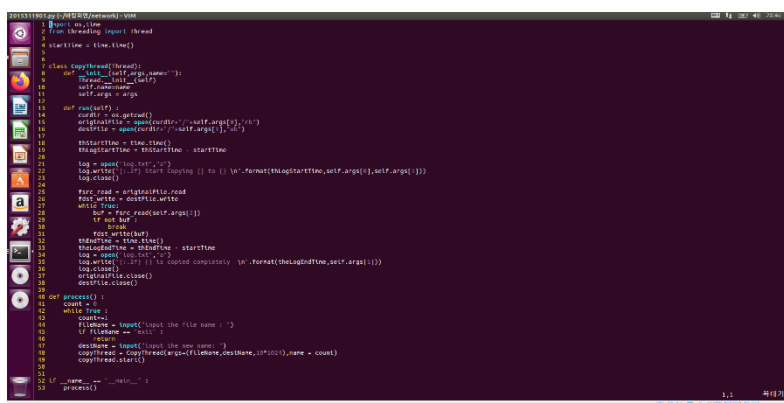
- I feel more familiar with Windows OS. Therefore, first I designed program in Windows10 with python3, then I tested my program in Oracle VirtualBox and Ubuntu(16.04.7) with Python3



```
junho918@junho918-VirtualBox: ~/바탕화면/network
junho918@junho918-VirtualBox:~/바탕화면/network$ python3 2015311901.py
input the file name : long.mp4
input the new name: long1.mp4
input the file name : ha.mp4
input the new name: ha1.mp4
input the file name : exit
junho918@junho918-VirtualBox:~/바탕화면/network$ cat /etc/issue
Ubuntu 16.04.7 LTS \n \l
junho918@junho918-VirtualBox:~/바탕화면/network$ python3 --version
Python 3.7.4
junho918@junho918-VirtualBox:~/바탕화면/network$
```

- There were some difference between Windows and Ubuntu while reading file directory. So, I slightly edited the program to run properly in Ubuntu environment.
- I also make sure that I did not use some libraries which are only dedicated for Windows OS.

2) How I designed assignment (data structures, algorithms)

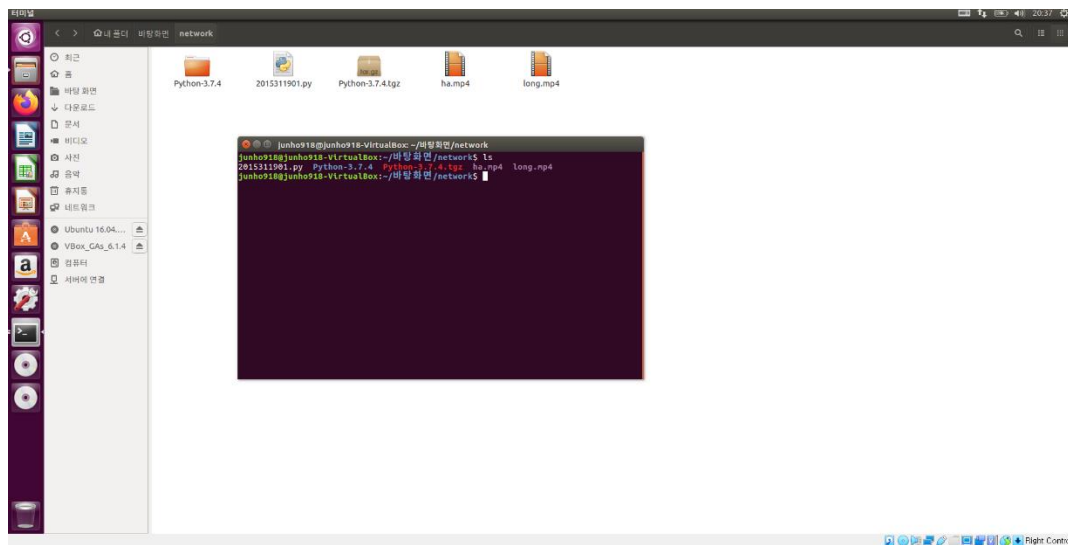


```
1 import sys, time
2 from threading import Thread
3
4 starttime = time.time()
5
6
7 class CopyThread(Thread):
8     def __init__(self,
9                 filename,
10                 destfile,
11                 length):
12         Thread.__init__(self)
13         self.filename = filename
14         self.destfile = destfile
15         self.length = length
16
17     def run(self):
18         cursor = 0
19         while cursor < self.length:
20             cursor += 1024
21             destfile.write(self.filename.read(cursor))
22             log.write("%d%%" % (cursor/self.length*100))
23             log.flush()
24             time.sleep(0.1)
25
26     def run(self):
27         cursor = 0
28         while cursor < self.length:
29             cursor += 1024
30             destfile.write(self.filename.read(cursor))
31             log.write("%d%%" % (cursor/self.length*100))
32             log.flush()
33             time.sleep(0.1)
34
35     def run(self):
36         cursor = 0
37         while cursor < self.length:
38             cursor += 1024
39             destfile.write(self.filename.read(cursor))
40             log.write("%d%%" % (cursor/self.length*100))
41             log.flush()
42             time.sleep(0.1)
43
44 def process():
45     while True:
46         filename = input("input the file name : ")
47         if filename == "exit":
48             break
49         destfile = input("input the new name : ")
50         copythread = CopyThread(filename, destfile, 1024*10)
51         copythread.start()
52
53 if __name__ == "__main__":
54     process()
```

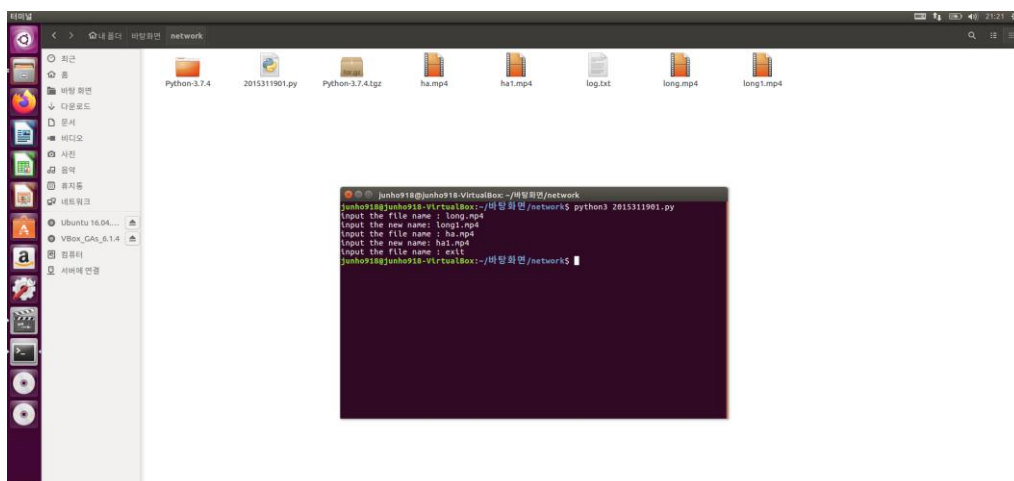
- If screenshot is too small, please check my source file.
- I implemented "CopyThread" class which inherits 'threading.Thread' so that I can implement multi threading program. The CopyThread class requires three arguments, source file name, destination file name, and length. I gave length argument as 1024*10 which means 10kb instructed in assignment notice. Therefore, program will read 10Kb of source file with buffer and write buffer into the new destination file during the loop.
- In CopyThread, "run" method is implemented which will run after calling thread.start().

- The program will be terminated when input in console is “exit”.
- Each time program gets input(source file name, destination file name), it will create each instance of thread.
- Threads run individually.

3) How I tested my program

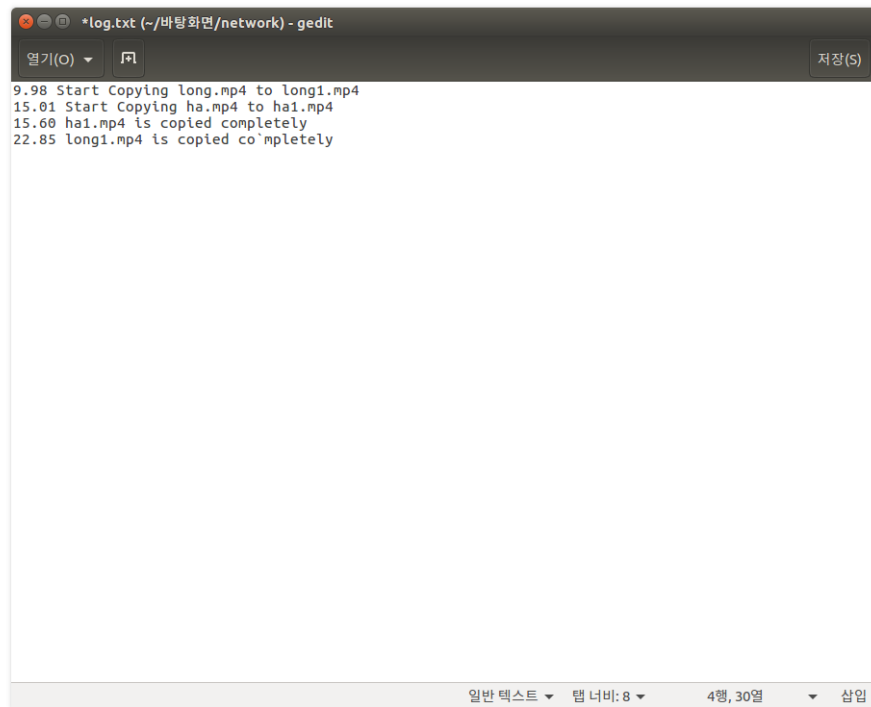


- There are two video files(long.mp4, ha.mp4). long.mp4 has 1.2GB size and ha.mp4 has 57.2MB which is lighter than long.mp4.



- If I run the program with command “python3 2015311901.py” then program will run.
- I can give inputs with command line. The program will run until first input is “exit”.
- We can see there are three outputs. long1.mp4,ha1.mp4 those are the copies of original files. Log.txt is for the logs of program.
- Please run program with command “python3”
- If inputs are text, then command line input does not need “.txt” extension While other object(mp4,

jpg) needs extension.

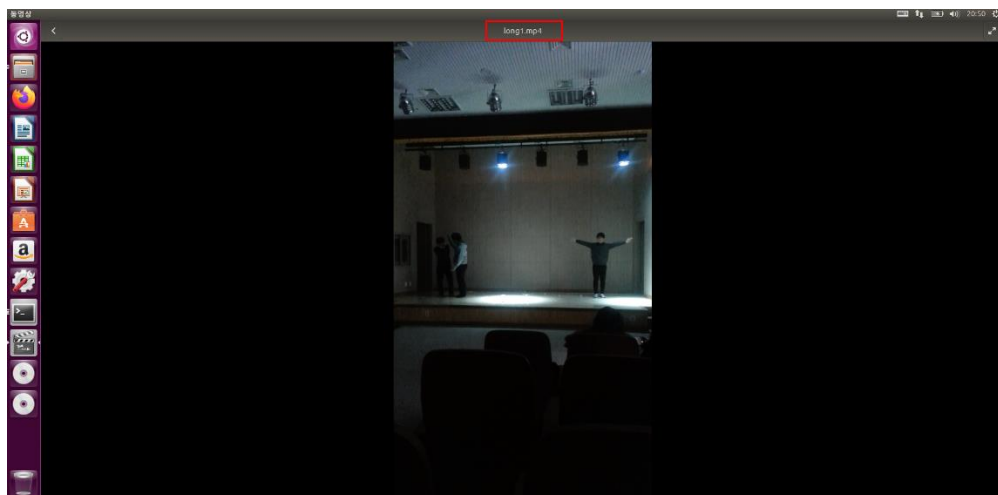


A screenshot of a gedit window titled '*log.txt (~/.바탕화면/network) - gedit'. The window contains the following text:

```
9.98 Start Copying long.mp4 to long1.mp4
15.01 Start Copying ha.mp4 to ha1.mp4
15.60 ha1.mp4 is copied completely
22.85 long1.mp4 is copied co'mpletely
```

The status bar at the bottom indicates '일반 텍스트' (Plain Text), '탭 너비: 8' (Tab width: 8), '4행, 30열' (4 lines, 30 columns), and '삽입' (Insert).

- In the log.txt, though copying long.mp4 process started first, copying ha.mp4 process ended first because long.mp4 is much larger than ha.mp4



- Also, I can see that copied file can be played properly.