

Programming Project 2

CS5352 Advanced Operating Systems Design

Spring 2018

*Napster Style Peer to Peer File Transfer*

*(Test file and Output Cases)*

by

Abhishek Kumar

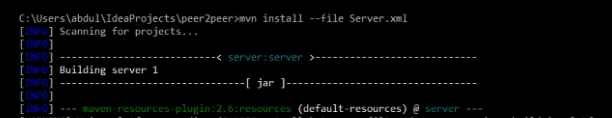
**GitHub Link:**

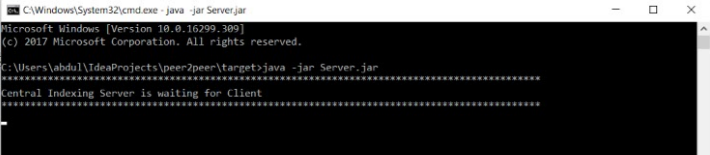
**Building the project (peer2peer\_Abhishek)**

**The following document explains how to build the project and execute the test cases.**

The build tool used is Maven.

Instructions to build and run the p2p system.

1. **Server**
   1. Open command prompt and go to the source code of central indexing server.
   2. > C:\Users\Abhishek\Desktop\Computer Science Texas Tech University\Spring 2018\CS5352 Advanced Operating Systems\Project2\_NapsterP2P\peer2peer\_Abhishek\peer2peerRun **mvn install --file Server.xml**
   3. It will generate target folder in the current directory.
   4. Go to target folder.
      1. > cd target
   5. Run **java -jar Server.jar**
   6. Index server is up and running.



1. **Client** 
   1. Open command prompt and go to the source code of Client.
   2. > C:\Users\Abhishek\Desktop\Computer Science Texas Tech University\Spring 2018\CS5352 Advanced Operating Systems\Project2\_NapsterP2P\peer2peer\_Abhishek\peer2peerRun **mvn install --file Server.xml**
   3. It will generate target folder in the current directory.
   4. Create two folders send and rcv in the **target** folder.
   5. Go to target folder.
      1. **> cd target**
   6. Run **java -jar Client.jar localhost 1** on a new command prompt.
   7. Enter the path of the directory to be registers in the central indexing server

C:\Users\Abhishek\Desktop\Computer Science Texas Tech University\Spring 2018\CS5352 Advanced Operating Systems\Project2\_NapsterP2P\peer2peer\_Abhishek\peer2peer\Peer1

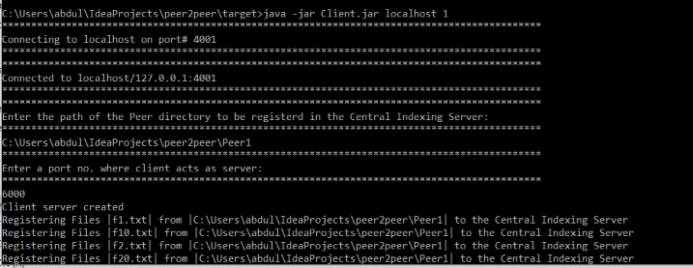
* 1. Enter a random port number where client acts as a server (e.g 6000)
  2. Files get registered in the central indexing server
  3. Run **java -jar Client.jar localhost 2** on a new command prompt.
  4. Enter the path of the directory to be registers in the central indexing server

C:\Users\Abhishek\Desktop\Computer Science Texas Tech University\Spring 2018\CS5352 Advanced Operating Systems\Project2\_NapsterP2P\peer2peer\_Abhishek\peer2peer\Peer2

* 1. Enter a random port number where client acts as a server (e.g 6001)
  2. Files get registered in the central indexing server
  3. Run **java -jar Client.jar localhost 3** on a new command prompt.
  4. Enter the path of the directory to be registers in the central indexing server

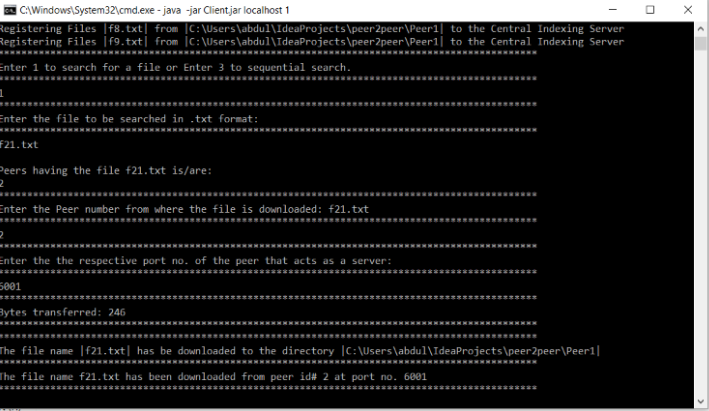
C:\Users\Abhishek\Desktop\Computer Science Texas Tech University\Spring 2018\CS5352 Advanced Operating Systems\Project2\_NapsterP2P\peer2peer\_Abhishek\peer2peer\Peer3

* 1. Enter a random port number where client acts as a server (e.g 6004)
  2. Files get registered in the central indexing server



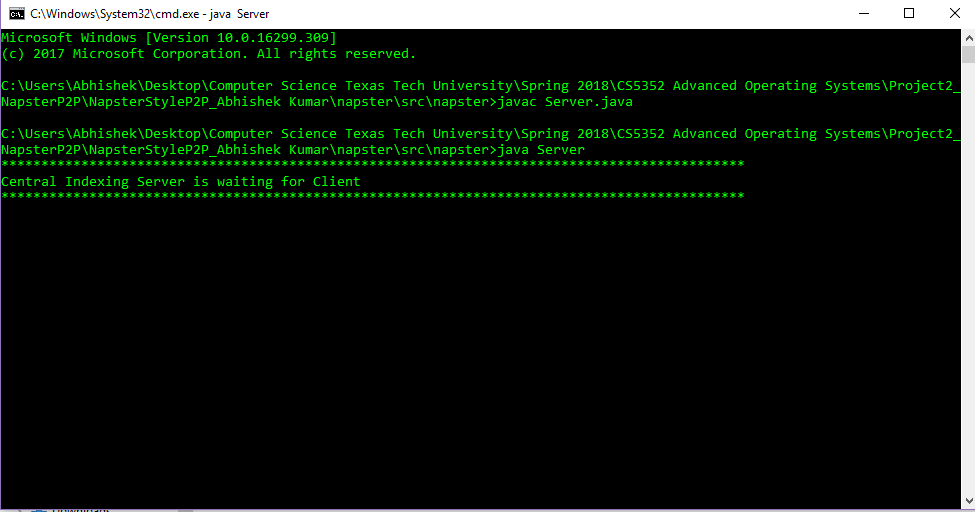
**3. Execution of the program**

1. Enter the option 1 for searching a file (or option 2 for sequential search). *In the below image it has an error showing 3 for sequential search.*
2. Enter the file to be searched in .txt format. (e.g f21.txt). It will display the peers having the following file.
3. Enter the peer number from where the file is supposed to be downloaded. In this example it is 2.
4. Next, enter the port number of the above peer that is acting as a server. In this case port# is 6001.
5. It shows the bytes transferred and displays the successful message of the file being downloaded.

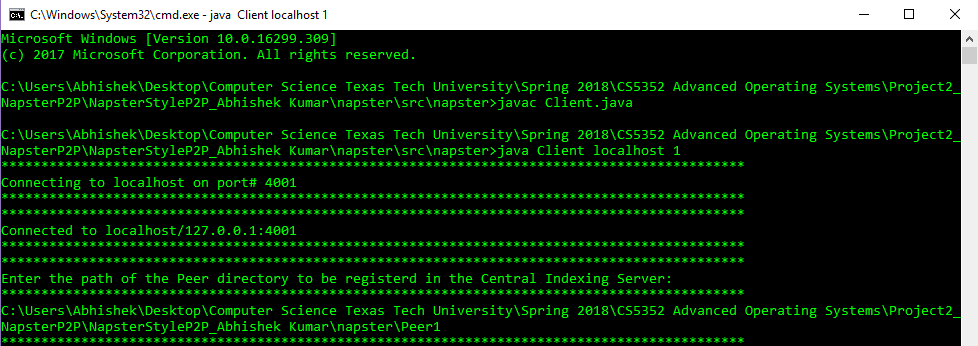
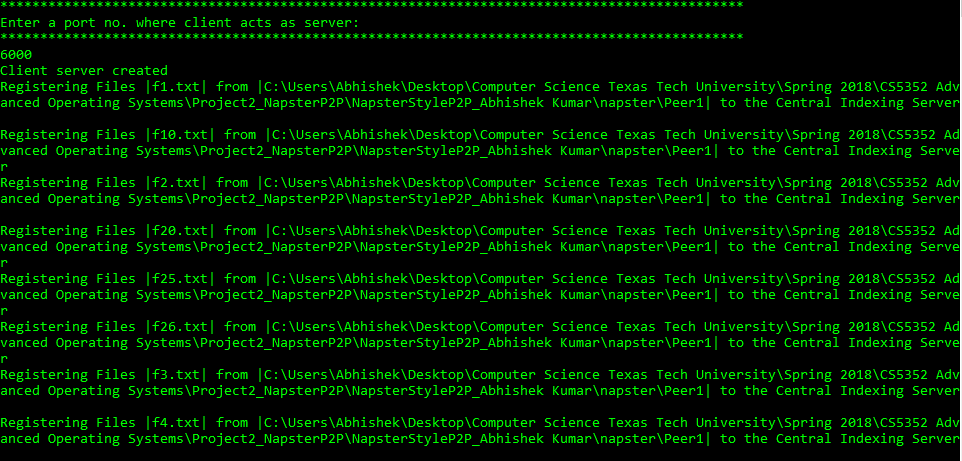


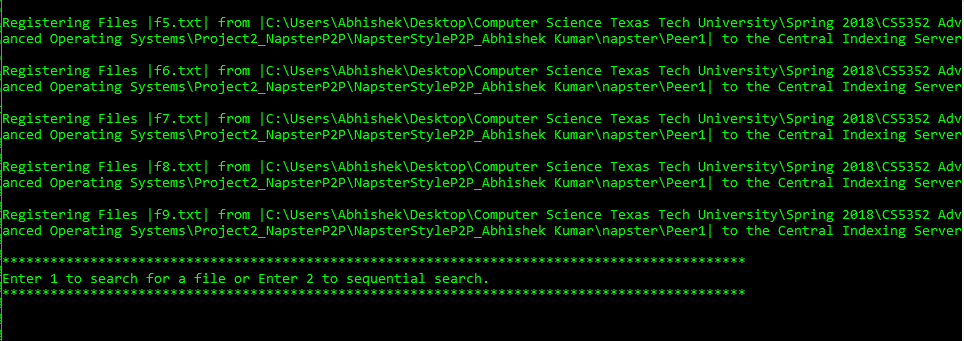
**Building the project (NapsterStyleP2P\_Abhishek Kumar)**

This file has almost a similar code but is slightly tweaked in order to run on the command prompt. The screenshots below clearly depict peer-to-peer file sharing.

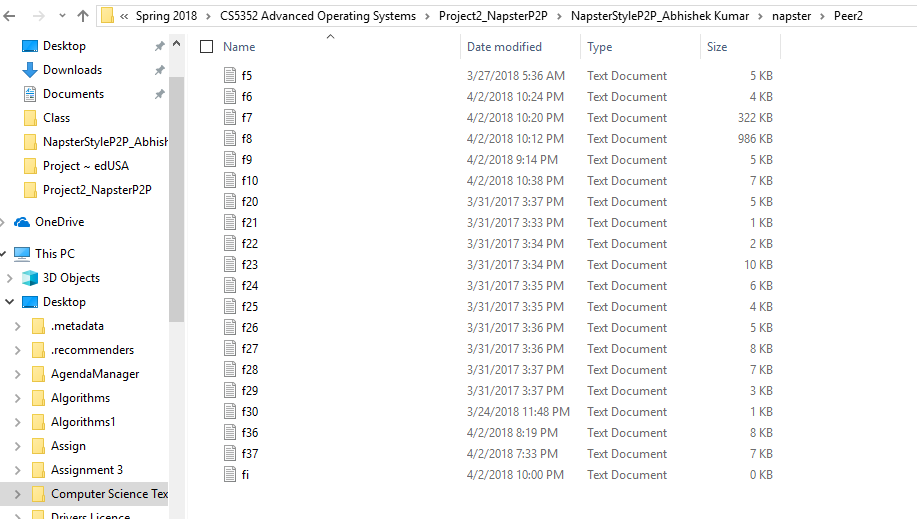


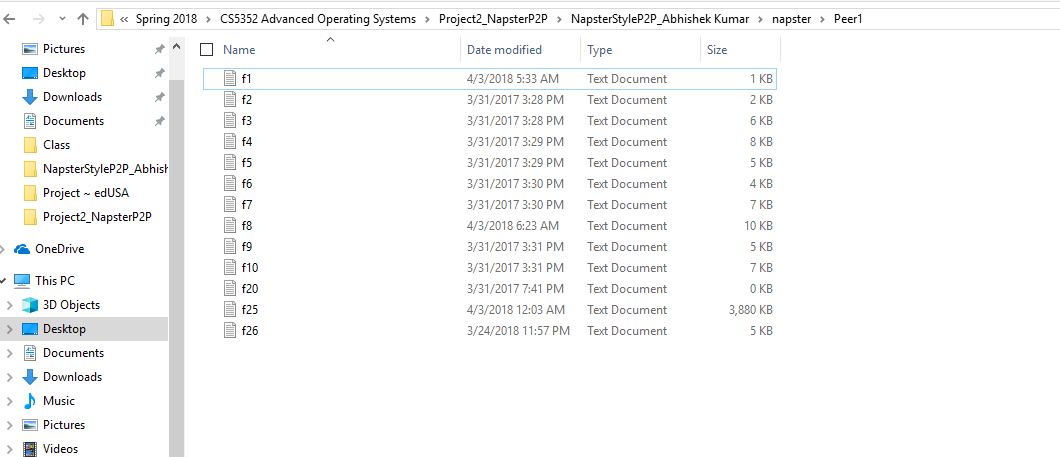
1. Server online

2. Peer 1 registering its files

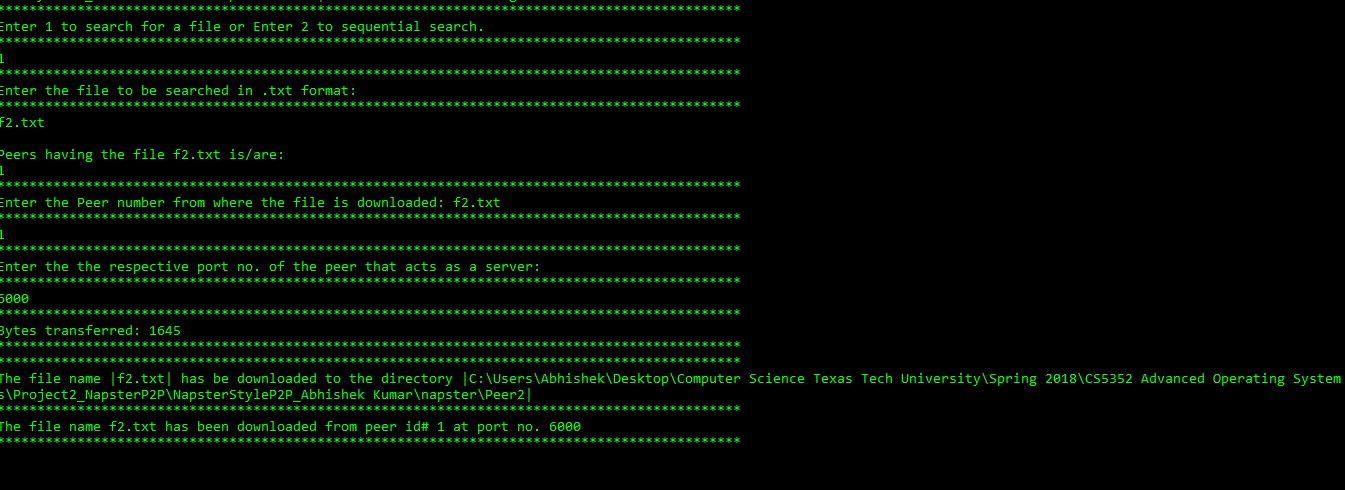


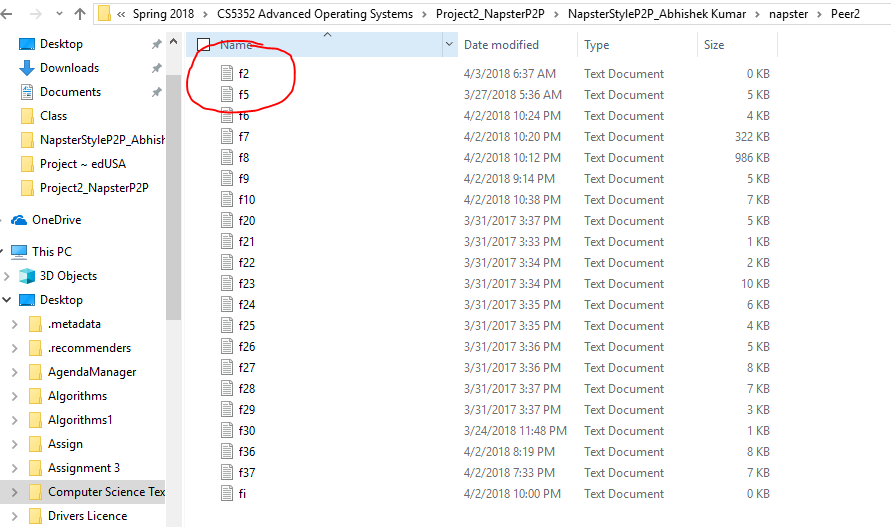
Similarly file of Peer 2 and Peer 3 are registered in the a different command prompt using the argument java Client localhost 2 and java Client localhost 3 (not shown here).

3. Peer 2 directory before downloading file f2.txt from Peer 1



4. Peer 1 directory having file f2.txt.

5. Commands to start a search and later download a file.



7. Peer 2 directory after downloading file f2.txt from Peer 1. Clearly now it has file f2.txt.