Verification

To verify the functionalities of the system, we designed several test cases.

Registry: The Peer uploads its file information (PeerID and its filename list) to Server. We have defined a method (DisplayFileList) to show all the information Server stores. We prepare a folder consisting of 10 files that their size ranges from 1KB to 10KB, and their names are defined as 1K, 2K, … 10K. When this Peer finished registry, through invoking display file information in Server, we can see the terminal shows: XXXXX (port number) 1K 2K 3K 4K 5K 6K 7K 8K 9K 10K. Then registry ran successfully.

Search: A Peer can directly invoke the method 'search(filename)' to inquire the information of a specific file. We have three Peers, and each Peer has its own folder. A file named '2K' exists in all these three folders. When calling search(2K), we get the result below: XXXXX-XXXXY-XXXXZ

What the terminal display is three port numbers.

Obtain: If a Peer wants to obtain a specific file, firstly, it needs to search this file, but this step has been emerged into obtain method. We call obtain(2K) and get three results: XXXXX-XXXXY-XXXXZ. Then we choose XXXXZ, '2K' was downloaded from Peer XXXXZ. The system displays "Download succeed!".

Add, Delete or Modify: In order to verify the automatic update mechanism, user operated these three actions in the folder. We create a new file named '11K' in the folder we used before. Then the terminal display： XXXXX 1K 2K 3K 4K 5K 6K 7K 8K 9K 10K 11K.

We try to delete the file '1K', displaying: XXXXX 2K 3K 4K 5K 6K 7K 8K 9K 10K 11K.

And we modify the name of file '3K' to 'ThreeK', then the terminal shows: XXXXX 2K ThreeK 4K 5K 6K 7K 8K 9K 10K 11K.