For Verification purposes we considered the following scenarios and tested our code on those scenarios:

1. When files are replicated at multiple peers

- Here you can see that we searched for file P0-1KB and we are able to track that this file exists in two different peers. Also we were able to download it to the last peer (count of peer = last peer).

```
Enter the fileName you want to download
P0-1KB
Select Peer from which you want to download file.
Directory logging thread is getting started for peer id: rmi://localhost:3000/peerServer-8940
1. rmi://localhost:3000/peerServer-8884
2. rmi://localhost:3000/peerServer-3428
2
Retrieving file P0-1KB from peer rmi://localhost:3000/peerServer-3428'. You'll be notified when it finishes.
File retrieval started from peerId: rmi://localhost:3000/peerServer-3428 | Time:1645411470043
com.models.Peer rmi://localhost:3000/peerServer-8940 is asking to get the file info of P0-1KB
Creating a New File | Directory: ./shared/2 | FileName : P0-1KB
Change detected in the shared directory. | Shared Directory : ./shared/2
Reading the shared directory: ./shared/2
```

2. When files are only at one peer

```
Enter the fileName you want to download
P0-6KB
Retrieving file P0-6KB from peer rmi://localhost:3000/peerServer-175'. You'll be notified when it finishes.
Directory logging thread is getting started for peer id: rmi://localhost:3000/peerServer-3500
File retrieval started from peerId: rmi://localhost:3000/peerServer-175 | Time:1645417359781
com.models.Peer rmi://localhost:3000/peerServer-175 is asking to get the file info of P0-6KB
Creating a New File | Directory: ./shared/1 | FileName : P0-6KB
```

3. Peers registry and de-registry

```
prashant@prashant-VirtualBox: ~/Documents/PeerToPeerFileSharingSystem-master/scripts
registry method invoked
File got register | FileName: P1-6KB
File got register | FileName: P1-9KB
File got register | FileName: P0-1KB
File got register | FileName: P1-1KB
File got register | FileName: P0-3KB
File got register | FileName: P1-3KB
File got register | FileName: P1-8KB
File got register | FileName: P0-2KB
File got register | FileName: P1-2KB
File got register | FileName: P1-7KB
File got register | FileName: P0-5KB
File got register | FileName: P1-5KB
File got register | FileName: P1-4KB
File got register | FileName: P0-4KB
File got register | FileName: P1-10KB
Reading the shared directory: ./shared/2
Sharing : P2-4KB
Sharing : P2-10KB
Sharing : P2-5KB
Sharing : P2-7KB
Sharing : P2-2KB
Sharing : P2-9KB
Sharing : P2-8KB
Sharing : P2-3KB
Sharing : P2-1KB
Sharing : P2-6KB
registry method invoked
File got register | FileName: P2-4KB
File got register | FileName: P2-10KB
File got register | FileName: P2-5KB
File got register | FileName: P2-7KB
File got register | FileName: P2-2KB
File got register | FileName: P2-9KB
File got register | FileName: P2-8KB
File got register | FileName: P2-3KB
File got register | FileName: P2-1KB
File got register | FileName: P2-6KB
```

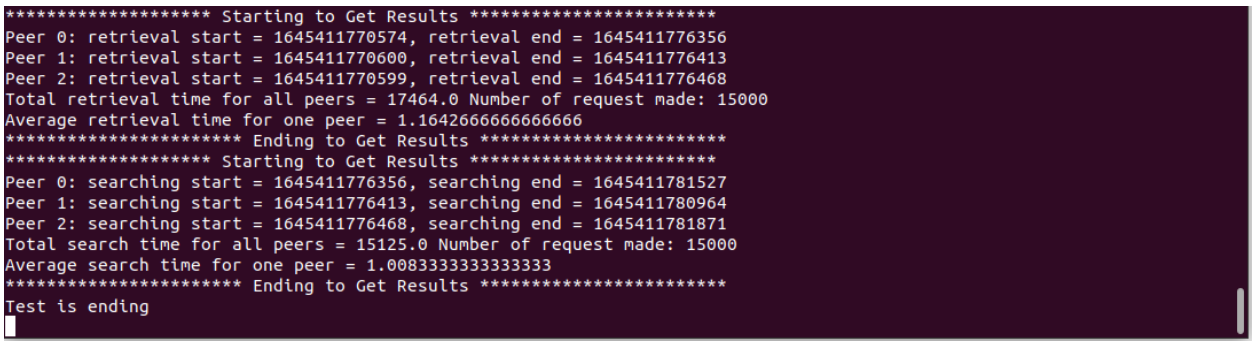
```
DeRegistry Started for PeerId: rmi://localhost:3000/peerServer-9219
File got deregister | FileName: P0-6KB
File got deregister | FileName: P0-9KB
File got deregister | FileName: P0-1KB
File got deregister | FileName: P0-3KB
File got deregister | FileName: P0-10KB
File got deregister | FileName: P0-2KB
File got deregister | FileName: P0-5KB
File got deregister | FileName: P0-7KB
File got deregister | FileName: P0-8KB
File got deregister | FileName: P0-4KB
DeRegistry Started for PeerId: rmi://localhost:3000/peerServer-3248
File got deregister | FileName: P1-6KB
File got deregister | FileName: P1-9KB
File got deregister | FileName: P0-1KB
File got deregister | FileName: P1-1KB
File got deregister | FileName: P0-3KB
File got deregister | FileName: P1-3KB
File got deregister | FileName: P1-8KB
File got deregister | FileName: P0-2KB
File got deregister | FileName: P1-2KB
File got deregister | FileName: P1-7KB
File got deregister | FileName: P0-5KB
File got deregister | FileName: P1-5KB
File got deregister | FileName: P1-4KB
File got deregister | FileName: P0-4KB
File got deregister | FileName: P1-10KB
DeRegistry Started for PeerId: rmi://localhost:3000/peerServer-9884
File got deregister | FileName: P2-4KB
File got deregister | FileName: P2-10KB
File got deregister | FileName: P2-5KB
File got deregister | FileName: P2-7KB
File got deregister | FileName: P2-2KB
File got deregister | FileName: P2-9KB
File got deregister | FileName: P2-8KB
File got deregister | FileName: P2-3KB
File got deregister | FileName: P2-1KB
File got deregister | FileName: P2-6KB
```

4. Sequential Search Request

- Here we ran multiple peers one after another and saw that directory watcher was able to detect the changes while retrieving files from one peer to another.
- Also while doing search our program was giving correct peers which had the file we were searching for. Can be tested easily in program with PeerServerTest we wrote.

```
Exiting Peer 7656
******************** Starting to Get Results ************************
Peer 0: retrieval start = 1645411681883, retrieval end = 1645411685154
Peer 1: retrieval start = 1645411686281, retrieval end = 1645411688108
Peer 2: retrieval start = 1645411689546, retrieval end = 1645411691155
Total retrieval time for all peers = 6707.0 Number of request made: 52500
Average retrieval time for one peer = 0.12775238095238095
********************** Ending to Get Results ************************
******************** Starting to Get Results ************************
Peer 0: searching start = 1645411685155, searching end = 1645411686265
Peer 1: searching start = 1645411688108, searching end = 1645411689542
Peer 2: searching start = 1645411691155, searching end = 1645411693161
Total search time for all peers = 4550.0 Number of request made: 52500
Average search time for one peer = 0.08666666666666667
********************** Ending to Get Results ************************
Test is ending
```

5. Parallel Search Requests & Synchronization
    - To see the multiple requests were hitting concurrently we used the command line & system.out.println to output which peer is calling for which request.
    - Multiple peers were hitting the same search or retrieve functions together.
    - Program is also not giving concurrency exceptions, which also gave surety that program is running fine and is synchronized.



Program covers all the cases for sequential and parallel search. It also expects the user to give the commands according to the needs. If a user tries to give wrong stdin commands then the program can crash. For example, while asking for integer input, giving string will result in program crash. So following the command as given after executing the program is very necessary.