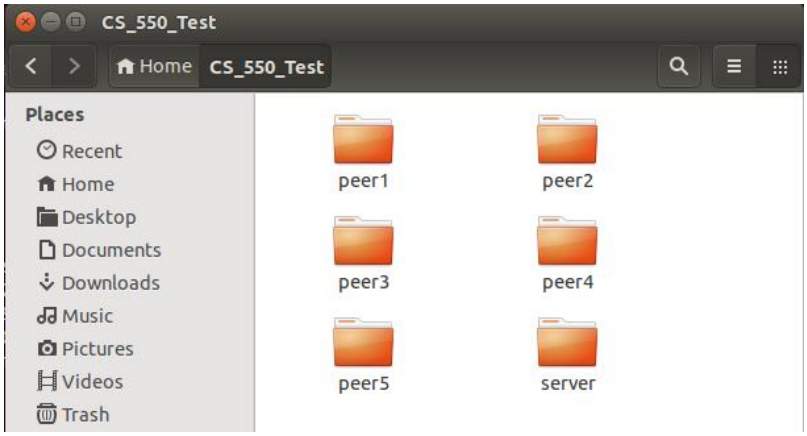
Performance Results

1. Test Environment and Nodes Deployment

In this performance test, we deploy 5 peers and 1 index server and test the performance of this program on the Linux operating system.



1.1 Test Parameters and Design

In the first, register 10 files on the index server for testing.

In the second, allow one peer to search these 10 files one by one. Each file is searched 200 times. Totally 2000 times search requests are made and the response time is recorded. The situation of each peer to search these 10 files one by one is to eliminate error of file location in the array list on the index server. In another word, maybe the file 1.txt is located on the first place of the array list, so that to search this file may take less time than the last one in the array list.

In the third, test the program performance under the situation of 1 peer, or multiple peers make search requests at the same time, evaluating the concurrent requests.

2. Average Response Time of One Peer

2.1 Start Server

Start Index server and register 10 files for searching request. For one peer, we will calculate the response time of searching 10 files one by one and each for 200 times. Record the start time and stop time.

```
tayloryy@Tayloryy: ~/server
tayloryy@Tayloryy:~/server$ java -jar server.jar
Start Server
File:3.txt from Client:127.0.1.1:9020 is registried!
File:9.txt from Client:127.0.1.1:9020 is registried!
File:7.txt from Client:127.0.1.1:9020 is registried!
File:10.txt from Client:127.0.1.1:9020 is registried!
File:5.txt from Client:127.0.1.1:9020 is registried!
File:1.txt from Client:127.0.1.1:9020 is registried!
File:8.txt from Client:127.0.1.1:9020 is registried!
File:6.txt from Client:127.0.1.1:9020 is registried!
File:2.txt from Client:127.0.1.1:9020 is registried!
File:4.txt from Client:127.0.1.1:9020 is registried!
```

2.2 Search Response

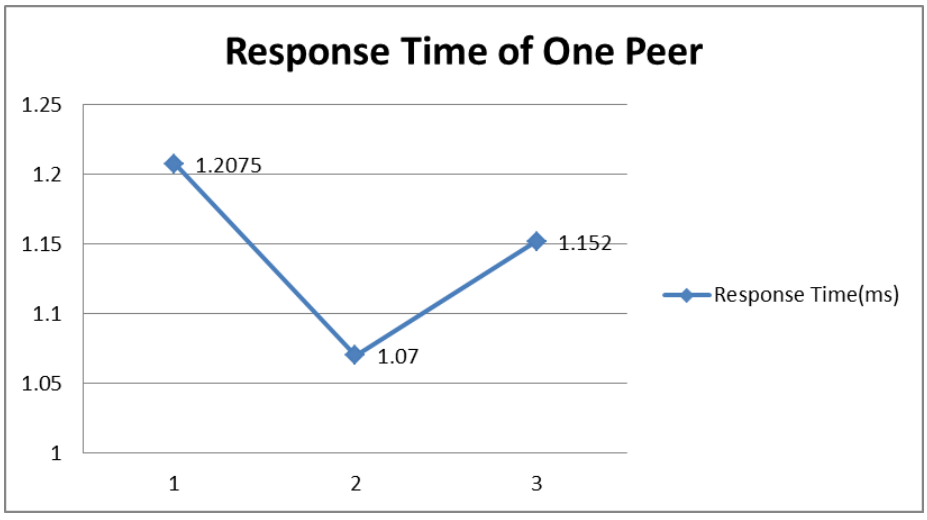
From the following figure, we can see the average response time of one peer is about 1.2075ms.

```
tayloryy@Tayloryy: ~/peer1
tayloryy@Tayloryy:~/peer1$ java -jar peer1.jar
Client server 1 started!

1 Register a file
2 Search a file
3 Exit
2
Start 2000 times search request test!
Start time: 1411919949460
End time: 1411919951875
Total time of 2000 times search request is: 2415ms
Average sponse time: 1.2075ms
```

2.3 Evaluation

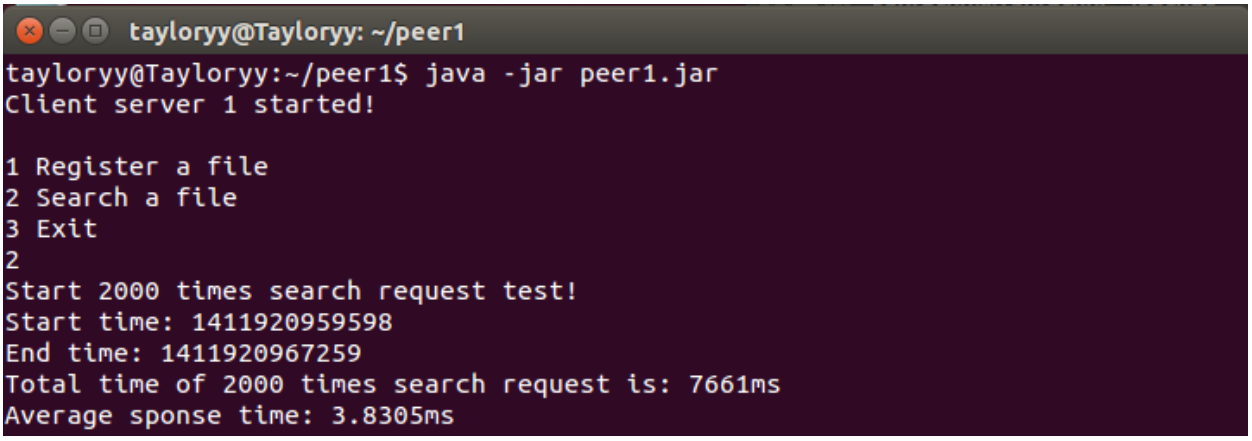
We test the research request of one peer for 3 times, its response times are showed on the following figure. The average response for one peer is about 1.14 ms/response.

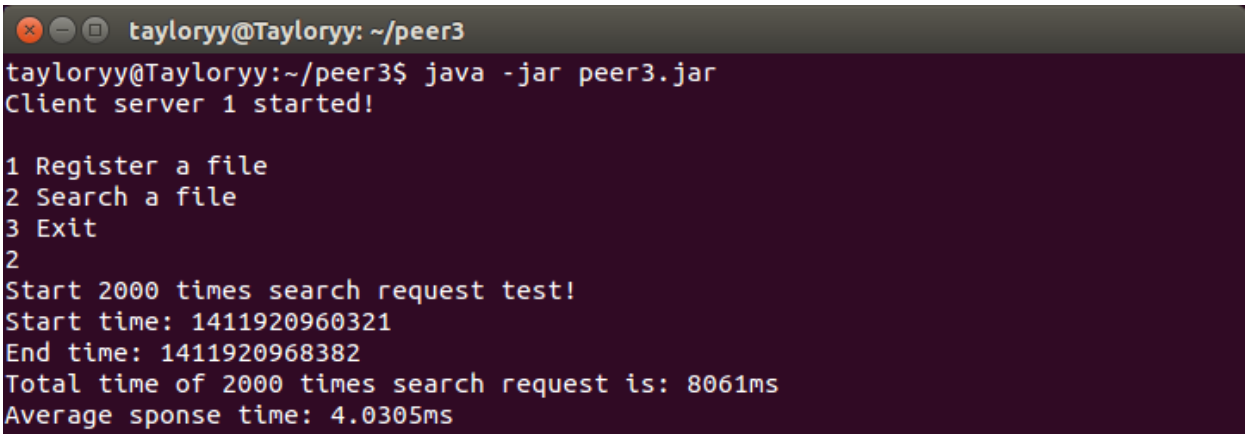


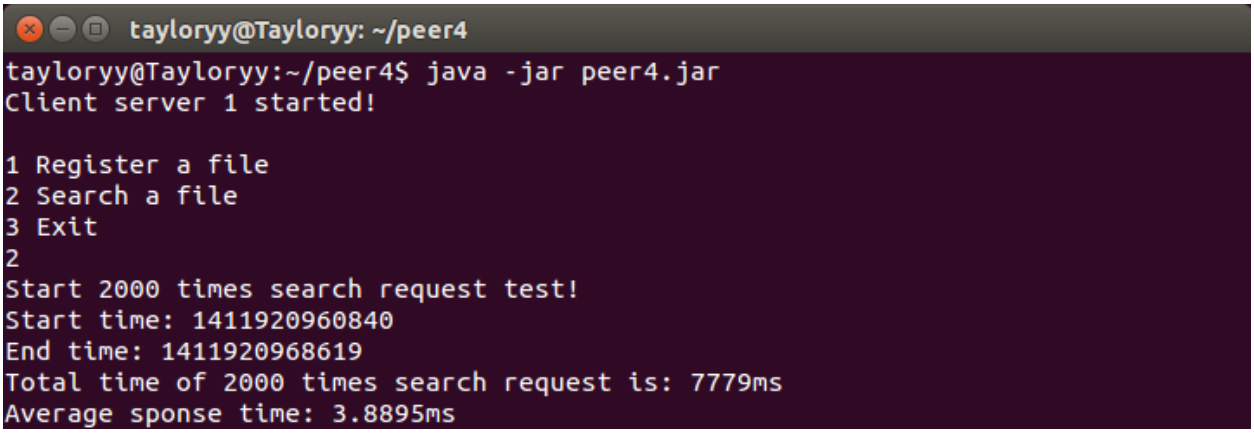
3. Concurrent Response Time

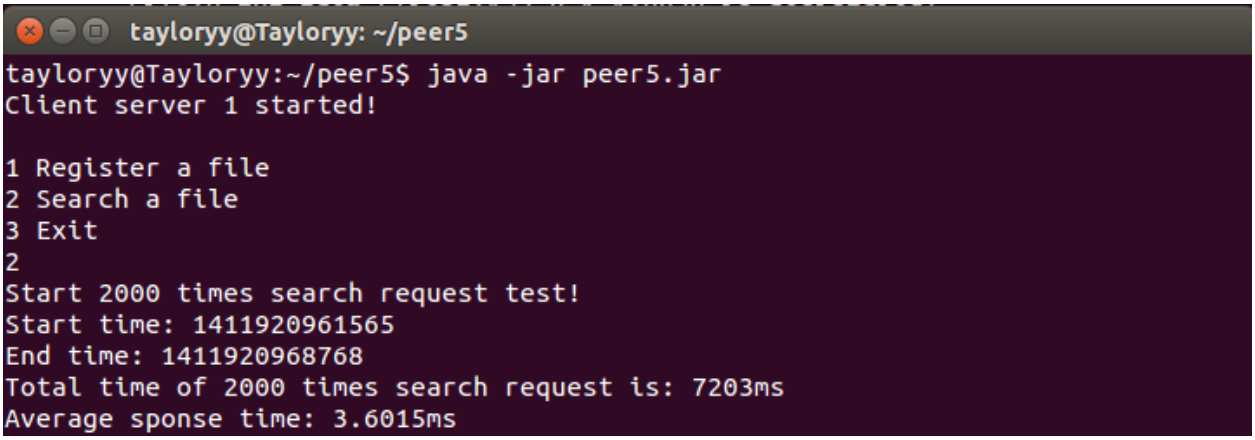
In this part, we run our test for several situations, such as 2 peers concurrent, 3 peers concurrent and 4 peers concurrent. In the following, concurrent tests data of all the situations listed above will be showed through tables and the results will be analyzed according to figures. However, briefly, just the screenshot of the running of 4 peers situation is showed through figures.

3.1 Four Peers Concurrent









3.2 Testing Results for All situations have been Tested

Table 1. 1 Peer Response Test

	Average Response Time (ms)	Start Time	End Time	Total Time (ms)
Peer_1	1.2075	1411919949460	1411919951875	2415

Table 2. 2 Peers Response Test

	Average Response Time (ms)	Start Time	End Time	Total Time (ms)
Peer_1	2.819	1411919613744	1411919619382	5638
Peer_3	2.5205	1411919614651	1411919619692	5041

Table 3. 3 Peers Response Test

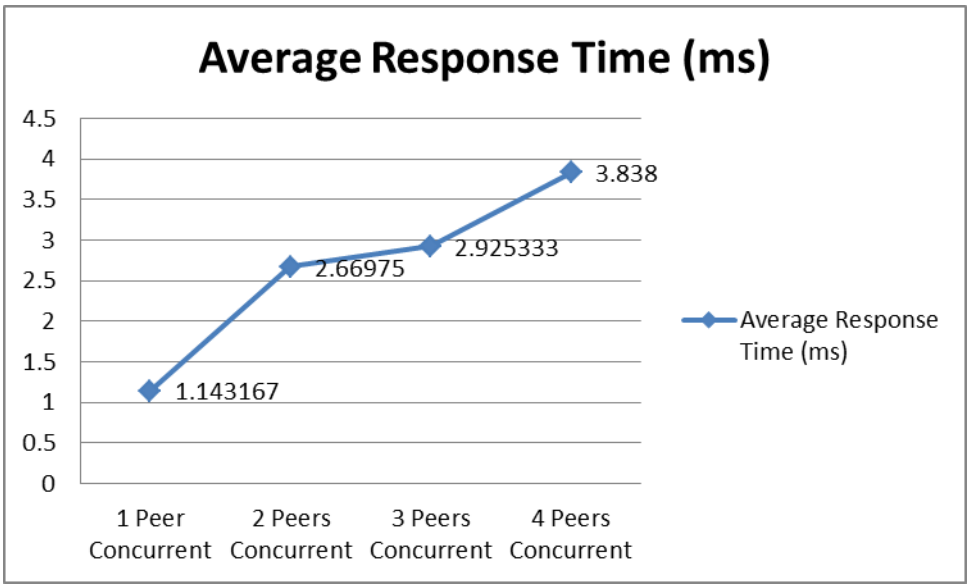
	Average Response Time (ms)	Start Time	End Time	Total Time (ms)
Peer_1	2.955	1411920487211	1411920493121	5910
Peer_3	3.0285	1411920487725	1411920493782	6057
Peer_4	2.7925	1411920488324	1411920493909	5585

Table 4. 4 Peers Response Test

	Average Response Time (ms)	Start Time	End Time	Total Time (ms)
Peer_1	3.8305	1411920959598	1411920967259	7661
Peer_3	4.0305	1411920960321	1411920968382	8061
Peer_4	3.8895	1411920960840	1411920968619	7779
Peer_5	3.6015	1411920961565	1411920968768	7203

3.3 Analysis and Conclusions

The following figure shows the average response times of 1 peer, 2 peers, 3 peers and 4 peers concurrent situations.



From this figure, we can see that the more peers concurrently make search requests to the index server, the more time one request response time will be. In another word, concurrent will increase the search time or decrease the efficiency of the program.