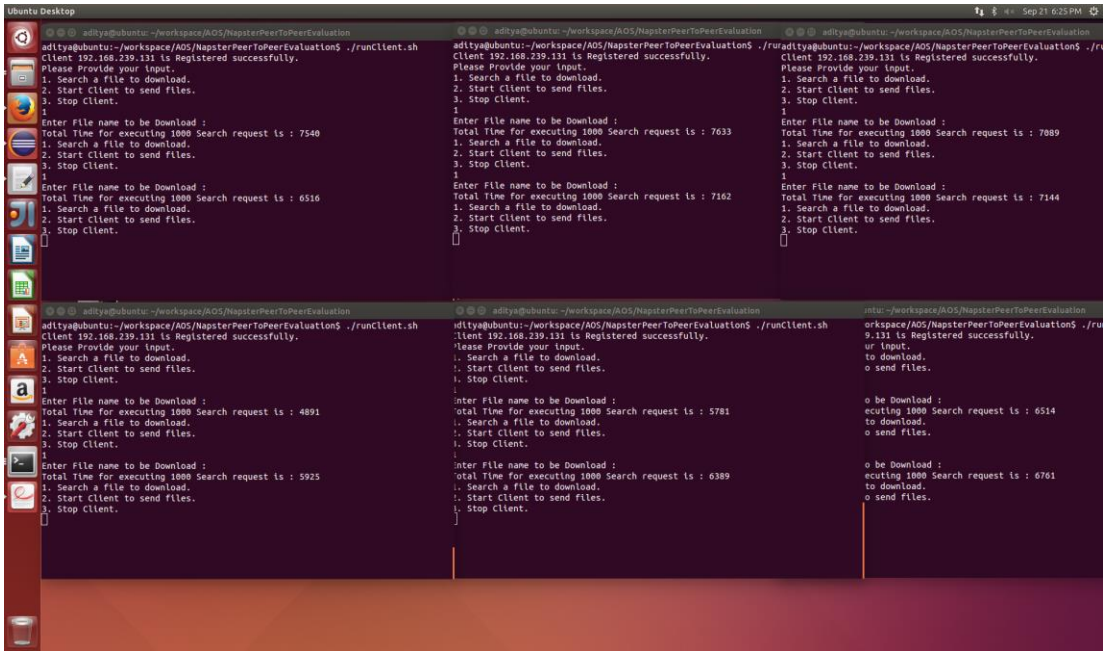
Programming Assignment 1

Napster Style Peer to Peer.

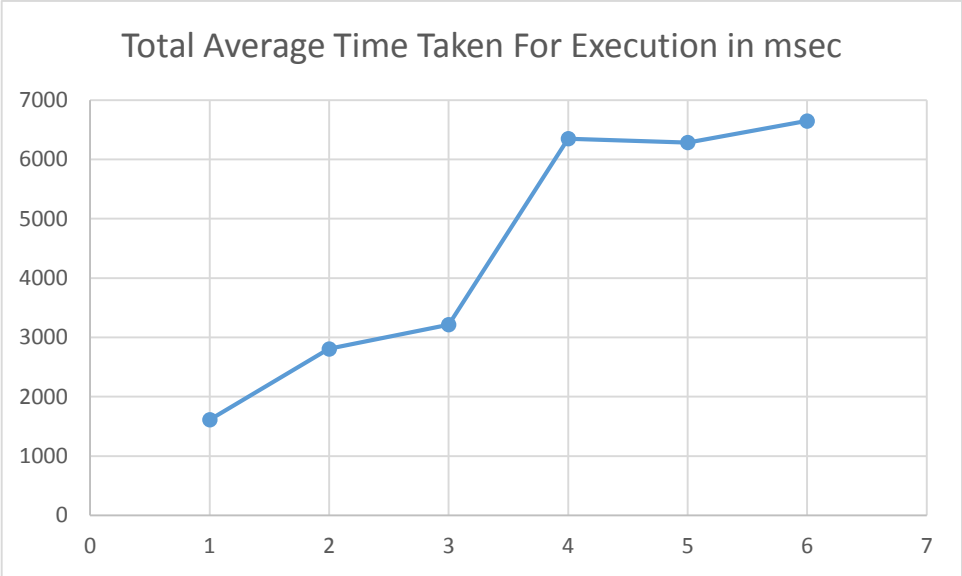
Performance Evaluation 1: By increasing number of clients for searching.

- Performance of a system is measured by generating **1000** sequential **search request** from a single client to an Indexing Server. And then varying the number of clients for concurrent search requests.
- First a single client is run to make 1000 search request with a particular file name in an Indexing Server.
- Then number of clients are increased one by one and their average time for execution of 1000 threads are noted.
- Following results are observed for multiple clients and their multiple concurrent requests.

Number of Clients	Average Time Taken For Execution of 1000 request in milliseconds
1	1613ms
2	2807ms
3	3215ms
4	6348ms
5	6283ms
6	6649ms

- 

- A line graph is plotted using the above observations.



X-axis : Number of Clients.

Y-axis : Average time taken to process 1000 requests per Client.

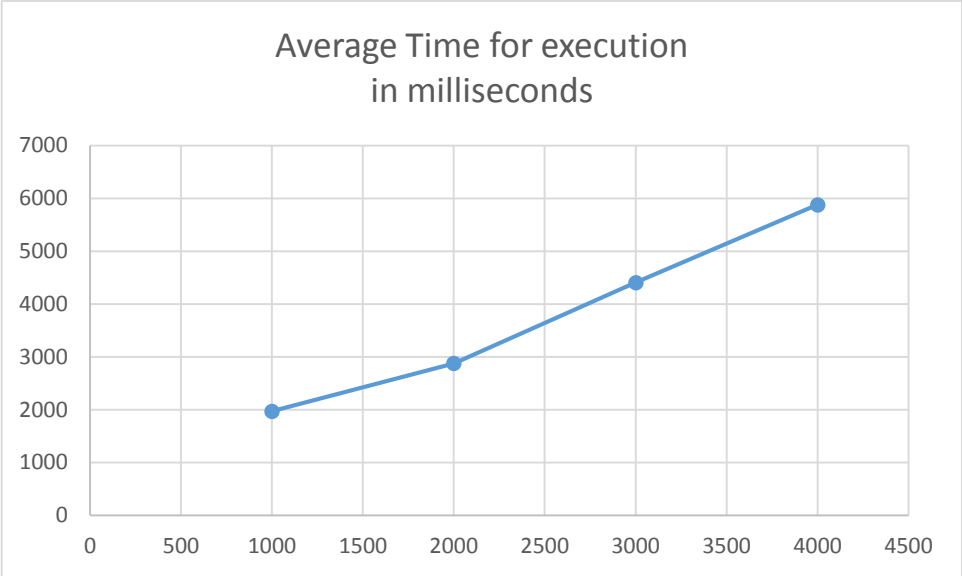
- **Observation**:
  - As number of thread increase (multiple clients) there come some overhead and because of that average execution time increases.
  - But even if we increase the request count by 1000, execution time is not increased much in many scenarios.

Performance Evaluation 2: By increasing number of sequential search requests.

- First a **single client** is run to make **1000 search request** with a particular file name to an Indexing Server.
- Then each time the number of request are increased by 1000 and their average time for execution is noted.

Total Number of Threads/Request to Indexing Server	Average Time Taken For Execution in millisecond
1000	**1972ms**
2000	**2877ms**
3000	**4407ms**
4000	**5880ms**

- A line graph is plotted using the above observations.



X-axis : Number of Request to an Indexing Server.

Y-axis : Average time taken to process all requests.

- **Observation**:
  - As number of thread increases average execution time increases.

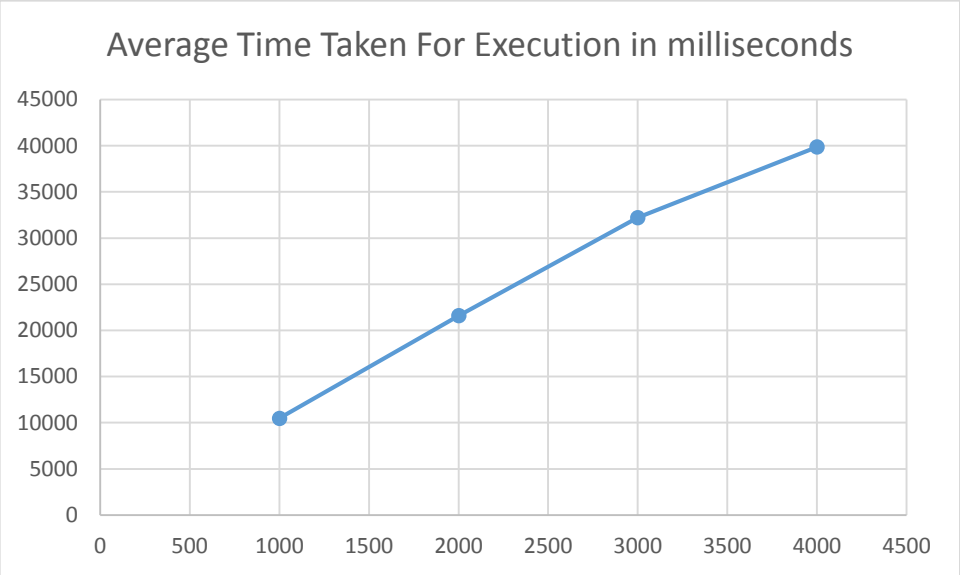
Performance Evaluation 3: By increasing number of registration requests.

- First a **single client** is run to make **1000 Registration request** to an Indexing Server.
- Then each time the number of request are increased by 1000 and their average time for execution is noted.

Total Number of Threads/Request to Indexing Server	Average Time Taken For Execution in milliseconds
1000	**10471ms**
2000	**21587ms**
3000	**32216ms**
4000	**39870ms**

```
aditya@ubuntu: ~/workspace/AOS/NapsterPeerToPeerEvaluation
aditya@ubuntu:~/workspace/AOS/NapsterPeerToPeerEvaluation$ ./runClient.sh
Total Time for executing 1000 Register request is : 10471
Client 192.168.239.131 is Registered successfully.
Please Provide your input.
1. Search a file to download.
2. Start Client to send files.
3. Stop Client.
^C
aditya@ubuntu:~/workspace/AOS/NapsterPeerToPeerEvaluation$ ./runClient.sh
Total Time for executing 2000 Register request is : 21587
Client 192.168.239.131 is Registered successfully.
Please Provide your input.
1. Search a file to download.
2. Start Client to send files.
3. Stop Client.
^C
aditya@ubuntu:~/workspace/AOS/NapsterPeerToPeerEvaluation$ ./runClient.sh
Total Time for executing 3000 Register request is : 32316
Client 192.168.239.131 is Registered successfully.
Please Provide your input.
1. Search a file to download.
2. Start Client to send files.
3. Stop Client.
```

- A line graph is plotted using the above observations.



X-axis : Number of Request to an Indexing Server.

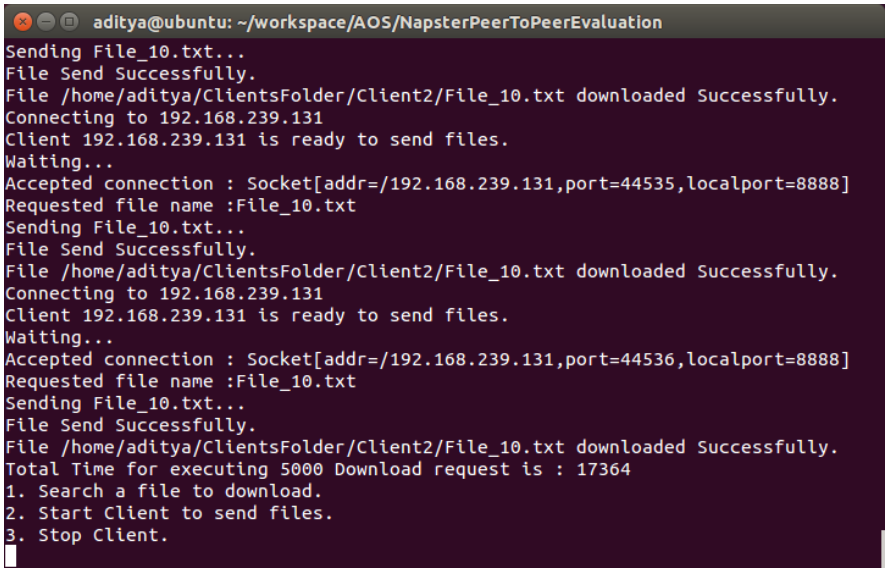
Y-axis : Average time taken to process all requests.

- **Observation**:
  - As number of registration increases average execution time increases linearly.
  - First it is increased by a significant amount, but when request count increases from 3000 to 4000 time increases by 7 seconds.

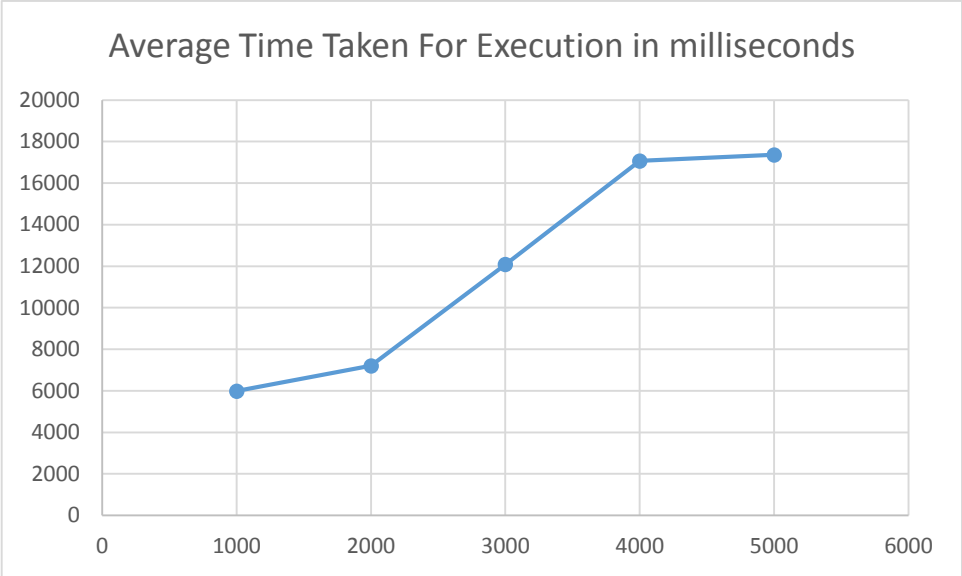
Performance Evaluation 4: By increasing number of Download requests.

- First a **single client** is run to make **1000 Download request** to a peer having that particular file.
- Then each time the number of request are increased by 1000 and their average time for execution is noted.

Total Number of Threads/Request to another client for downloading a file	Average Time Taken For Execution in milliseconds
1000	**5983ms**
2000	**7203ms**
3000	**12085ms**
4000	**17064ms**
5000	**17364ms**



- A line graph is plotted using the above observations.



X-axis : Number of Download Request to another Peer.

Y-axis : Average time taken to process all requests.

- **Observation**:
  - As number of download request increases average execution time increases.
  - First it is not increased by a significant amount, but when request count increases from 2000 to 3000 and 3000 to 4000 time increases by 4 seconds.