Illinois Institute of Technology

Advanced Operating System (CS-550)

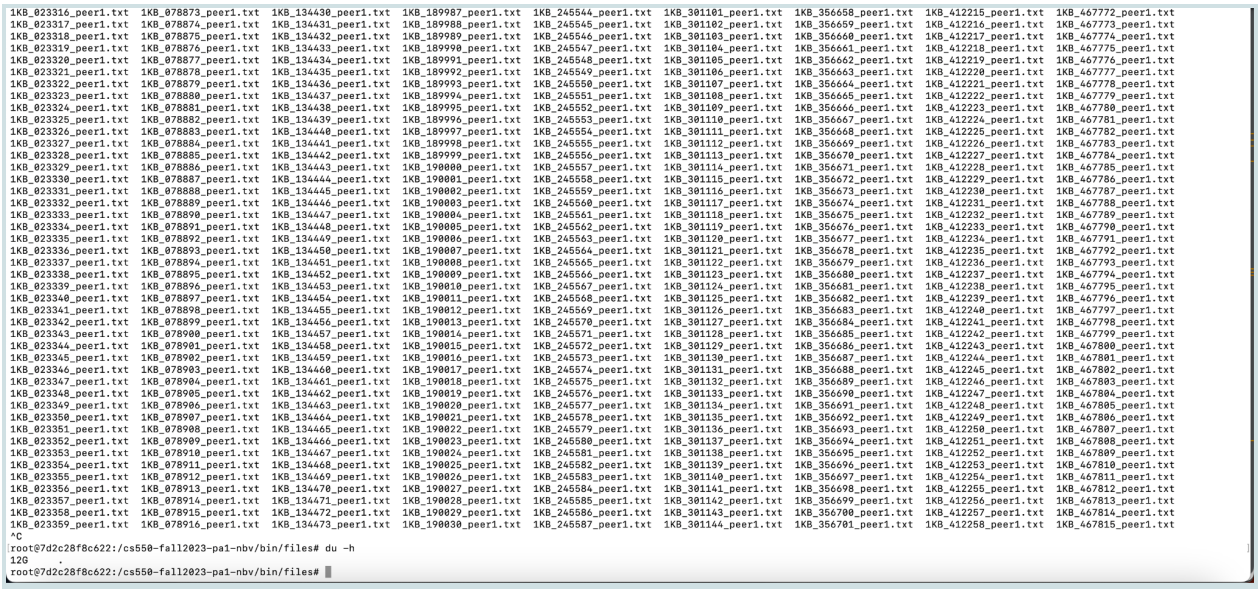
PA-1

Submitted By:

Batkhishig Dulamsurankhor (#A20543498)
Nitin Singh (#A20516824)
Vaishnavi Papudesi Babu (#A20527963)

Instructor: Professor Ioan Raicu
TA: Mr. Lan Nguyen
TA: Ms. Sonal Gaikwad
TA: Mr. Adarsh Agrawal

Filenames in each peer are in the format -
1KB_<7 digit integer ranging from 0 to 500000>_peer1.txt
Or 1MB_<7 digit integer ranging from 0 to 500000>_peer1.txt
Or 1GB_<7 digit integer ranging from 0 to 500000>_peer1.bin



Register files -

```
Peer started!
[Server]: Waiting for a peer ...
[Client]: What do you want to do?
        [0] - Register
        [1] - Search a file
        [2] - Obtain a file
0
[Client]: Connected!
[Client]: Peer ID:peer1
[Client]: Updated peer files!
[Client]: What do you want to do?
        [0] - Register
        [1] - Search a file
        [2] - Obtain a file
1
```

CIS screenshot-

1KB_056492_peer1.txt
1KB_461688_peer1.txt
1KB_492521_peer1.txt
1KB_075774_peer1.txt
1KB_317623_peer1.txt
1KB_416917_peer1.txt
1KB_360183_peer1.txt
1KB_480946_peer1.txt
1KB_273317_peer1.txt
1KB_299009_peer1.txt
1KB_324866_peer1.txt
1KB_440553_peer1.txt
1KB_129622_peer1.txt
1KB_413501_peer1.txt
1KB_413539_peer1.txt
1KB_344764_peer1.txt
1KB_211007_peer1.txt
1KB_362162_peer1.txt
1KB_280291_peer1.txt
1KB_429237_peer1.txt
1KB_441600_peer1.txt
1KB_306787_peer1.txt
1KB_243917_peer1.txt
1KB_082592_peer1.txt
1KB_471355_peer1.txt
1KB_408880_peer1.txt
1KB_140317_peer1.txt
1KB_088106_peer1.txt
1KB_285571_peer1.txt
1KB_104620_peer1.txt
1KB_082220_peer1.txt
1KB_138795_peer1.txt
1KB_147596_peer1.txt
1KB_029247_peer1.txt
1KB_010314_peer1.txt

Search files

```
Peer started!
[Server]: Waiting for a peer ...
[Client]: What do you want to do?
        [0] - Register
        [1] - Search a file
        [2] - Obtain a file
1
[Client]: File name:zl.txt
[Client]: Connected!
[Client]: peer1 peer1.mynetwork
[Client]: What do you want to do?
        [0] - Register
        [1] - Search a file
        [2] - Obtain a file
1
[Client]: File name:1KB_130040_peer1.txt
[Client]: Connected!
[Client]: peer1 peer1.mynetwork
[Client]: What do you want to do?
        [0] - Register
        [1] - Search a file
        [2] - Obtain a file
1
[Client]: File name:
```

Do a weak scaling scalability study to measure search time of 10K requests per peer, on 1 node and 2 nodes. Report the average and standard deviation. Plot your data in figures graphically.

1 node: mean and standard deviation

Standard Deviation, σ: **3.8248576444098**

Count, N: 1000
Sum, Σx: 6092
Mean, μ: 6.092
Variance, σ^2: 14.629536

Steps

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}.$$

$$\sigma^2 = \frac{\Sigma(x_i - \mu)^2}{N}$$

$$= \frac{(5 - 6.092)^2 + \ldots + (124 - 6.092)^2}{1000}$$

$$= \frac{14629.536}{1000}$$

$$= 14.629536$$

$$\sigma = \sqrt{14.629536}$$

$$= 3.8248576444098$$

2 node: mean and standard deviation

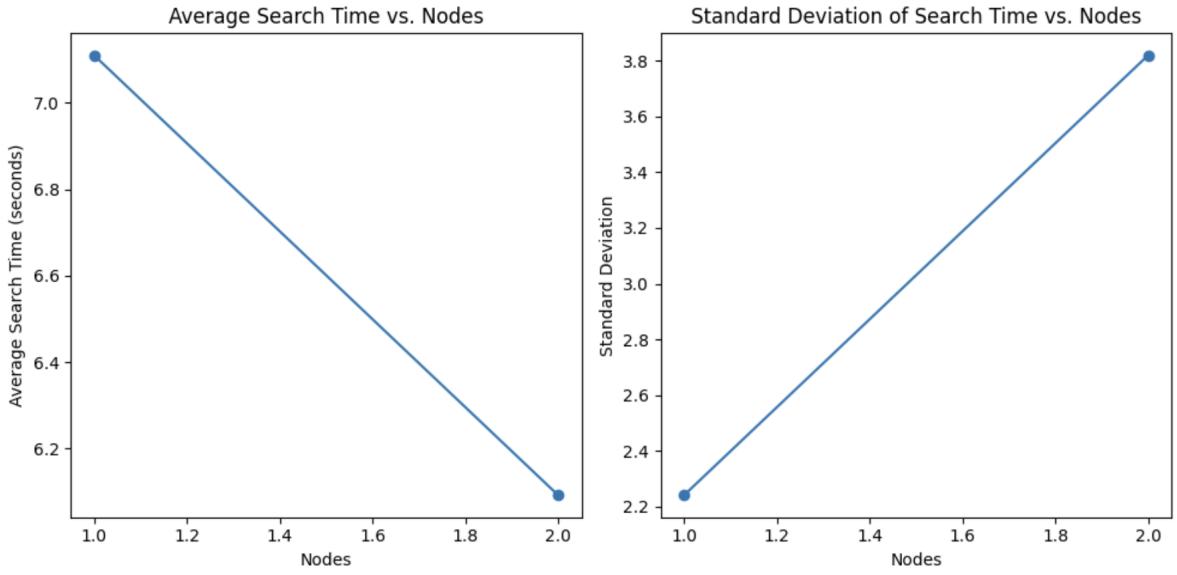
Standard Deviation, σ: **2.2461769743277**

Count, N: 1000
Sum, Σx: 7117
Mean, μ: 7.117
Variance, σ^2: 5.045311

Steps

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}.$$

$$\sigma^2 = \frac{\Sigma(x_i - \mu)^2}{N}$$
$$= \frac{(64 - 7.117)^2 + \ldots + (8 - 7.117)^2}{1000}$$
$$= \frac{5045.311}{1000}$$
$$= 5.045311$$
$$\sigma = \sqrt{5.045311}$$
$$= 2.2461769743277$$



Do a strong scaling scalability study that measures the search and transfer time of 10K small files (1KB), on 1 node and 2 nodes. Repeat the study on 1K medium files (1MB). Repeat the study on 8 large files (1GB). Report the average and standard deviation. Plot your data in figures graphically.

Can you deduce that your P2P centralized system is scalable up to 2 nodes? Does it scale well for some file sizes, but not for others? Based on the data you have so far, what would happen if you had 1K peers with small, medium, and large files? What would happen if you had 1 billion peers?

*i)*Yes, scaling up to two nodes is doable. Scalability, however, is influenced by a variety of elements, such as the system's architecture, design, and implementation.
Every peer (node) in a typical P2P system communicates directly with every other peer on the network. As a result, if your system is designed to run on just two nodes, it can be regarded as scalable because it can effectively manage a relatively small number of nodes.

A p2p system, on the other hand, is typically designed to function with additional p2p nodes to distribute the load and give fault tolerance. Therefore, it might be a little inaccurate to refer to it as "scalable" with only 2 p2p nodes.

ii)The size of the files shared can also affect scalability. We encountered this issue while working on a P2P system. Some p2p systems work well with small files, while others struggle with larger files because of bandwidth and latency limitations.

iii)Managing 1,000 p2p connections can be very resource-intensive, and it would require powerful algorithms to exchange data efficiently.

The performance of the system would depend on the capabilities and design of the system.

Small p2p files can be widely distributed among peers, particularly if the system supports simultaneous downloading and uploading.

Larger p2p files may have issues with bandwidth, storage and latency, but medium-sized p2p files might also work well.

iv)If your P2P system is centralized, one billion peers is a sizable number of peers. You'll need to control connections, monitor your resources, and make sure your data transmission and reception processes are effective. All of this would require a significant amount of infrastructure, bandwidth, and processing power.

At such a size, centralized systems could experience congestion and single points of failure, so you might want to think about decentralization or a hybrid strategy. To effectively manage a large number of peers, you might also want to think about how centralized or decentralized your system is.