

**Illinois Institute of Technology**

**Advanced Operating System**  
**(CS-550)**

**PA-1**

**Submitted By:**

**Batkishig 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**





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\_010214\_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:z1.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,  $\sigma$ : **3.8248576444098**

Count, N: 1000  
Sum,  $\Sigma x$ : 6092  
Mean,  $\mu$ : 6.092  
Variance,  $\sigma^2$ : 14.629536

#### **Steps**

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

$$\begin{aligned}\sigma^2 &= \frac{\sum (x_i - \mu)^2}{N} \\ &= \frac{(5 - 6.092)^2 + \dots + (124 - 6.092)^2}{1000} \\ &= \frac{14629.536}{1000} \\ &= 14.629536 \\ \sigma &= \sqrt{14.629536} \\ &= 3.8248576444098\end{aligned}$$

### 2 node: mean and standard deviation



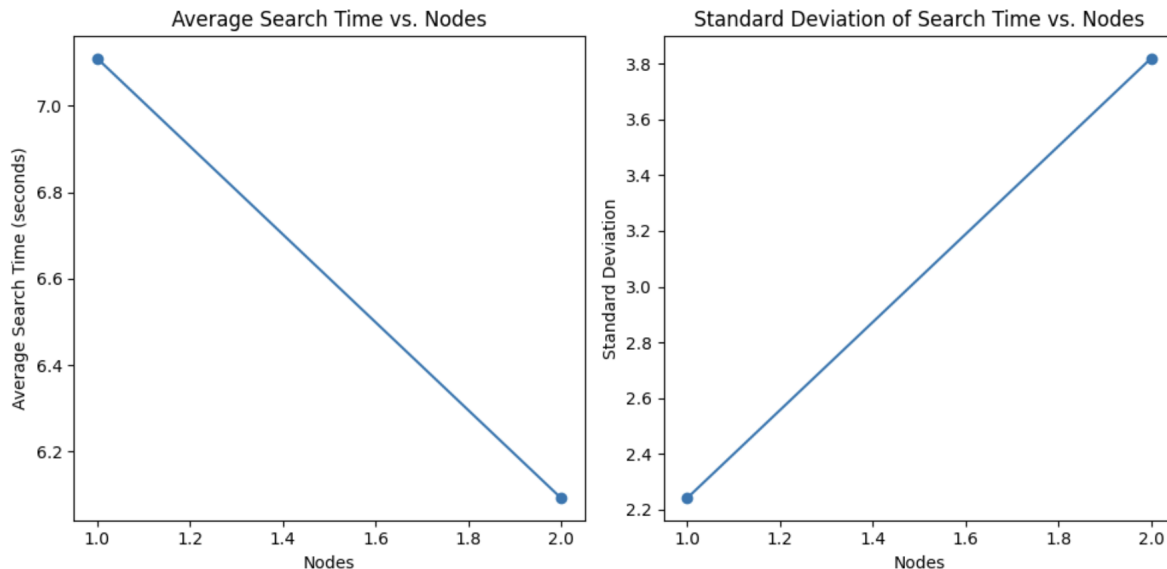
Standard Deviation,  $\sigma$ : **2.2461769743277**

Count, N: 1000  
Sum,  $\Sigma x$ : 7117  
Mean,  $\mu$ : 7.117  
Variance,  $\sigma^2$ : 5.045311

### Steps

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

$$\begin{aligned}\sigma^2 &= \frac{\sum (x_i - \mu)^2}{N} \\ &= \frac{(64 - 7.117)^2 + \dots + (8 - 7.117)^2}{1000} \\ &= \frac{5045.311}{1000} \\ &= 5.045311 \\ \sigma &= \sqrt{5.045311} \\ &= 2.2461769743277\end{aligned}$$



*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.*



**For 10K File:**

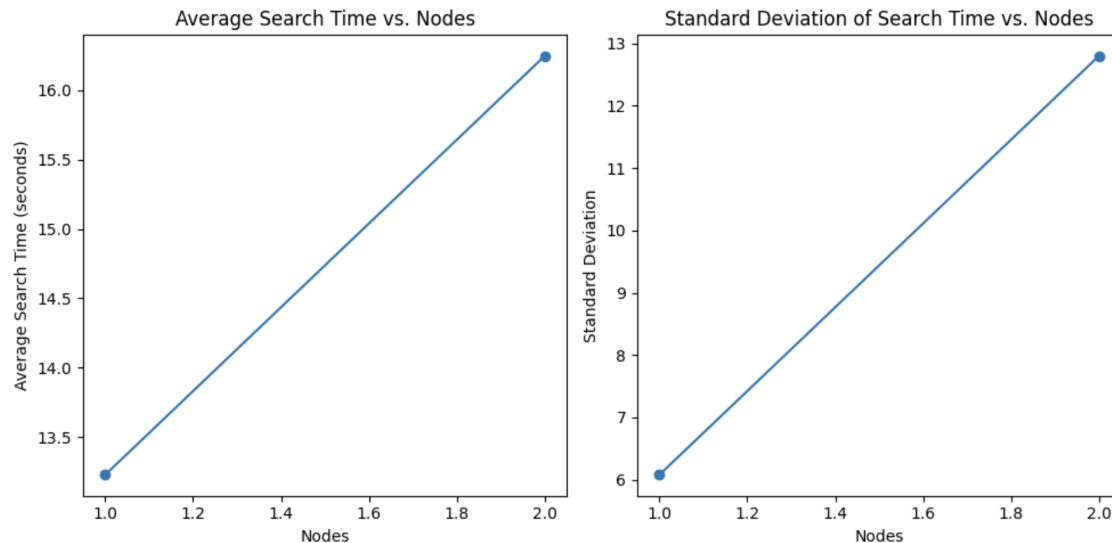
**Mean: 13.2272**

**Standard Dev: 6.0773426013676**

**For 1M File:**

**Mean: 16.248**

**Standard Dev: 12.80**



*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.

