Design:

The program is designed to work as a Napster style peer to peer file sharing system. It runs one a single system based on the Linux environment and is created using python. The program uses sockets to connect to peers and the indexing server.

The program contains a central indexing server to register data in a python dictionary. The dictionary allows for peers to register the files present in their directories. Multi-threading allows for peer clients to simultaneously register files and perform other functions.

The index server facilitates a search function to allow for the peers to find where a file is located. The search is a linear search which goes through the dictionary until a match is found based on the key.

The index server also provides a obtain option which allows the peers to connect to the peer where the file is located and download the file from there. The download of files is done using a simple copy function to copy files from one peer's directory to another peer's directory.

Improvements:

The program is designed in such a way that we assume the user enters the exact name of the file with case-sensitivity. We can use a function to allow for less restrictive inputs.

The program does not check whether the file is present at a peer or not. The user is expected to enter the right data as inputs. We can improve this by going to each peer's directory and cross checking with the users inputs.

The download of files is done via copying files from one directory to another and this will not work when the peers reside on different systems. We can overcome this by use of sockets and file sharing functions.