# Programming project requirements for COMP4621 2024 Spring

In this project, we will develop a p2p application, including a centralized directory server and p2p client that exchange files. The sever works as a global file information system (just like the DNS server). The client works as a p2p server to serve files to other p2p nodes at the same time as a p2p client that downloads files from other p2p servers.

## Requirements:

1. Programming in C
2. You must implement rdt3.0 based on UDP to deal with all communications between the server and the client.
3. The server should be able to provide services to multiple clients simultaneously. That means it should provide a concurrent rdt3.0 based on UDP.
4. You must implement a p2p server and p2p client in “client.c” file. Start the p2p server using the multi-thread function when the client is started.
5. The p2p server must handle concurrent p2p clients using the poll() method.
6. The submission must contain two code files “client.c” and “server.c” as well as a report of 2-3 pages describing your approach in a file named with your <studentid>.pdf.
7. Your code must pass the provided test scripts, which will test your code with some simple commands.

## Marking Scheme:

1. Your code passes the provided test1 for the simplest scenario. This test case will test your code with two clients without packet loss and without latency. Both these two clients connect to the server one by one and one client downloads a file from another client. (10%)
2. Your code passes the provided test2 for concurrency of the server. This test case is similar to the above one. Except that these two clients connect to the server at the same time. (10%)
3. Your code passes the provided test3 for multi-threading of p2p client and server. There are three clients start one by one. Client2 downloads a file from client1 and client3 downloads a file from client2 at the same time. And they are required to finish the transfer within 10s. (10% + another similar case for 10%)
4. Your code passes the provided test4 for packet losses. We will test your server code with a modified client, which will drop some ACKs for RETURN messages. (10% + another similar case for 10%)
5. Your code passes the provided test5 for packet losses. We will test your client code with a modified server, which will drop some ACKs for REGISTER/UPDATE/QUERY messages. (10% + another similar case for 10%)
6. A report for this programming project. (20%)