CSC573 Project Demo Rubric

1 Project 1 Demo

Server	40
Once a peer has connected to the server, the server should update the data structure with peer reported states including RFC number, title, hostname, upload port number When a peer requested for an RFC number, the server should reply with all the available places where store the document When a peer requested for whole index of RFCs from the server, the server should response with all its maintained information in the data structure When a peer left the system, the server should delete the information associate with the peer	
Peer	40
A peer requests a specific document from its peers with correct request format. Peer has the document (200 OK) A peer requests a specific document from its peer with an unexpected format (404 Bad Request) A peer requested document is not in destination any longer (404 Not Found) A peer requests a document with a not supported P2P-CI version (505 P2P-CI Version Not Support)	
Concurrency	10
Server has a capability to handle more than two peer connections	
Message Format	10
All the message (P2P, P2S) formats should follow the requirement in the document	

2 Project 2 Demo

Prepared a file with reasonable size to do all three tasks	10
Successfully transfer file from the client to the server	10
The argument of both server and client program should not be hard coded and able to be tuned at run time	10
Demo of task1 with tunable window size	10
Demo of task2 with tunalbe MSS	10
Demo of task3 with tunable probability of packet loss on the server side	10
Print out pkt loss with sequence number on the server side, Print out ACK timeout on the client side	10