# INFO 72220 Lab 4 – IPC over Sockets

Due Date: November 13, 2020

Submission Format:

* 2 C programs

In the class, we have learned that Socket is one of the ways to facilitate IPC, and we have supplied the TCP and UDP versions of the client-server sample code for your studies and project integration. This lab is intended to investigate how to design two processes on two different computers across the network to communicate with each other.

Modify the TCP/UDP sample codes so that you can:

* Launch the client and the server programs on two different computers   
  (or two different VMs with different IPs)
* Connect the client to the server
* Implement the client with a command-line menu that can send three commands -“Who”, “When”, and “Where” – to the server via the Socket, and receive and display the responses from the server. All client interactions should be outputted on the command line window on the client end.
  + The client also needs to implement the exit command and help information
* Implement the server that can accept client communications, respond to the client with the following answers – “Who”: INFO72220, “When”: [current time], and “Where”: Cambridge – via the Socket. Detail server activities should be outputted on the command line window on the server end.
  + The server doesn’t have to deploy an exit command (above and beyond)

**Requirement-based Marking Scheme (35 marks in total)**

* Client and Server can launch and connect with each other **(10 marks)**
* Client deploys the interaction menu – 3 commands, 1 exit, 1 help **(5 marks)**
* Client can successfully send commands to the Server **(5 marks)**
* Server can successfully respond to the Client’s command, and recognize the incorrect commands and notify the client accordingly **(5 marks)**
* The Client and Server run continuously until the user deliberately shuts them down **(5 marks)**
* All codes are well documented and structured **(5 marks)**
* Your program **MUST** compile without syntax error in order to receive any marks. The professor will not spend time figuring out how to get your program compiled properly. (as a result, a compilable partial solution is much better than a uncompilable full solution)