# Assignment 1

# 3P01

**{Due Date Feb.13 2015, 4:00. Late date Feb. 18 2015, 4:00}**

Objective:

Using Socket programming create a client server application.

Assignment:

Consider a server application which will calculate math expressions for clients. Clients will connect to the server. The connection dialog will look as follows on the client side:

Welcome to the Big Math Expression solver

Enter you username: **Client1**

Enter your password: **Some Password**

Welcome Client1

Enter an expression to evaluate: **7 + 5**

7 + 5 =13

Enter an expression to evaluate: **A + B**

A + B is not valid.

Enter an expression to evaluate: **Exit**

Bye Bye Client1

A username must be in the format shown including case. The number can vary e.g. Client301. The password can be any valid string with mixed case, alpha only. After the user has entered the password, the server will validate the username and password. If either does not conform, the server will issue a message stating so and break the connection.

For the purpose of this assignment, mathematical expressions will always be in the format **operand operator operand**, where operands are valid numbers and the operator can be **+ - \* /.** Any other input is invalid.

Since this is an internet application, it makes sense that multiple clients can connect to the server simultaneously. It also makes sense that each client is unique, that is no two clients can log in with the same username.

As part of the assignment ensure once a client opens a connection, the server moves the socket port away from the main port to allow other clients to connect. Clients will always connect to the server using port **43001**. The server has ports **43005** to **43009** allocated to it. This implies that the server can only handle 5 simulations clients. Should a sixth client attempt to connect, they sit there and wait.

Your assignment must be written in C using the gcc compiler. A working example is available on the website under examples. The actual platform is up to you, cygwin, linux will work fine. Ensure that you test you program by opening up multiple terminal window.

## Submission

1. Completed and properly documented source code for client and server.
2. Hard copy to be submitted to the assignment drop box across from J333. Include a completed coverpage.
3. Electronic copy to be submitted: <submit c3p01a> on sandcastle.

db