Weeks 1-8 Topics

1. **Review of basic concepts from Computer Communications & Networking**
2. **Local Area Networks**
   - Extending LANs using Repeaters and Bridges
   - MAC/Hardware Addressing
   - MAC frame formats
   - Bridges – operation, functionality, routing (frame filtering) and routing table
3. **Internetworks**
   - Universal service provision and associated issues
   - Routers and protocol software
   - IP Addressing (classful, classless, netid/hostid etc.)
   - IP Address Allocation and Address Tables
   - IP Datagram Format
   - Forwarding IP datagrams, routing tables etc.
   - Encapsulation
   - Address Resolution Protocol (ARP)
4. **Protocols, Protocol Architecture and Reference Models**
   - What is a protocol?
   - Difference between a *service* and a *protocol*
   - Protocol Architectures layer-by-layer
     - ISO-OSI 7-Layer model
     - TCP/IP 5-layer model
   - Encapsulation at work (Wireshark)
   - The transport services to upper layers
   - The transport entity
5. **Client-Servers and C-S Interaction**
   - Characteristics of clients and servers
   - Multiple services on one server-class machine
6. **Berkeley Sockets (the socket primitives)**
   - Example client and server application (*daytimeClient/Server* and *echoClient/Server*)
   - The primitives explained
   - Sockets, Listening and Connected Sockets
   - Addressing, port numbers and socket pairs
   - Socket address structures and socket addresses
   - Byte ordering/manipulation functions (inet_pton and inet_ntop)
7. **The WWW**
   - Browser Architectural Components
   - Telnet interaction with a HTTP server
   - HTTP clients-server interactions (theory and practical)
   - HTTP/1.0 and HTTP/1.1 requests/responses as a Protocol Box Diagram and in Syntactical form.

Lab Exercises completed:

- Installing the following software onto you own host machine: Virtual Box Manager, WinSCP, PuTTY and Wireshark.

- Setting up two Virtual Machines on your own host computer including: NICs and IP configuration and, installing ***gcc***. Testing the configuration using **ping**.

- Looking at packets using **Wireshark**.

- Downloading the following files onto each of the VMs: **daytimeClient/Server.c** files, **Practical.h** and **DieWithMessage.o**

- Creating and testing the following executables: **compile** script, **daytimeClient/Server**, **echoClient/Server, basic httpclient/Server and complex httpServer (with File functionality)**.