Your task for lab 1 will be to set up a simple interface with VNET. Look at our provided dummy IP layer—not the source code, which you do not have, but at what the executable does. You should be producing something similar.

You'll have a config setup file. I will need to know who is in what group so I can assign unique port numbers for your VNET. The reason for this is that two different groups might want to use hydra3 at the same time as one of their hosts. If both groups happen to like 5555 as the port, then we have conflicts. So we'll give you port numbers.

You can have up to 4 hosts (or more) on your LAN. Let's say that you want to call your hosts delta1, delta2, delta3, and delta4 running on hydra1, 3, 12, and 13. You want “IP” address 10.10.5.0 as your subnet, with delta1 as 10.10.5.1, etc, and you have port 5099 as the VNET port for each of the 4 hydra machines. Typically, you'll open windows to these hosts via ssh. You'll fire up VNET on the 4 hosts—same config file on each. Then for the hydra1 (= delta1) host, you should be able to type in a message and a destination—and your “IP” layer on delta1 calls VNET with the address of the message buffer, the size of the message, and the next hop—e.g. 10.10.5.2 or delta2—where the message should be delivered.

You need not worry about IP protocols—e.g. source and destination IP address fields in the packet—not yet. When we (i.e. you) add the UDP layer, then you need the IP protocols.

You'll need a minimum of 2 threads (not counting VNET)—one to watch for keyboard input, the other watching for input coming up from VNET.