Purpose: This assignment requires an in-depth understanding of using broadcast messages over UDP. You will build a time client that sends broadcasts and a time server that listens for broadcasts. The time will be sent using the `gettimeofday` format used in your `timec.c` and `timed.c`.

THE BROADCAST TIME SERVER

The `btime` server will listen on your well-known UDP port. The server will enable the receipt of broadcast messages. Whenever it receives a message from a client it will send the time back to that client (gettimeofday).

Except for responding to broadcasts, this is the same as the `timed.c` server you submitted as homework.

THE BROADCAST TIME CLIENT

The `btime` client will accept input from either the network or the keyboard. You will need to put in a `while(1)` loop and a select statement. Except for using UDP broadcasts, this part is similar to your `timec` client.

If there is input from the keyboard, only the first letter of the input is examined. If that letter is a 'q', the client exits. If that letter is a 'b', the client sends a broadcast message to all servers listening on your well-known port. All other keyboard inputs are ignored.

If there is input from the network, it is 8 bytes of time coming in from some server that received your broadcast. Unpack and print (`%x %x`) (exactly like you did in the `timec` homework) the time the server sent to you (seconds, milliseconds). Also print the address (`inet_ntoa`) of the server you got the response from (like you did in timec).

Submit: The a fully commented copy (print-out) of the source code for the client and the server. In addition, the source code for the client must be placed in your home directory in a file named `btimec.c` and the source code for the server must be placed in your home directory in a file named `btimed.c`.

Discussion:

When you send a broadcast, each server on the subnet should send a response. For example, if you start three servers and your client sends a broadcast (user types 'b'), then the client should print 3 times, one from each of the servers. Remember, UDP is "unreliable" so occasionally one of the servers may not respond.

To test this, start 3 servers and one client on the same subnet.

There are two subnets in the lab, both with netmask 255.255.255.224. (That's 32 host number per subnet.) These subnets are 134.139.248.64 and 134.139.248.32. Make sure all three of your servers and your client are on the same subnet. You cannot use `cheetah` for testing this assignment, it is not on one of the above subnets.