Purpose: This is designed to familiarize you with TCP clients.

Build a TCP time client in a file called `TCPtime.c`. Your client will contact the "special" time server located on port 5001 of panther.

Use the main program–client procedure format. The default service should `"5001"`. The default host should be `"panther"`.

The time server will pass you back an 4 byte integer representing a UCT time in network standard order. You will read that integer, convert it to host order, and then UNIX epoch time, then print it using the `ctime` system call. For an example of how to convert and print the integer see Comer's `UDPtime` client. You will need to include `time.h`.

If the read fails, or the connection closes before you have received 4 bytes; print an error message and exit. No call to `ctime` is made in this case.

You don't have to handle the case of "can't make a connection" because Comer's code will print an error message and exit for you.

Discussion: One of the points of the TCP discussion was that, in a stream, boundaries are not preserved. You will need to assemble the 4 bytes (since they may come in on separate reads) before doing the conversion. The `panther` server is perverse and will try to force the bytes to arrive at your client in separate reads. It will occasionally lose (close) the connection (1 time in 5).

Submit: a printout of your `TCPtime.c` file.

A copy of this file must be located in your class home directory; and have the name above. This is to allow the instructor to use a script to test your homework.

Testing: what you should see is something fairly close to the current date. About 1 time in 5 you should see the error message.