IC221 Socket Programming Lab, AY22S

Description In this lab, you will complete the implementation your own version of the netcat utility, called navycat. Format:

./navycat <hostname> <port>

Submission: navycat.c

Grading

* The program navycat.c compiles with no errors or warnings. (10%)
* Reports all errors (failure to supply arguments, connection failure) to stderr. (10%)
* Proper resolution of domain name (15%)
* Proper socket connection (15%)
* Program is multithreaded and each thread loops without crashing after one read/write (10%)
* The program reads whatever is written to the socket and sends it to stdout (20%)
* The program writes anything from stdin to the socket (20%)

Testing

You should test your navycat program against the built-in netcat, on the same machine (localhost). For example, in two *separate* terminals:

$ netcat -l -p 1992 # one terminal

$ ./navycat localhost 1992 # another terminal

Your program should be able to perform continuous, two-way communications with the built-in netcat program, regardless of whether input comes first from the socket (the netcat end), or from the user (stdin). Reminder: use -lpthread at the end of your compilation command, as in:

gcc navycat.c -o navycat **-lpthread**

Errors

You should print error messages using perror (see course notes for examples). Sample output:

$ ./navycat

ERROR: require arguments

$ ./navycat a

ERROR: require arguments

$ ./navycat a b

ERROR: invalid port

$ ./navycat a 10

getaddrinfo: name or service not known

$ ./navycat mich302csd01u 10

connect: connection refused