GROUP:

* Mark Gross (mag210)
* Mike Folz (maf152)

FILES INCLUDED:

* client.c
* server.c
* multipleclient.sh
* sudokusolver.c
* sudokusolver.h
* puzzle1.txt
* puzzle2.txt
* puzzle3.txt
* puzzle4.txt
* puzzle5.txt
* makefile

BASIC PREMISE:

* Our project is a client-server Sudoku solver. The output printed to terminal for each client is the originally submitted unsolved Sudoku board and a solution to the board if a valid solution exists. Otherwise it will output that no valid solution exists.

HOW TO RUN:

It is expected that this is run on one of the Case eecs servers.

* compile the necessary files using the “make” command
* Open 2 shell windows. One can run the server, the other can run the client.
* NOTE: We included some pre-made puzzle files but they can be edited. When editing, the program interprets zeros as empty spaces in the Sudoku board.
* STARTING THE SERVER:
  + Use command: $ ./server
* RUNNING A SINGLE CLIENT:
  + Use command: $ ./client 127.0.0.1 8000 puzzle1.txt
    - Note: port number (8000) can be any value from 8000 to 8004
    - Note: puzzle (puzzle1.txt) can be any puzzle text file included
  + Note: The server does not terminate until 5 clients have run, each on a different port number (5 is the max number of clients our server can handle at once). To terminate before running five clients, just use the Ctrl+C keyboard shortcut.
* RUNNING MULTIPLE CLIENTS:
  + Use command: $ sh ./multipleclient.sh
  + This runs 5 clients and prints their outputs to terminal. The server automatically terminates after all 5 clients have run.

OUTPUT:

All output is found on in the terminal.

* Server prints that its threads are initialized
* Each Client prints its unsolved and solved board (or that no solution exists)
* Example Output:

