

10 May 2015

Jakub Ciecierski

Unix Programming

Time extended game of checkers.
Documentation.



1 Project Description

Server holds any number of checkers games, clients connect at any time to check current game state and make next move. Server does not validate the moves according to the rules, the only checks server does:

- is it player's turn ?
- is disc of player colour at the starting position ?
- is destination position empty or occupied by the opponent piece ?
- If empty, just make a move, if occupied, remove opponents disc.

Game is over when one of the players give up (server does not check the terminal rules of checkers game).

When connected to the game user can:

- display the board
- check game status (active,resolved, waiting for opponent)
- check the turn (mine,opponents)
- list all moves made
- read messages from the opponent
- make move in form of coordinates e.g. b6 to c6
- leave any number of messages to the opponent
- give up

When connected to the server user can:

- identify himself with nick (no passwords)
- connect to the game of given id (only, if he is the player in this game)
- list all games connected to him (by nick)
- start a new game:
 - if there is no other new game on the server
 - new game with status "waiting for opponent" is started
 - if there is a new game without a opponent, the player joins this game
 - id and status of a new game is sent to the player

All game data must be stored in folders and files. All network communication is based on tcp. Server does not remove resolved games. Each client connection must be handled by it's own process. All interprocess communication must be based on sysV IPC.

2 Modules

Source files are stored in *src* folder and contains the following modules

- util - all the common utility functions
- server - server related stuff
- client - client related stuff

TODO Detailed documentation.