Language Study: Erlang

CMPT 333

Mid-term Project - 150 points

Goals

- to continue bask in the glory that is recursion whilst enjoying the challenge of developing a "perfect" Tic-Tac-Toe game. "Perfect" in this context means that the computer should never lose; it should always be able to tie the human player or win.

Requirements, Notes, and Hints

Use a divide and conquer approach here, much like we do when using recursion to solve problems. Break the task into small parts and solve those parts individually. Here's some advice on doing that:

Phase One

- Represent the game with Erlang functions and constructs
- Use a list for the **X** and **O** and **blank** values on the board.
- Write functions to get and set those board values.

Phase Two

- Develop a natural interface to the game. Write a ttt() or play() (or whatever you want to name it) function that launches the game.
- The computer should:
  - display the board
  - prompt the user for the **X** player's move
  - make the **X** move
  - display the board
  - determine the **O** (computer's) move
  - make the **O** move
  - repeat until the end of the game

Phase Three

- Check all input for range validity, including that the user is not moving into an already occupied space.

Phase Four

- Detect the winner. You should probably check for this after every move.
- Detect ties as soon as possible. You can certainly tell if it's a tie after move nine because the board will be full and there will be no winner. You can detect a tie after move eight in most cases. There are some cases where you can detect a tie after move seven or even sooner.

Resources

Our book, links on our class website, and Erlang itself.

Submitting Your Work

Commit the following to your *Midterm Project* directory in your **private** GitHub repository on or before the due date (see our syllabus):

- your source code;
- all of your test cases;
- a transcript of two different successful runs where you handle unexpected input;
- a transcript of two different successful runs where the computer wins; and
- a transcript of two different successful runs the end in a tie.

There should be no runs where the human player wins. That's not supposed to happen.