CptS 483/580

Homework #5 – Concurrent Erlang first assignment

Assigned: 3/3/10

Due: 3/12/10 11:59:59PM. Turn in to the course turn-in page
http://www.eecs.wsu.edu/~hauser/cs483/turnin

As discussed in class on 3/1, this assignment is essentially problem 2 in section 8.11, p. 150 of the Erlang book. Further details are provided here.

Please implement your code for module `ring` in file `ring.erl` and turn in a file by that name. The file must export a function `ring(nProcesses, nMsgs)` that does all the work of creating the process ring, sending `nMsgs` around it, and reporting the total time taken. The program should work for values of `nProcesses` and `nMsgs` into the tens of thousands.

You may write additional functions and you may export additional functions – it's good practice to include some self-tests in a module that can be called from outside the module.

See the `processes` example on p. 141 for how to determine how much time an operation takes and for how to do formatted output. Refer to the documentation at http://www.erlang.org/doc/apps/stdlib/index.html for explanations of the primitives used in the example.

The purpose of this homework is to give you familiarity writing functional concurrent code and explaining how it works. Therefore, include at the beginning of your file a section of comments that explains how your code works – in particular explain the way that the processes are created and why you chose to do it the way that you did. Of course, also comment your individual functions (but these should be small and not require too much explanation).

Note: you do *not* need to implement this in another language nor create a blog entry about what you discovered!