CS 511 – Quiz 7: Message Passing in Erlang

8 November 2023

Names:
Pledge:

Exercise 1

We wish to model a dry cleaner in a factory using message passing. Employees leave their overalls for dry cleaning, pick up a clean one and leave. If there are no clean overalls, employees must block waiting for one. Multiple dry cleaning machines do the dry cleaning. You are asked to model the dry cleaner, the employees and the dry cleaning machines. The dry cleaner can process the following messages:

- `{dropOffOverall}` Sent by employee
- `{From,pickUpOverall}` Sent by employee
- `{From,dryCleanItem}` Sent by dry cleaning machine

Additional messages will be required. Also, you may find the use of "when" clauses useful in the patterns of a receive clause. Here is the stub you must complete. This code must be placed in a file called `dc.erl`.

```
-module(dc).
-compile(nowarn_export_all).
-compile(export_all).

dryCleanerLoop(Clean,Dirty) -> %% Clean, Dirty are counters
     todo.

employee(DC) -> % drop off overall, then pick up a clean one (if none
                % is available, wait), and end
    todo.

dryCleanMachine(DC) ->  % dry clean item (if none are available, wait),
                        % then sleep for a while (timer:sleep(1000)) and repeat
    todo.

start(E,M) ->  % E= no. of employees, M= no. of machines
    DC=spawn(?MODULE,dryCleanerLoop,[0,0]),
    [ spawn(?MODULE,employee,[DC]) || _ <- lists:seq(1,E) ],
    [ spawn(?MODULE,dryCleanMachine,[DC]) || _ <- lists:seq(1,M) ].
```