**TC2006 – Lenguajes de Programación***Programación Concurrente en Erlang*

Matrícula1\_\_\_\_\_\_\_\_\_\_\_ Matrícula2\_\_\_\_\_\_\_\_\_\_\_\_

1. Implementar el proceso **hola** en Erlang que acepte un mensaje {hola, P}, donde P se asume que es un pid. Para cada mensaje, el proceso debe mandar {reply, C} a P (donde C es un contador.)

**Programa de prueba**.

prueba\_hola() ->

H = spawn(hola, hola,[]),

prueba\_hola (10, H).

prueba\_hola(N, H) when N > 0 ->

H ! {hola, self()},

receive

{reply, C} ->

io:format("Recibido ~w~n", [C]),

prueba\_hola(N-1, H)

end;

prueba\_hola(\_, \_) ->

io:format("Mi trabajo está hecho").

También trata de probar el proceso **hola** interactivamente.

1. Implementar el proceso **calcula** en Erlang que maneje los siguientes mensajes:

* {suma, X, Y} calcula la suma de X y Y
* {multiplica, X, Y} calcula el product de X y Y
* {ultimo, P} manda un mensaje de la forma {resultado, L} al proceso P. L debe ser el último valor calculado.
* {total, P} manda un mensaje de la forma {resultado, S} al proceso P. S debe ser la suma de todos los valores calculados.

Escribe el programa **prueba\_calcula** como el programa **prueba\_hola** del ejercicio anterior para probar tu código.