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

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

1. Programar la función **formula\_general** en Erlang que resuelva ecuaciones cuadráticas de tipo a x^2+b x+c = 0, a partir de sus coeficientes.

Probar con:

> ej1:formula\_general(1,-5,6). => {3.0,2.0}

> ej1:formula\_general(3,2,2). => imaginarias

1. Programar la función **secuencia** en Erlang que cree una lista con los números naturales del 1 a N.

Probar con:

> ej1:secuencia(0). => []

> ej1:secuencia(5). => [1,2,3,4,5]

**secuencia ( 0 ) -> [];**

**secuencia ( N ) when N > 0 -> secuencia(N - 1) ++ [N].**

1. Programar la función de orden superior **mapea** en Erlang que trabaje igual que la FOS lists:map (sin utilizarla).

Probar con:

> ej1:mapea(fun math:sqrt/1, [1,4,9]). => [1.0,2.0,3.0]

> ej1:mapea(fun(X) -> X\*X end, [1,2,3]). => [1,4,9]

**mapea ( \_ , [] ) -> [];**

**mapea ( F , [ X | XS ] ) -> [F(X)] ++ mapea(F , XS).**

1. Programar la función **menores** que sin utilizar recursividad explícita regrese todos los elementos de una lista de sublistas que sean menores que su segundo argumento.

Probar con:

> ej1:menores([[1,2,3],[4,5,6]], 4). => [[1,2,3],[]]

> ej1:menores([[2,-3],[-1,2],[5,2]], 1). => [[-3],[-1],[]]