Programming Paradigms Exercise 8 - Erlang 1

Johann Gamper Marco Montali Thomas Tschager

2nd Semester 2017/18

1. Write a function `sign(N)` (in a module `mymodule`) that gets a number `N` as an input parameter and outputs -1 if `N` is negative, 1 if `N` is positive, and 0 if `N` is equal to zero.

2. (a) In a module `greeter` write a function `hello` with one input parameter `Name`. The output of the function should be "`Hello "Name"!`" on the console. (Hint: the Erlang command for printing is `io:format`)

   (b) In the console write an anonymous function that does the same as in (a) and assign it to a variable.

   (c) Use the anonymous function in (b) and apply it to a list, greeting every element in the list.

3. Download `flights.txt` and paste it into the Erlang shell. The file stores flight data in a list of tuples. Each tuple contains the departure and arrival city, the flight number, and a list of days of the week on which the flight operates. For example, in the tuple

   ```
   {paris,new_york,124,[tu,we,th,sa,su]}
   ```

   `paris` is the departure city, `new_york` is the arrival city, `124` is the flight number, and `[tu,we,th,sa,su]` is the list with the days of the week on which this flight operates.

   (a) Write function `flight:fromto(A,B,L)` that given a departure city `A`, an arrival city `B`, and a list `L` of flights, finds all flights from `A` to `B`.

   (b) Write a function `flight:fromon(A,D,L)` that given a departure city `A`, a departure day of the week `D`, and a list `L` of flights, outputs a list of cities than can be reached from `A` on `D`.

   (c) Print in the console all the flights from `paris` on each day of the week. (Hint: use the same approach as in 2c)

4. Write a function that just echoes whatever message it receives via `io:format`, i.e., it will print any received message in the console. Start a process running this function and send it several messages from the shell.