We have a grid of \( N \) rows and \( M \) columns. On each cell \((i, j)\) of the grid, a device (an Erlang process) is making some task. Concurrently with the execution of the devices in the grid, there is a monitor which controls the whole system. The monitor can receive the following messages from the devices:

- \{\text{fail},i,j\}: there has been a failure for the device in cell \((i, j)\) (this message has been issued before the device’s termination). In this case a new device must be created to substitute the faulty device.

- \{\text{finish},i,j,Pid\}: the device in cell \((i, j)\) has finished. Depending on the number of tasks to be performed (defined when the monitor is created), the device can be re-started by sending the message \text{restart}.

- \text{statistics}: the monitor prints the number of failures and the number of finished tasks.

We ask you to provide the functionality of the monitor, assuming that the functionality of the devices is already implemented. A device can be created with the following sentence:

\[
\text{DevicePid} = \text{spawn(device, init\_device, [i,j,PidMonitor])},
\]

More concretely, for the monitor you have to provide two main Erlang functions:

- \text{init\_monitor}(N,M,Number\Of\Tasks): the Erlang function that creates the system. Here we assume \text{Number\Of\Tasks} is greater than \( N \times M \). For each device created, a message \text{start} must be sent in order to start its execution.

- \text{global\_monitoring}(Number\Of\Remaining,Number\Of\Failures,Number\Of\Finished): function that performs the main functionality of the monitor, i.e., interaction with the devices and printing statistics.

With respect to the device, make a simple implementation were the device randomly may fail.