

# Practice Script 6

## Summary:

- Object-Oriented Concurrent Programming (shared object communication model).
- Copy-on-Write (CoW) synchronization scheme.

## Exercise 6.1

It is intended to build a program to view and solve mazes. To that goal, a package (`pt.ua.gboard.jar`) supporting a simple graphic design containing classes to handle labyrinths will be used. In the class `Labyrinth`, labyrinths can be defined by text files, in which `S` denotes the starting point, the symbol `X` the point of arrival, the spaces corresponding to the road and all other symbols are wall.

Start by finding first good Abstract Data Types to solve the problem. Then, implement a solution based on a backtracking sequential approach.

For a concurrent labyrinth solving, try a solution in which a thread is forked in road crossings (and dies silently when reaches an already traveled road). Define a proper termination of threads when a path is found.

## Exercise 6.2

Solve the maze problem using a *copy-on-write* (CoW) synchronization scheme.

To that goal, consider that attached to each search thread a simple linked list containing the positions already traveled (a new position links to the previous one).

## Exercise 6.3

Change the problem 6.2 in order to find the shortest path (if any).

## Exercise 6.4

Implement a shared stack with a linked list using a CoW synchronization scheme.

