

## PCD1718 - Assignment #3

### **1) *Game of Life - Actor-based***

Implementare in Java o in Scala una versione ad attori del gioco "The Game of Life" - già incontrato nell'assignment #1. Considerare i medesimi requisiti specificati per l'assignment #1. In questo caso la soluzione deve essere basata completamente sul modello ad attori (usando Akka come tecnologia di riferimento), eventualmente integrandolo con parti non ad attori quando necessario (es: uso librerie per la GUI). Valutare le performance del sistema, includendo le relative considerazioni sul report.

NOTA: è possibile riusare le parti già sviluppate con l'assignment #1, incluse le soluzioni presentate a lezione.

### **2) *Actor-based distributed chat***

Implementare una chat distribuita ad attori, con insieme dei partecipanti dinamico - ovvero un utente può aggiungersi e rimuoversi dinamicamente. Ogni messaggio inviato da un utente deve essere visualizzato da tutti gli altri utenti. Il sistema deve essere completamente decentralizzato, a parte - eventualmente - la presenza di un (attore) registro - con indirizzo/nome noto - che tenga traccia dei partecipanti. I messaggi inviati nella chat devono essere visualizzati da tutti i partecipanti nel medesimo ordine.

Facoltativo: Supporto per una modalità "sezione critica": un partecipante può chiedere di entrare in sezione critica inserendo un comando predefinito (es: ":enter-cs"). Quando un partecipante entra in sezione critica, possono essere visualizzati solo i suoi messaggi, senza intervallarli a quelli degli altri utenti. Un solo partecipante alla volta può essere in sezione critica. Per uscire dalla sezione critica si può prevedere un altro comando predefinito (es: ":exit-cs"). Un utente può rimanere in sezione critica per un certo tempo massimo  $T_{max}$  - dopodiché l'uscita è forzata. Durante una sezione critica i messaggi inviati dagli altri partecipanti devono essere rigettati.