

1 Introduzione

Il progetto consiste nella creazione di un ambiente bidimensionale in cui sono presenti delle entità, detti robot, e delle forme geometriche. I robot possono muoversi nello spazio tramite una lista di istruzioni assegnatogli.

2 Responsabilità assegnate

- “Entity”, interfaccia responsabile della definizione di un'entità nel sistema. Permette alle entità di percepire l'ambiente e di eseguire le istruzioni del programma assegnatogli. L'entità, grazie alla percezione dell'ambiente, è capace di conoscere la posizione delle altre entità e di comprendere se risiede all'interno di una determinata forma geometrica.
- “Environment”, definisce l'ambiente in cui risiedono le forme e le entità. Permette alle entità di interagire con esso in modo che esse possano percepirlo.
- “Command”, interfaccia che rappresenta un comando del programma, permettendone l'esecuzione e di conoscere se il risultato è andato a buon fine.
- “IterativeCommand”, i comandi che devono permettere l'esecuzione di altri comandi al loro interno sono rappresentati da questa interfaccia. Essa è un'estensione di “Command” e permette di gestire l'aggiunta dei “Command” al comando iterativo.
- “Simulate”, interfaccia che si interessa della simulazione del sistema. Rende possibile l'esecuzione del programma da parte dei robot.
- “Robot”, implementazione dell'entità, segue i principi di “Entity” e riconosce se si trova all'interno di un rettangolo o di un cerchio.
- “Direction”, rappresenta la direzione, da un punto verso un altro, nel sistema. Permette quindi lo spostamento nell'ambiente 2D.
- “RobotSwamrSimulator”, implementazione della simulazione, viene inserito un tempo dt , corrispondente all'unità di tempo assegnata ad una singola istruzione, ed un tempo $time$, che rappresenta il tempo massimo assegnato alla simulazione. Permette l'esecuzione passo-passo del sistema, in modo da percepirne i cambiamenti in tempo reale. Permette inoltre di sapere se la simulazione è finita o meno.
- “Controller”, responsabile della gestione delle entità, delle forme geometriche e delle istruzioni, utilizza “ParserHandler” per effettuare il parsing di quest'ultime. Permette anche il parsing delle entità e delle forme geometriche e di far comunicare l'ambiente con l'utente.

- “SwarmSimulator”, rappresenta l’UI del sistema, comunica col “Controller” per ricevere le entità del sistema e modificarne la posizione.

3 Come avviare il progetto

Il progetto utilizza il sistema di gestione delle dipendenze Gradle, perciò per poterlo avviare è necessario posizionarsi nella cartella principale del progetto ed eseguire i seguenti comandi:

- *gradlew build*
- *gradlew run*

In seguito apparirà una finestra in cui l’utente può decidere di caricare i propri file di configurazione, oppure usare quelli di default.