## GameMaster

### CRC

|  |  |
| --- | --- |
| GameMaster | |
| Responsibilities: Hold a game board and control a robot based on commands from the fourth interpreter. | Collaborators: The game master collaborates with the fourth interpreter to receive and send information about the game state. It talks with Game to get that information, and Hex Node to for selecting a specific tile. Finally the Game Master collaborates with Robot to keep track of which robot is currently playing. |

### Description / Overview

GameMaster will be keeping track of the overall mechanics of the game such as: controlling robots, (turning, shooting, firing, scanning and identifying) and communicates with the fourth interpreter.

### Instance Variables

#### game

Data Type:Game

Holds an object of type game class this object includes the game board as well as some key elements for the game master to its job.

#### selectedTile

Data Type:HexNode

SelectedTile is the tile that the player has clicked on.

#### curRobot

Data Type:Robot

CurRobot is the robot who is currently playing.

#### forthInterpreter

Data Type:ForthInterpreter

The forthInterpreter holds a refrence to a ForthInterpreter object.

### Method Overview

*public GameMaster(Game game)*

*public void damageTile(int damage, HexNode node)*

*public void moveRobot()*

*public void turnRobot()*

*public int scan()*

*public void identify(int index)*

*public void getCurrentRobot()*

### Method Writeups

#### public GameMaster(Game game)

The constructor for the GameMaster takes in a pre-built board and gives back a GameMaster object.

#### public void damageTile(int damage, HexNode node)

DamageTile takes in an integer for how much damage to do and a HexNode for which tile to damage. Damage will be the robots damage and HexNode will be the selectedTile.

#### public void moveRobot()

MoveRobot moves the robot one tile in the direction that he is facing if he is allowed to. Facing a null tile or out of moves.

#### public Team getCurrentTeam()

GetCurrentTeam will ask the curRobot for his colour attribute.

#### public void turnRobot()

TurnRobot will rotate the robot to face an adjacent tile when one is selected. If its not an adjacent tile the robot will not turn. (turning a robot does not cost the robot a move!!)

#### public int scan()

Scan returns the number of robots currently in view of the current robot.

#### public void identify(int index)

Identify will take will push TeamColour, how far the robot is away and the direction the robot is in, as well as the robots remaining health.