**InterpreterFunctions:**

This class acts as an interface to the system for the Forth interpreter. It provides a list of functions that directly translate from the Forth language functions, allowing the Interpreter to simply call whichever one matches each word in a body of Forth text. These functions will act on a stack, and will call the internal functions of the Robot class.

Interactions:

When the Interpreter traverses a list of Forth code, it will call these InterpreterFunctions

These functions will then call the robot class to act out the desired functionalty.

Because the Robot.move() function requires coordinates to be passed in as an argument, it is necessary to call on the GameBoard’s getRobotCoord function.

Variables:

* stack: a stack which holds all variables for the functions to use.
* Robot: a robot whose functions will be called

Functions:

* InterpreterFunctions() Initialize the above variables.

* setRobot(Robot): sets the robot whose functions will be called.

All of the following functions mirror those in the RoboSport370 Language document unless additional specification is given.

* String pop(): returns the value off of the top of the stack.
* push(String): pushes a Word onto the stack
* popAndPrint(): pops the stack and prints the result.
* drop()
* dup()
* swap()
* rot()
* arithmetic()
* modulo()
* comparison()
* and()
* or()
* invert()
* conditionStatement()
* guardedLoop()
* countedLoop()
* random()

The following functions will call the Robot functions using variables popped from the stack, and pushing their return values to the stack.

* health()
* healthLeft()
* moves()
* movesLeft()
* attack()
* range()
* team()
* type()
* turn()
* move(): This function will need to call getRobotCoord from the GameModel with the robotID.
* shoot()
* check()
* scan ()
* identify()
* send ()
* mesg()
* recv()