Artificial Intelligence Assignment 2019

About the Game:

The purpose of this assignment is to create an AI controlled maze game, from a set of features that were provided. The games objective is for the player to escape from the maze, by either fighting or avoiding the enemy characters that are roaming around the game environment. Stubs where provided to create a basic maze game, the player is controlled with the keyboard arrows, while the enemies are immobile. The player is given a variety of weapon choice, such as swords, help points and bombs.

Generating maize with multiple exits:

As required, the maze in the game must have one ore more exits. Where the player is placed far away as possible from the exits. Then the player is to find this exit while fighting the enemies. The maze is generated using brute force algorithm, this works by randomly creating a passage for the player when they move, either north or south and west or east both not both at the same time. The players movements are mapped, the number of nodes that is taken to reach this exit is outputted in the command prompt when the game is ran.

Sprite Features:

Player:

The player can be controlled around the maze with the keyboard arrows up, down, left and right. The player can obtain weapons from the maze by moving close to the maze and touching the edge of the maze. This weapon will be added to the collections of weapons. When the player is threatened by any spider the weapon, I automatically used to fight them away. The player can’t pass through the hedges of the maze, they must find another path to follow unless they may get caught of trapped in a dead end. The player most move continuously to avoid being caught by the spiders. The player also is tracked with a health score board. If touched by the spider, the health score decreases.

Spiders:

The enemy spiders have live scores this checks their health, when they are killed or low in energy. The spiders have extra features for checking their anger in the game, this can be seen in the nav -bar. All the spiders are fully threaded, using a cocktail of three different Traversal search algorithms, Astar, Breadth Fist search and a Recursive Depth First search algorithm. These find the players path, which sends the spider after the player even when it moves. The types of AI used are either fuzzy logic or neural network, this allows how the spiders attack the player.

AI Features:

The AI functionality of the game is seen in the navigation bar between the player and the spiders. If on of the spiders are a fuzzy logic type it will start a fuzzy-fight with the player. The input values for the fight function block are the value of the players weapon and the spiders anger level. The output value is the damage that is caused to the spider.

These rules are as follows:

* RULE 1 : IF playerHealth IS low THEN fuzzyLogic IS attack;
* RULE 2 : IF playerHealth IS medium OR angerLevel IS medium THEN fuzzyLogic IS attack;
* RULE 3 : IF playerHealth IS high AND angerLevel IS high THEN fuzzyLogic IS run;
* RULE 4 : IF playerHealth IS medium AND angerLevel IS high THEN fuzzyLogic is run;
* RULE 5 : IF playerHealth IS medium OR angerLevel IS low THEN fuzzyLogic is attack;

If any of the spiders are controlled by a neural network, it will start an nnfight with the player. There are two neural network activation functions used - Sigmoid and Hyperbolic Tangent.

Each neural network consists of:

* Three input nodes which represent spider health, player sword and spider anger
* Three hidden nodes
* Three output nodes which tell the spider how to react - either Panic (deducts health), Attack (deducts from weapon) or Hide (which deducts from health because no one likes a sissy)
  + Training data and expected data are contained inside the fight class and are used to train the Back Propagation Trainer