COMP 5211 Advanced Artificial Intelligence
Project 1: Reactive Agents

Ali Symeri
20591029

Problem 2

2.2 Marking scheme

1. Give the boolean expression that corresponds to your perceptron for North action.

Answer: The boolean expression for my North action perceptron [0.1, -0.2, -0.2, 0.0, 0.0, 0.0, 0.0, 0.1] is: $\overline{x_1}x_2x_3\overline{x_8}$

Problem 3

1. What's your fitness function?
   Answer: My fitness function is; the number of desired output – actual output (d-f) that are equal to zero for a given training set. This means that I have a population of x amount of randomed weight tuples (user input) and that for every training set tuple I compute the weighted sum for the each of the x amount of weights and check the error rate for all of these weights for this single training set. So, the number of computed errors that are equal to zero is what I use to increment the score for this weight. This also means that for every weight tuple where the error is equal to zero, its score gets incremented. Later on, the top 30% fittest are selected for crossover.
2. What's your crossover operator?
   Answer: My crossover operator (90%) is that I randomly take a father and randomly a mother from the population of weight tuples. Using the father and mother I create a child weight tuple that contains a random amount from the father and the remaining from the mother.
3. What's your copy operator?
   Answer: I do not use a copy operator as it could converge without it.
4. What's your mutation operator, if you use any?
   Answer: My mutation operator (1%) is that I for one percent of the children from the crossover operation, randomly select an index of the new child weight tuple and random a new number on that index.
5. What's the size of the initial generation, and how are programs generated?
   Answer: The initial size of the generation is 1000, but can vary with user input. Programs are generated for every training set entry in the list. So I loop through the list of training set and calculate the fitness and score the weights individually per training set entry. After this loop a new population is created with the fittest and mutated and crossovered weights, and the loop runs again on this new population.
6. When do you stop the evolution? Evolve it up to a fixed iteration, when it satisfies a condition on the fitness function, or a combination of the two?

Answer: The evolution stops whenever there is a valid perceptron for the entire training set, i.e. the desired output is equal to the actual output for the entire training set.

7. What's the output of your system for the training set in the next page? This training set is also included in project1.zip as gp-training-set.csv.
   Answer: The output I get is: [-0.11, -0.21, 0.57, -0.29, 0.97, 0.42, -0.82, -0.64, 0.34, -0.17]