COMP 3211 – Fundamentals of Artificial Intelligence
2026 Spring Semester – Assignment 1
Date Assigned: Feb 13, 2026
Due Time: 23:59 on Feb 27, 2026

How to submit it:

- Submit your written answers of Question 1 - Question 3 as a pdf file to canvas. If your submission is a scan of a handwritten solution, make sure that it is of high enough resolution to be easily read.
- Submit your codes for Question 3 and the Programming Assignment 1 (in another document) as a zip file named YourStudentID.zip to canvas which contains the three python files: `gp_system.py`, `production_rules_agent.py`, `state_machine_agent.py`.
- Please avoid plagiarism. ***You must acknowledge individuals who assisted you, or sources where you found solutions.*** Failure to do so will be considered plagiarism.
- No late submission will be accepted.

1. (10 points) Consider again our boundary-following robot in lecture notes (Page 5-6 in Lecture 2), assume there is no tight space in the environment. We want our robot to either go clockwise around the inside of the outer boundary, or go counterclockwise around the outside boundary of the solid object. But now our robot is somewhat sensory-impaired: its sensory inputs are only (s_2, s_4, s_6, s_8).

   1.1 Can you design a reactive (stimulus-response) agent for following a boundary based only on these four sensors? If your answer is yes, describe how such an agent can be designed. If your answer is no, give your reason for it.

   1.2 Now suppose that we have the following features for recording the last performed action and the sensor values in the previous state:

   - pa (previous action), with 4 values: N (the last action was heading north), E (the last action was heading east), S (the last action was heading south), and W (the last action was heading west).
   - ps_i: in the previous state, the sensor value of s_i is 1, where $i = 2, 4, 6, 8$.

   Recall that s_7 is 1 (true) if, relative to the robot's current location, the bottom left corner cell is blocked. In our boundary-following robot, this feature is determined by a sensor. Now, we want to compute s_7 when given the above features (pa and ps_i, $i = 2, 4, 6, 8$). You can assume that initially, when the robot is turned on, s_7 is 0. We want to find a way to compute s_7 using the above features after the robot has started moving.

Can s_7 be computed using the above features? If yes, describe how s_7 can be computed. If no, please justify in which cases s_7 cannot be computed and why.

2. (10 points)

   2.1 Which boolean function does the following TLU implement? The TLU has five inputs. Its weight vector is $(1.9, -2.5, 5.2, 2.3, -0.5)$, and the threshold is 3. Give your answer as a boolean expression.

   2.2 Which boolean function does the following TLU implement? The TLU has five inputs. Its weight vector is $(1.9, -2.5, 5.2, 2.3, -0.5)$, and the threshold is 10. Give your answer as a boolean expression.

3. **(Programming)** (30 points) Design and implement a genetic programming system to evolve some perceptrons that match well with a given training set. A training set is a collection of tuples of the form $(x_1, ..., x_n, l)$, where x_i's are real numbers and l is either 1 (positive example) or 0 (negative example). So for your genetic programming system, a "program" is just a tuple $(w_1, ..., w_n, \theta)$ of numbers (weights and the threshold). Answer the following questions:

   3.1 What's your fitness function?

   3.2 What's your crossover operator?

   3.3 What's your copy operator?

   3.4 What's your mutation operator, if you use any?

   3.5 What's the size of the initial generation, and how are programs generated?

   3.6 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?

   3.7 What's the output of your system for the provided training set `gp-training-set.csv`? Your score for this sub-question will depend on how many training data can be correctly predicted by your output program.

   Note:

   - Include your written answers of the above 7 sub-questions in your submitted pdf file.
   - Include your code as `gp_system.py` in your submitted zip file.
   - Please write down comments in your codes, indicating which sub-question the component is for.
   - Please make sure your submitted code is consistent with your answers. Otherwise we may consider it as potential plagiarism.