

Calculators may be used in this examination
provided they are not capable of being used
to store alphabetical information other than
hexadecimal numbers

UNIVERSITY OF BIRMINGHAM

School of Computer Science

MSc Introduction into Artificial Intelligence

Main Summer Examinations 2019

Time allowed: 1:30

[Answer all questions]

Note

Answer ALL questions. Each question will be marked out of 20. The paper will be marked out of 60, which will be rescaled to a mark out of 100.

Question 1 General Knowledge

- (a) Consider that you would like to install a set of cameras to monitor human activity in an amusement park. You will start with a small number of cameras, but would like to acquire more cameras in the future. The company that is providing you with the cameras offers a service where they can check the cameras' working order once every 4 months for a fixed yearly fee. If a camera breaks between these checkup points and you would like to have it fixed before the next checkup point, you will have to pay an additional fee which you wish to avoid. In addition, your cameras need to have their clocks synchronised to facilitate tracking a person if necessary.

Explain **two reasons** why firefly synchronisation would be an adequate algorithm for synchronising the cameras **in this specific context**.

[4 marks]

- (b) Explain the term “overfitting” in the context of machine learning and discuss how each of “regularization” and “model complexity” impact this phenomenon. Be sure to include the following:
- (i) What do the terms “regularization” and “model complexity” mean and how does the increase and decrease of each impact model overfitting.
 - (ii) How is regularization related to the magnitude of weights.
 - (iii) How is model complexity related to the presence of non-linear terms.
 - (iv) Provide an illustration of what a non-linear decision boundary looks like.

[8 marks]

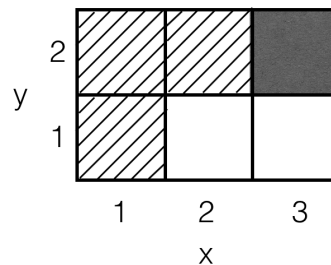
- (c) Recall the three following elements, which are common to several learning algorithms: a) Hypothesis function, b) Cost function, and c) the (partial) derivative of the cost function. Clearly explain what each of them are, how they relate to each other and their purpose in learning algorithms.

NOTE: You are **NOT** expected to provide the actual equations for any of these. You are only required to provide an overview of what they are.

[8 marks]

Question 2 Search and Optimisation

Assume that you are developing an algorithm to find the lowest cost path to move an army from a starting to a goal position in a strategy game. The field is organised as a grid, where the position whose coordinates (x, y) are $(3, 2)$ has an earth terrain; positions $(2, 1)$ and $(3, 1)$ have sand terrain; and positions $(1, 1)$, $(1, 2)$ and $(2, 2)$ have water terrain:



Each move can take the army one position up, down, left or right on the grid, so long as this does not move the army outside the grid. Independent of the type of terrain of the current position of the army, the cost of moving to an earth, sand and water position is 1, 3 and 10, respectively. For example, the cost of moving up from $(2, 1)$ is 10.

(a) Your army is in position $(1, 2)$ and you wish to reach goal position $(3, 1)$. An A* search algorithm is available to solve this problem, respecting the following rules:

- A state in the state space graph is identified by the (x, y) coordinates of the current position of the army, meaning that there are 6 possible states.
- The heuristic is the Manhattan distance between the state of the current node and the goal state, defined as follows:

$$\text{distance}((x_1, y_1), (x_2, y_2)) = |x_1 - x_2| + |y_1 - y_2|,$$

where (x_1, y_1) and (x_2, y_2) are the state of the current node and the goal state, respectively, and the operation $|k|$ represents the absolute value of k .

- Do not place children in the frontier if their corresponding state is already in the frontier or list of visited nodes with a smaller or equal $g(n)$, where $g(n)$ is the true cost from the origin to node n .
- Stop when you visit a node which contains the goal state.
- When deciding which node to visit, if there is a draw, choose to visit the node with the smallest x -coordinate first. If this still results in a draw, choose to visit the node with the smallest y -coordinate first. For example, if there is a draw between nodes whose states are $(1, 2)$ and $(2, 2)$, visit $(1, 2)$ first. If there is a draw between $(1, 2)$ and $(1, 1)$, visit $(1, 1)$ first.

Question 2 continued over the page

Write down the following information:

- Search tree produced by A*, indicating the following: $f(n)$, $g(n)$ and $h(n)$ values of each node n ; which nodes are visited nodes; and which nodes are in the frontier when the algorithm terminates.
- Sequence of nodes *visited* by A*. Note: you can identify a node through its state.
- Sequence of states that compose the path retrieved as a solution by A*.
- The cost of the solution retrieved by A*.

[8 marks]

- (b) Consider that you would like to solve this problem using Simulated Annealing instead of A*. The pseudocode of Simulated Annealing assuming maximisation is shown below. How would you change the pseudocode to minimise rather than maximise the objective function, so that you can apply it to this problem? Refer to the line number(s) that you would change, and how you would change it(them).

Simulated Annealing (assuming maximisation)

Input: initial temperature T_i , minimum temperature T_f

1. current_solution = generate initial solution randomly
2. $T = T_i$
3. Repeat until a minimum temperature T_f is reached or until the current solution “stops changing”:
 - 3.1 generate neighbour solutions (differ from current solution by a single element)
 - 3.2 rand_neighbour = get random neighbour of current_solution
 - 3.3 If $\text{quality}(\text{rand_neighbour}) \leq \text{quality}(\text{current_solution})$ {
 - 3.3.1 With probability $e^{(\text{quality}(\text{rand_neighbour}) - \text{quality}(\text{current_solution}))/T}$,
 - 3.3.1.1 current_solution = rand_neighbour
 - 3.4 Else current_solution = rand_neighbour
 - 3.5 $T = \text{schedule}(T)$

[6 marks]

Question 2 continued over the page

- (c) Consider that you would like to use Hill Climbing for a variation of this problem where your grid is much larger (with size $m \times m$, where $m > 100$) and you would like to move from $(1, m)$ to $(m, 1)$. Moves are forbidden if they take your army outside the grid or into any of the positions known to be well defended by the enemy. Your implementation of Hill Climbing is prepared to deal with minimisation problems, and your problem formulation has an objective function corresponding to the sum of the costs of the moves in your candidate solution, i.e.:

$$objective(solution) = \sum_{move \in solution} cost(move).$$

You decide to deal with the constraint related to forbidden moves by changing the objective function. For feasible solutions, the objective function is calculated as above. However, for infeasible solutions, you set the objective function to the value $C = 100$, i.e.:

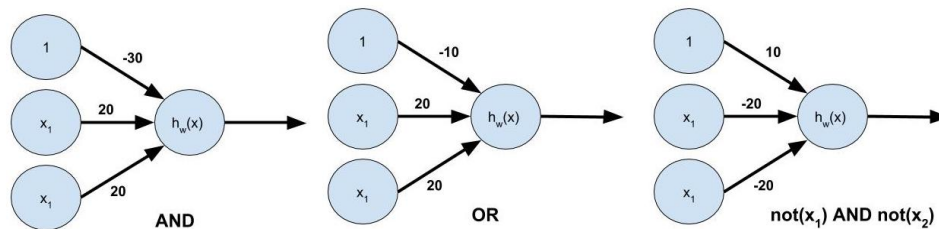
$$objective(infeasible_solution) = C.$$

Explain two problems of using this strategy to deal with the constraint.

[6 marks]

Question 3 Machine Learning

- (a) Describe the what the Backpropagation and Gradient Decent algorithms do with specific emphasis on their relationship. **[4 marks]**
- (b) Consider the following Neural Networks each of which perform the associated logical operation shown in the Truth Table.



x_1	x_2	$x_1 \text{ OR } x_2$	$x_1 \text{ AND } x_2$	$\text{not}(x_1) \text{ AND } \text{not}(x_2)$	$x_1 \text{ XNOR } x_2$
0	0	0	0	1	1
0	1	1	0	0	0
1	0	1	0	0	0
1	1	1	1	0	1

The activation function of the above Neural Network is the sigmoid function given by the equation $\frac{1}{1+e^{-x}}$. You may assume that:

$$\text{sigmoid}(x) \approx 0; \text{ for } x \leq -4$$

$$\text{sigmoid}(x) \approx 1; \text{ for } x \geq +4$$

Create a neural network to perform the non-linear separable operation XNOR whose Truth Table is given above. Draw a table to illustrate the input and output for each of your nodes that clearly shows why your networks performs this operation.

[8 marks]

- (c) Consider the scenario wherein we wish to train a model to drive a virtual car in a virtual world. Describe how you would set up a reinforced learning algorithm to learn this objective given that the car has sensors providing its distance from various objects around it and has control over the the angle of the steering, the accelerator and brakes. Can this problem remain a reinforced learning problem if we were interested in training a model to control a car in the real world? Why? How would you modify it? **[8 marks]**

This page intentionally left blank.

Do not complete the attendance slip, fill in the front of the answer book or turn over the question paper until you are told to do so

Important Reminders

- Coats/outwear should be placed in the designated area.
- Unauthorised materials (e.g. notes or Tippex) must be placed in the designated area.
- Check that you do not have any unauthorised materials with you (e.g. in your pockets, pencil case).
- Mobile phones and smart watches must be switched off and placed in the designated area or under your desk. They must not be left on your person or in your pockets.
- You are not permitted to use a mobile phone as a clock. If you have difficulty seeing a clock, please alert an Invigilator.
- You are not permitted to have writing on your hand, arm or other body part.
- Check that you do not have writing on your hand, arm or other body part – if you do, you must inform an Invigilator immediately
- Alert an Invigilator immediately if you find any unauthorised item upon you during the examination.

Any students found with non-permitted items upon their person during the examination, or who fail to comply with Examination rules may be subject to Student Conduct procedures.