THE UNIVERSITY OF WESTERN AUSTRALIA

Computer Science and Software Engineering

SEMESTER 1, 2015 EXAMINATIONS

CITS3001
Algorithms, Agents and Artificial Intelligence

FAMILY NAME:_____ GIVEN NAMES: _____

STUDENT ID: | | | | | | | | | SIGNATURE:_____

This Paper Contains: **6** pages **(including title page)**
Time allowed: **2:10 hours (including reading time)**

INSTRUCTIONS:

Answer all questions. Each question is worth 10 marks. The total for the paper is 100.

Most questions require only brief answers: point form answers are fine where appropriate.

PLEASE NOTE

Examination candidates may only bring authorised materials into the examination room. If a supervisor finds, during the examination, that you have unauthorised material, in whatever form, in the vicinity of your desk or on your person, whether in the examination room or the toilets or en route to/from the toilets, the matter will be reported to the head of school and disciplinary action will normally be taken against you. This action may result in your being deprived of any credit for this examination or even, in some cases, for the whole unit. This will apply regardless of whether the material has been used at the time it is found.

Therefore, any candidate who has brought any unauthorised material whatsoever into the examination room should declare it to the supervisor immediately. Candidates who are uncertain whether any material is authorised should ask the supervisor for clarification.

Supervisors Only - Student left at:

This page has been left intentionally blank

Q1. String algorithms

(a) Explain how the *Rabin-Karp algorithm* improves the naïve pattern-matching algorithm. **4 marks**

(b) Consider this example:

```
T = the_one_phone
P = one_phone
```

Illustrate how the *bad-character* and *good suffix* heuristic calculates the shift advance in the *Boyer-Moore algorithm*. **6 marks**

Q2. Optimisation algorithms

(a) Define the *activity selection* problem, and list **four** reasonable greedy rules. **3 marks**

(b) Using this example {[6,9), [1,10), [2,4), [1,7), [5,6), [8,11), [9,11)}, illustrate how sometimes a greedy rule can lead to the optimal solution. **4 marks**

(c) Describe how simulated annealing helps to avoid local optima. **3 marks**

Q3. Uninformed search

(a) Describe the difference between *breadth-first search* and *depth-first search* in selecting which node to expand. **2 marks**

(b) Explain why breadth-first search is *complete* and *optimal*, whereas depth-first not. **3 marks**

(c) What is the motivating principle behind *depth-limited search*? **2 marks**

(d) Describe **three** problem features that can cause problems for *bidirectional search*. **3 marks**

Q4. Informed search

(a) What is the difference between *informed search* and *uninformed search*? **2 marks**

(b) Describe how *A** uses heuristics to guide its search procedure. **3 marks**

(c) Take the game tic-tac-toe as an example, explain and illustrate what an admissible heuristics for A* is. **3 marks**

(d) How to come up with a heuristic to ensure A* to be optimal*?* **2 marks**

Q5. Game-playing

(a) Describe the **three** usual approaches to dealing with incompleteness and non-determinism. **3 marks**

(b) Briefly explain how *quiescence search* and *horizon problem* related the level of *look-ahead* in a game-playing AI? **2 marks**

(c) Briefly explain how *minimax* works in a game-playing AI. **3 marks**

(d) What's the role of *alpha-beta pruning* in minimax? **2 marks**

Q6. Sequential decision problems (SDPs)

(a) What is a *policy* in the context of an SDP? **2 marks**

(b) Given a policy and a transitional model, describe the *value determination* and *action determination* process. **4 marks**

(c) Describe the operation of the *policy iteration* algorithm for solving SDPs. **4 marks**

Q7. Learning agents

(a) What are the **four** main connections between the four main components in a learning agent? **4 marks**

(b) Define and contrast *Supervised Learning* and *Unsupervised Learning*. **4 marks**

(c) What is *inductive learning*? **2 marks**

Q8. Reinforcement learning

(a) What is the difference between *passive learning* and *active learning*? **2 marks**

(b) Describe the operational behaviour of *Adaptive Dynamic Programing*. **3 marks**

(c) Describe the operational behaviour of *temporal-difference learning*. **3 marks**

(d) What is *Q-Learning*? **2 marks**

Q9. Logical agents

(a) What are the three main steps in proof by *resolution*? **3 marks**

(b) How to turn a *conditional* sentence ($S_1 \rightarrow S_2$) or a *bi-conditional* ($S_1 \leftrightarrow S_2$) into conjunctive normal form? **2 marks**

(c) Describe and illustrate with an example the main way in which *first-order logic* is more expressive than propositional logic. **2 marks**

(d) Describe and illustrate with an example what it means to *unify* two sentences in first-order logic. **3 marks**

Q10. Planning and acting

(a) What is the main difference of *factored planning* as compared to traditional *search algorithms*? **2 marks**

(b) Describe briefly how a *partial-order planner* works. **4 marks**

(c) What is the role of *topological sorting* in partial-order planning? **2 marks**

(d) What are the **two** remedies when step S_k *clobbers* step S_i? **2 marks**

END OF PAPER