CSCE476/876 **Fall 2019**

Homework 4

Assigned on: Monday September 30^{th}, 2019.

Due: Monday, October 7^{th}, 2019.
Except for the programming questions (i.e., Exercices 1 and 7), which must be submitted with `webhandin` as problem#.lisp, you may turn in your homework on paper or type it and submit it to `webhandin`.

Value: 90 points for ugrads and 95 points for grads.

---

1 Implementing a simple-reflex agent. Total: 20 points

- Write in Common Lisp a function that 'models' the simple-reflex agent for the vacuum-cleaner problem in an environment with two locations, as summarized on page 5 of the Intructor's notes #4. The function should take as input the percepts of the agent as location of the agent and status of the room.
- Write a Common Lisp function that takes any of the 8 possible states of the vacuum-cleamer of Figure 3.3 of AIMA and runs the simple-reflect agent until the goal is reached.
- Design a performance measure that penalizes the agent for each step and each suck action. Record the agent performance for each one of the above 8 possible states.

2 AIMA, Exercise 3.6, Page 113. Total 10/15 points

- a: for ugrads and grads. 5 points
- b: for ugrads and grads. 5 points
- d: grads (bonus for ugrads). 5 points

3 AIMA, Exercise 3.15, Page 116. Total: 10 points

4 Evaluation function. Total: 6 points

Adapted from AIMA, Edition 1.

With $g(n)$ being the path length,

1. Suppose that we run a greedy search algorithm with $h(n) = -g(n)$. What sort of search will the greedy search emulate?
   Explain. 3 points
2. Suppose that we run a search algorithm with $h(n) = g(n)$. What sort of search will the greedy search emulate?
   Explain. 3 points

5 AIMA, Exercise 3.21, Page 117. Total: 9 points

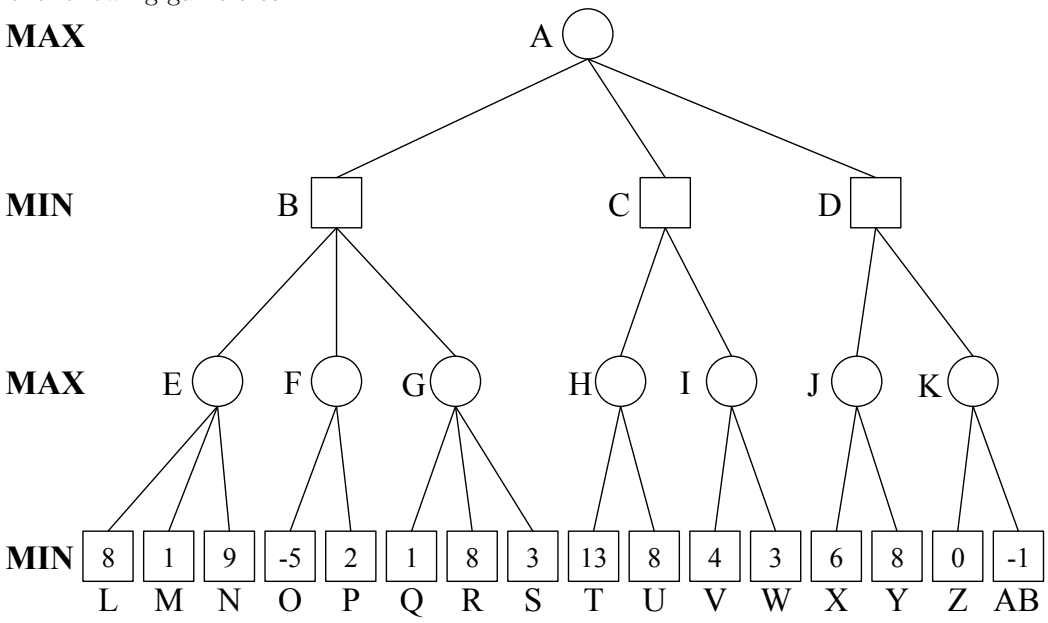
6 AIMA, Exercise 3.23, Page 118. Total: 10 points

7 AIMA, Exercise 3.30, Page 119. Total: 15 points

- Question a — 10 points
- Question b — 5 points
- Question c: Optional challenge — 15 bonus points
- Question d: Optional challenge — 30 bonus points

8 Adverserial Search — 5 points

Consider the following game tree:



1. Compute the minimax decision. Show your answer by writing the values at the appropriate nodes in the above tree. 4 points

2. What move should Max choose? 1 point

9 Alpha-beta Pruning — 5 points

Using the *alpha-beta pruning method*, with standard left-to-right evaluation of nodes, show what nodes are *not* examined by alpha-beta.