Homework 4

Assigned on: Monday September 30th, 2019.

Due: Monday, October 7th, 2019.

Except for the programming questions (i.e., Exercises 1 and 7), which must be submitted with webhandin as probem#.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 Instructor’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-cleaner 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:

MAX

MIN

MAX

MIN

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 \textit{alpha-beta pruning method}, with standard left-to-right evaluation of nodes, show what nodes are \textit{not} examined by alpha-beta.

\begin{center}
\begin{tikzpicture}
  \node (A) at (0,0) [circle] {A};
  \node (B) at (-2,-1) [rectangle] {B};
  \node (C) at (2,-1) [rectangle] {C};
  \node (D) at (2,-3) [rectangle] {D};
  \node (E) at (-2,-3) [circle] {E};
  \node (F) at (-2,-4) [circle] {F};
  \node (G) at (0,-4) [circle] {G};
  \node (H) at (2,-4) [circle] {H};
  \node (I) at (2,-5) [circle] {I};
  \node (J) at (2,-6) [circle] {J};
  \node (K) at (2,-7) [circle] {K};
  \node (L) at (-2,-7) [rectangle] {L};
  \node (M) at (-2,-8) [rectangle] {M};
  \node (N) at (-2,-9) [rectangle] {N};
  \node (O) at (-2,-10) [rectangle] {O};
  \node (P) at (0,-10) [rectangle] {P};
  \node (Q) at (2,-10) [rectangle] {Q};
  \node (R) at (0,-11) [rectangle] {R};
  \node (S) at (2,-11) [rectangle] {S};
  \node (T) at (0,-12) [rectangle] {T};
  \node (U) at (2,-12) [rectangle] {U};
  \node (V) at (2,-13) [rectangle] {V};
  \node (W) at (2,-14) [rectangle] {W};
  \node (X) at (2,-15) [rectangle] {X};
  \node (Y) at (2,-16) [rectangle] {Y};
  \node (Z) at (2,-17) [rectangle] {Z};
  \node (AB) at (2,-18) [rectangle] {AB};

  \draw (A) -- (B);
  \draw (A) -- (C);
  \draw (B) -- (E);
  \draw (B) -- (F);
  \draw (B) -- (G);
  \draw (C) -- (H);
  \draw (C) -- (I);
  \draw (C) -- (J);
  \draw (C) -- (K);
  \draw (D) -- (L);
  \draw (D) -- (M);
  \draw (D) -- (N);
  \draw (D) -- (O);
  \draw (D) -- (P);
  \draw (D) -- (Q);
  \draw (D) -- (R);
  \draw (D) -- (S);
  \draw (D) -- (T);
  \draw (D) -- (U);
  \draw (D) -- (V);
  \draw (D) -- (W);
  \draw (D) -- (X);
  \draw (D) -- (Y);
  \draw (D) -- (Z);
  \draw (D) -- (AB);

  \draw (A) -- (E);
  \draw (A) -- (F);
  \draw (A) -- (G);
  \draw (A) -- (H);
  \draw (A) -- (I);
  \draw (A) -- (J);
  \draw (A) -- (K);
  \draw (A) -- (L);
  \draw (A) -- (M);
  \draw (A) -- (N);
  \draw (A) -- (O);
  \draw (A) -- (P);
  \draw (A) -- (Q);
  \draw (A) -- (R);
  \draw (A) -- (S);
  \draw (A) -- (T);
  \draw (A) -- (U);
  \draw (A) -- (V);
  \draw (A) -- (W);
  \draw (A) -- (X);
  \draw (A) -- (Y);
  \draw (A) -- (Z);
  \draw (A) -- (AB);
\end{tikzpicture}
\end{center}