Linköpings Universitet
Institutionen för Datavetenskap
Patrick Doherty

TypTentamen
TDDC17 Artificial Intelligence
xx monthl 200x kl. xx-xx

Points:

The exam consists of exercises worth 31 points.
To pass the exam you need 16 points.

Auxiliary help items:

Hand calculators.

Directions:

You can answer the questions in English or Swedish.
Use notations and methods that have been discussed in the course.
In particular, use the definitions, notations and methods in appendices 1-2.
Make reasonable assumptions when an exercise has been under-specified.
Begin each exercise on a new page.
Write only on one side of the paper.
Write clearly and concisely.

Jourhavande: xxxx. Will arrive for questions around xxxx.

1. The Situation Calculus is a logical formalism for representing action and change.
   (a) Give a concrete example of the Frame Problem in the Situation Calculus using logical formulas.**[2p]**
   (b) Describe a solution to the problem using the example.**[1p]**
   (c) Is it possible to use the Situation Calculus for planning? If so, explain how. If not, why not?**[1p]**
2. The following questions pertain to partial-order planning.
   (a) What is a partial-order planner? **[1p]**
   (b) Partial-order planning can be implemented as a search in the space of partial-order plans. The course book describes four components for each plan. Name and describe each of these components. **[4p]**
3. Convert the following formula (where x, y and z are variables and i is a constant) into clause form using the steps in Appendix 1. **[2p]**

$$\forall x([A(x) \wedge B(x)] \Rightarrow [C(x, \mathsf{i}) \wedge \exists y(\exists z[C(y, z)] \Rightarrow D(x, y))]) \vee \forall x(E(x))$$

4. Consider the following theory (where x, y and z are variables and $\mathsf{history}$, $\mathsf{lottery}$ and john are constants):

$$\forall x([Pass(x, \mathsf{history}) \wedge Win(x, \mathsf{lottery})] \Rightarrow Happy(x)) \tag{1}$$

$$\forall x \forall y([Study(x) \vee Lucky(x)] \Rightarrow Pass(x, y)) \tag{2}$$

$$\neg Study(\mathsf{john}) \wedge Lucky(\mathsf{john}) \tag{3}$$

$$\forall x(Lucky(x) \Rightarrow Win(x, \mathsf{lottery})) \tag{4}$$

   (a) Convert formulas (1) - (4) into clause form. **[1p]**
   (b) Prove that $Happy(\mathsf{john})$ is a logical consequence of (1) - (4) using the resolution proof procedure. **[2p]**
   - Your answer should be structured using a resolution refutation tree (as used in the book).
   - Since the unifications are trivial, it suffices to simply show the binding lists at each resolution step.

5. The following question pertains to adverserial search. Consider the game tree in Figure 1 in which static scores are all from the first players point of view. (Triangle is maximizer. Upside-down triangle is minimizer.)

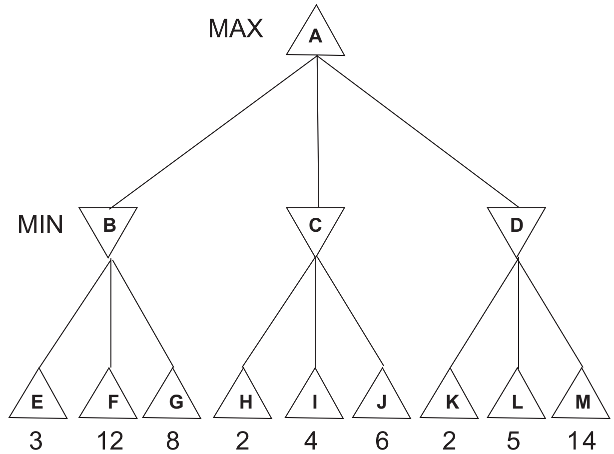

Figure 1: A Minimax Game Tree

(a) In the game tree in Figure 1, what nodes would not need to be examined using the alpha-beta procedure? Justify your answer by describing the α/β values in the nodes of the tree and why branches would be cutoff based on this.[**3p**]

6. The following question pertains to Decision Tree Learning. Use the definitions and data table in Figure 3, Appendix 2 to answer this question. Figure 2 shows a partial decision tree for the Table in Figure 3 with target attribute *PlayTennis*.

(a) What attribute should be tested in the box with the question mark on the right branch of the decision tree in Figure 2? Justify your answer by computing the information gain for the appropriate attributes in the Table in Figure 3. [**3p**]

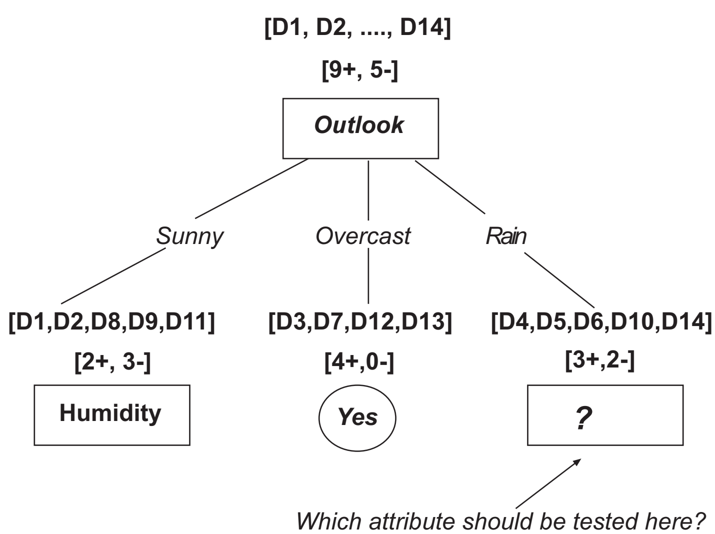

Figure 2: Partial Decision Tree for PlayTennis

7. A* search is the most widely-known form of best-first search. The following questions pertain to A* search:
   (a) Explain what an *admissible* heuristic function is using the notation and definitions in Appendix 2. **[1p]**
   (b) The 8-puzzle described in the book was one of the earliest heuristic search problems. Describe an admissible heuristic function for the 8-puzzle and justify why it is in fact an admissible heuristic. **[2p]**
   (c) Let $h(n)$ be the estimated cost of the cheapest path from a node n to the goal. Let $g(n)$ be the path cost from the start node n_0 to n. Let $f(n) = g(n) + h(n)$ be the estimated cost of the cheapest solution through n.

       Provide a general proof that A* using tree-search is optimal if $h(n)$ is admissible. If possible, use a diagram to structure the proof. **[2p]**

8. The following questions pertain to the course article by Newell and Simon entitled *Computer Science as an Empirical Enquiry: Symbols and Search.*.
   (a) What is a *physical symbol system* (PSS) and what does it consist of? **[1p]**
   (b) In the article, the authors discuss two notions central to the structures of expressions, symbols and objects which are used in PSS's: 1) designation and 2) interpretation. Provide a brief description of each of these notions in the context of a PSS. **[2p]**
   (c) What is the Physical Symbol System Hypothesis? **[1p]**
   (d) Do you agree with the hypothesis? Provide support for you opinion. **[1p]**
   (e) What is the Heuristic Search Hypothesis? **[1p]**

Appendix 1

Converting arbitrary wffs to clause form:

1. Eliminate implication signs.
2. Reduce scopes of negation signs.
3. Standardize variables within the scopes of quantifiers (Each quantifier should have its own unique variable).
4. Eliminate existential quantifiers. This may involve introduction of Skolem constants or functions.
5. Convert to prenex form by moving all remaining quantifiers to the front of the formula.
6. Put the matrix into conjunctive normal form. Two useful rules are:
   - $\omega_1 \vee (\omega_2 \wedge \omega_3) \equiv (\omega_1 \vee \omega_2) \wedge (\omega_1 \vee \omega_3)$
   - $\omega_1 \wedge (\omega_2 \vee \omega_3) \equiv (\omega_1 \wedge \omega_2) \vee (\omega_1 \wedge \omega_3)$
7. Eliminate universal quantifiers.
8. Eliminate $\wedge$ symbols.
9. Rename variables so that no variable symbol appears in more than one clause.

Appendix 2

Definition 1
Given a collection S, containing positive and negative examples of some target concept, the entropy of S relative to this boolean classification is

$$Entropy(S) \equiv -p_{\oplus} log_2 p_{\oplus} - p_{\ominus} log_2 p_{\ominus},$$

where $p_{\oplus}$ is the proportion of positive examples in S and $p_{\ominus}$ is the proportion of negative examples in S.

Definition 2
Given a collection S, containing positive and negative examples of some target concept, and an attribute A, the information gain, $Gain(S, A)$, of A relative to S is defined as

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{\mid S_v \mid}{\mid S \mid} Entropy(S_v),$$

where $values(A)$ is the set of all possible values for attribute A and S_v is the subset of S for which the attribute A has value v (i.e., $S_v = \{s \in S \mid A(s) = v\}$).

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Figure 3: Sample Table for Target Attribute PlayTennis

For help in converting from one logarithm base to another (if needed):

$$log_a\, x = \frac{log_b\, x}{log_b\, a}$$

$$ln\, x = 2.303\, log_{10}\, x$$