# 2007 Comprehensive Examination
# Artificial Intelligence

**1. Search.** (20 points) Assume we have a binary search tree with a maximum depth at level $d$. You can assume that the root node is level 0. Just give answers; no explanation necessary.

a. If there is a solution at depth $k<d$, what is the maximal number of nodes that will be examined by depth-first search, breadth-first search, and by iterative deepening (with depth starting at 0 and incrementing by 1 on each iteration)?

b. If there is a solution at depth $k<d$, what is the maximal amount of storage (in terms of nodes) required by depth-first search, breadth-first search, and by iterative deepening?

c. Suppose the tree is an adversary search tree with depth $d=2n$ and evaluation at depth $d$, what is a good asymptotic bound on the minimum number of nodes that must be searched with alpha-beta pruning?

d. What is the maximum amount of storage (in terms of nodes) required for alpha-beta pruning in this minimal case?

**2. Constraint Satisfaction Problems.** (20 points) Consider the following five algorithms for solving CSPs.

(a) Depth-first search with consistency checking and a fixed variable ordering.

(b) Depth-first search with forward checking and a fixed variable ordering.

(c) Depth-first search with forward checking and the most-constrained variable heuristic.

(d) Depth-first search with AC-3 and a fixed variable ordering.

(e) Depth-first search with AC-3 and the least-constrained variable heuristic.

Write down all pairs of algorithms where the first is better than the second, i.e. write down "$x \leq y$" if and only if algorithm $y$ is *guaranteed* to expand at least as many nodes of the search tree as $x$. No explanation necessary.

**3. Validity, Contingency, Unsatisfiability.** (20 points) For each of the following sentences, say whether it is valid, unsatisfiable, or contingent (neither valid nor unsatisfiable). You do not need to justify your answers.

(a) $\forall x.p(x) \Rightarrow p(x)$

(b) $p(x) \Rightarrow \forall x.p(x)$

(c) $\exists x.p(x) \Rightarrow p(x)$

(d) $p(x) \Rightarrow \exists x.p(x)$

(e) $\exists x.p(x) \Rightarrow \forall x.p(x)$

(f) $\forall x.p(x) \Rightarrow \exists x.p(x)$

(g) $\forall x.p(x) \Rightarrow \exists x.\neg p(x)$

(h) $\forall x.(p(x) \Rightarrow q(x)) \Rightarrow \exists x.(p(x) \wedge q(x))$

(i) $\forall x.(p(x) \Rightarrow q(x)) \wedge \neg\exists x.(p(x) \wedge q(x))$

(j) $(\exists x.p(x) \Rightarrow \forall x.p(x)) \vee (\forall x.p(x) \Rightarrow \forall x.q(x))$

**4. Resolution.** (20 points) Use the resolution method and the following premises to prove the conclusion shown below.

Premises:

a. $\forall x.\forall y.(p(x, y) \Rightarrow \exists z.q(x, y, z))$

b. $\forall x.(\exists y.r(x, y) \Rightarrow \neg\exists w.\exists z.q(x, w, z))$

c. $\forall x.\exists y.(p(x, y) \vee \forall z.q(x, y, z))$

Conclusion:

$\forall x.\exists y.\exists z.(\neg r(y, z) \wedge q(x, y, z))$

Note that this is a question about Resolution. You will get zero points (nil, nada, rien, zip, nothing) unless you prove it using the standard resolution procedure.

**5. Machine Learning.** (20 points)

(a) Consider a generative model for nonnegative real-valued inputs $x$ where each $x$ is sampled from the uniform distribution over the real interval $[0, \theta]$ and where $\theta$ is a real-valued parameter. Given a training set $\{x^1, x^2, \ldots, x^n\}$ with each $x^i$ is sampled i.i.d. from this generative distribution, give the maximum likelihood estimate for $\theta$.

(b) For a binary classification task, suppose you are given a training set with $n$ examples, and also a very large test set for evaluation. You train a logistic regression classifier for this task with the parameters estimated to maximize likelihood on the training set. For the trained classifier, let the misclassification error on the training set be $e_{\text{train}}$ and the misclassification error on the test set be $e_{\text{test}}$, and suppose the test set error $e_{\text{test}}$ is unacceptably high. You are given two choices: (A) Ask for more training data, or (B) Try a richer, non-linear classifier (e.g., try an SVM with the RBF kernel). For each of the following scenarios, which of the two choices should you investigate first. Give very, very, very brief explanations for your answers.

(i) $0 \approx e_{\text{train}} << e_{\text{test}}$

(ii) $e_{\text{train}} \approx e_{\text{test}} >> 0$