2006 Comprehensive Examination
Artificial Intelligence

1. Propositional Constraint Satisfaction. (20 points) In this question, we consider representing propositional satisfiability (SAT) problems as CSPs.

(a) Consider the following SAT Problem.

$$(\neg X_1 \vee X_2) \wedge (\neg X_2 \vee X_3) \wedge \ldots \wedge (\neg X_{n-1} \vee X_n)$$

How many solutions are there for this SAT problem as a function of *n*?

(1) 0
(2) 1
(3) n
(4) $n+1$
(5) n^2
(6) 2^n

(b) Suppose we apply backtracking to find *all* solutions to a SAT problem of the type given in (a) using iterated constraint satisfaction. (To find all solutions to a CSP, we continue searching after each solution is found until all possibilities are tried.) Assume that the variables are ordered $X_1, \ldots, X_n$ and *false* is ordered before *true*. How much time will it take to terminate the search?

(1) constant time
(2) linear in n
(3) quadratic in n
(4) exponential in n

(c) True or False: Every Horn-form SAT problem can be solved in time linear in the number of variables.

(d) True or False: Every tree-structured binary CSP with discrete, finite domains can be solved in time linear in the number of variables.

2. Logic. (20 points) Let Γ and Δ be sets of closed sentences in first-order logic, and let φ and ψ be individual closed sentences in first-order logic. State whether each of the following statements is true or false. No explanation is necessary.

(a) If $\Gamma \cap \Delta \models \varphi$, then $\Gamma \models \varphi$ and $\Delta \models \varphi$.

(b) If $\Gamma \cup \Delta \models \varphi$, then $\Gamma \models \varphi$ or $\Delta \models \varphi$.

(c) $\Delta \models (\varphi \Rightarrow \psi)$ if and only if $\Delta \cup \{\varphi\} \models \psi$.

(d) $\Delta \models \varphi$ or $\Delta \models \psi$ if and only if $\Delta \models (\varphi \vee \psi)$.

(e) If $\Delta \models \varphi$ and $\Delta \models \neg\psi$, then $\Delta \not\models (\varphi \Rightarrow \psi)$.

(f) If $\Delta \models \varphi$ and $\Delta \models \psi$, then $\Delta \models (\varphi \Rightarrow \psi)$.

(g) If $\Delta \models p(\tau)$ for some ground term τ, then $\Delta \not\models \forall x.\neg p(x)$.

(h) If $\Delta \models p(\tau)$ for every ground term τ, then $\Delta \models \forall x.p(x)$.

(i) If $\Delta \models \forall x.(p(x) \Rightarrow q(x))$, then $\Delta \models \exists x.(p(x) \wedge q(x))$.

(j) If $\Gamma \models (\varphi \Rightarrow \psi)$ and $\Delta \models (\psi \Rightarrow \varphi)$,then $\Gamma \cap \Delta \models (\varphi \Rightarrow \psi) \vee (\psi \Rightarrow \varphi)$.

3. 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. $\exists x.\forall y.\forall z.(r(y,z) \Leftrightarrow q(x,y,z))$

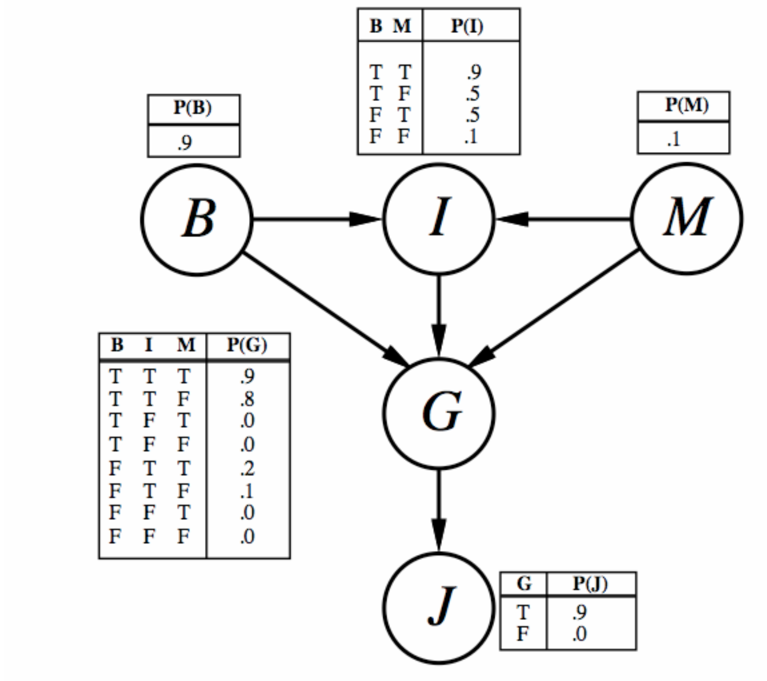
c. $\forall x.\exists y.(\neg p(x,y) \Rightarrow \forall z.q(x,y,z))$

Conclusion:

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

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.

4. Bayes Nets. (20 points) Consider the Bayes net shown below. Here, B means that a person broke the election law, I means that the person was indicted, M means that the prosecutor in politically motivated, G means that the defendant is found guilty, and J means that the defendant is jailed.



(a) Which, if any, of the following are asserted by the network structure (ignoring the CPTs for now)?

(1) $p(\text{B, I, M}) = p(\text{B})\, p(\text{I})\, p(\text{M})$
(2) $p(\text{J} \mid \text{G}) = p(\text{J} \mid \text{G, I})$
(3) $p(\text{M} \mid \text{G, B, I}) = p(\text{M} \mid \text{G, B, I, J})$

(b) Calculate the value of $p(\text{B, I, -M, G, J})$

(c) Calculate the probability that someone goes to jail given that the person broke the law, has been indicted, and faces a politically motivated prosecutor.

5. Learning. (20 points) In this question, we consider decision trees with numerical input attributes A_1 and A_2 and a Boolean output attribute Y. In such trees, the test at each internal node is an inequality of the form $A_i > c$ where c may be any number (to be chosen by the learning algorithm). The value at each leaf node is *true* or *false*. In a *test-once* tree, each attribute may be tested at most once on any path in the tree. In a *test-many* tree, each attribute may be tested more than once.

Suppose we are given the following training set.

A_1	A_2	Y
3	3	*false*
6	13	*true*
15	14	*true*
14	22	*false*

(a) Draw a test-once tree that classifies the examples correctly.

(b) Write down the information gain of your root test. Your answer may contain unevaluated logs.

(c) True or False: Every non-noisy training set can be correctly classified by a test-once decision tree.

(d) True or False: Every non-noisy training set can be correctly classified by a test-many decision tree.