CS-171, Intro to A.I. — Final Exam — Winter Quarter, 2014

NAME AND EMAIL ADDRESS: ______________________________

YOUR ID: ________ ID TO RIGHT: ________ ROW: _____ SEAT NO.: _____

The exam will begin on the next page. Please, do not turn the page until told.

When you are told to begin the exam, please check first to make sure that you have all 10 pages, as numbered 1-10 in the bottom-left corner of each page.

The exam is closed-notes, closed-book. No calculators, cell phones, electronics.

Please clear your desk entirely, except for pen, pencil, eraser, an optional blank piece of paper (for optional scratch pad use), and an optional water bottle.
<u>Please turn off all cell phones now.</u>

This page summarizes the points available for each question so you can plan your time.

1. (20 pts total, 5 pts each) Train/Test Sets, Accuracy, Overfitting.

2. (15 pts total, 5 pts each) Bayesian Networks.

3. (10 pts total) Mini-Max, Alpha-Beta Pruning.

4. (10 points total, 2 pts each) Constraint Satisfaction Problems.

5. (10 pts total, 2 pts each) Execute Tree Search.

6. (10 pts total) The Horned And Magical Unicorn.

7. (10 pts total, 1 pt each) ENGLISH TO FOPC CONVERSION.

8. Logic Concepts (6 pts total, 1 pt each).

9. Probability concepts and formulae (9 pts total, 1 pt each).

The Exam is printed on both sides to save trees! Work both sides of each page!

1. (20 pts total, 5 pts each) Train/Test Sets, Accuracy, Overfitting.

You are working on Face Recognition Problem, and you got the following result.

Table 1. Accuracies for each algorithm

	Nearest Neighbor	Decision Tree	Neural Network	Support Vector Machine
Accuracy on training data	70%	80%	85%	75%
Accuracy on testing data	65%	70%	60%	75%

1a. (5 pts) Which algorithm is the best algorithm (choose from 1-4 below)? ________(4)________.

1) Nearest Neighbor
2) Decision Tree
3) Neural Network
4) Support Vector Machine

See Section 18.2.

1b. (5 pts) Which algorithm is overfitting the most (choose from 1-4 below)?________(3)________.

1) Nearest Neighbor
2) Decision Tree
3) Neural Network
4) Support Vector Machine

See Section 18.3.5.

1c. (5 pts) Next you decided to use 3 fold cross validation method on your data. You split your data into three parts, data1, data2, and data3. You need to run your machine learning algorithm three times with different training/test data for 3 fold cross validation. Please select three training/testing data for each run from the below.

1) data1
2) data2
3) data3
4) data1 and data2
5) data1 and data3
6) data2 and data3
7) data1, data2, and data3

See Section 18.4.1 and Figure 18.8.

Run 1: Training Data ________(4)________. Test Data ________(3)________.

Run 2: Training Data ________(5)________. Test Data ________(2)________.

Run 3: Training Data ________(6)________. Test Data ________(1)________.

1d. (5 pts) After running 3 fold cross validation on you data, you got the following result.

Table 2 Cross validation results

	Run1	Run2	Run3
Accuracy on Training data	78%	80%	79%
Accuracy on Test data	76%	78%	77%

What is the cross validation accuracy for the result? ______77%______.

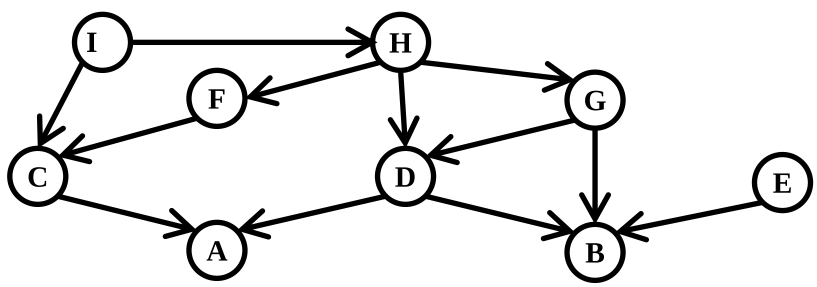
See Section 18.4.1 and Figure 18.8.

2. (15 pts total, 5 pts each) Bayesian Networks.

2a. (5 pts) Write down the factored conditional probability expression that corresponds to the graphical Bayesian Network shown.

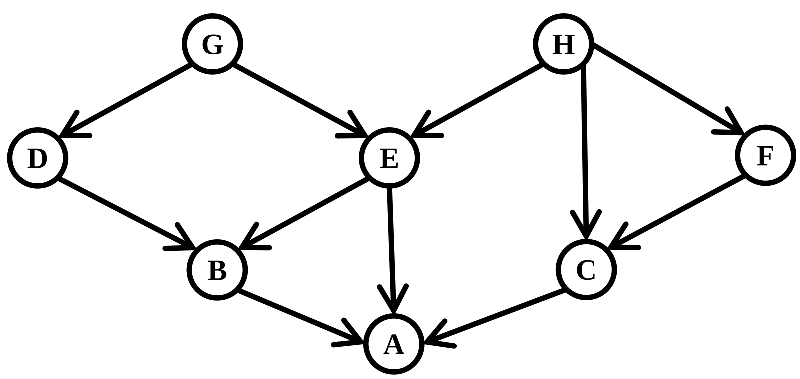
See Section 14.2.

P(A | C,D) .P(B | D,E,G) P(C | F,I) P(D | G,H) P(E) P(F | H) P(G | H) P(H | I) P(I)

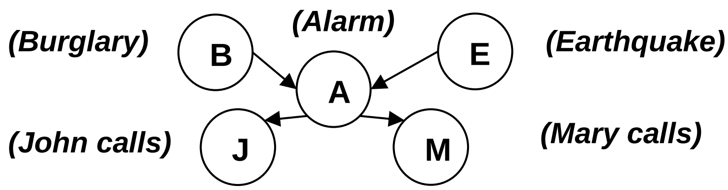


2b. (5 pts) Draw the Bayesian Network that corresponds to this conditional probability:

P(A | B,C,E) P(B | D,E) P(C | F,H) P(D | G) P(E| G,H) P(F | H) P(G) P(H)



2.c. (5 pts) Shown below is the Bayesian network corresponding to the Burglar Alarm problem, P(J | A) P(M | A) P(A | B, E) P(B) P(E).



P(E)
.002

A	P(M)
t	.70
f	.01

B	E	P(A)
t	t	.95
t	f	.94
f	t	.29
f	f	.001

P(B)
.001

A	P(J)
t	.90
f	.05

The probability tables show the probability that variable is True, e.g., P(M) means P(M=t). Write down an expression that will evaluate to P(j=t ∧ m=f ∧ a=f ∧ b = f ∧ e = t). Express your answer as a series of numbers (numerical probabilities) separated by multiplication symbols. You do not need to carry out the multiplication to produce a single number (probability). **SHOW YOUR WORK.**

P(j=t ∧ m=f ∧ a=f ∧ b = f ∧ e = t)
= P(j=t | a=f) * P(m=f | a = f) * P(a=f | b=f ∧ e=t) * P(b=f) * P(e=t)
= .05 * .99 * .71 * .999 * .002

3. (10 pts total) Mini-Max, Alpha-Beta Pruning. In the game tree below it is **Max**'s turn to move. At each leaf node is the estimated score of that resulting position as returned by the heuristic static evaluator.

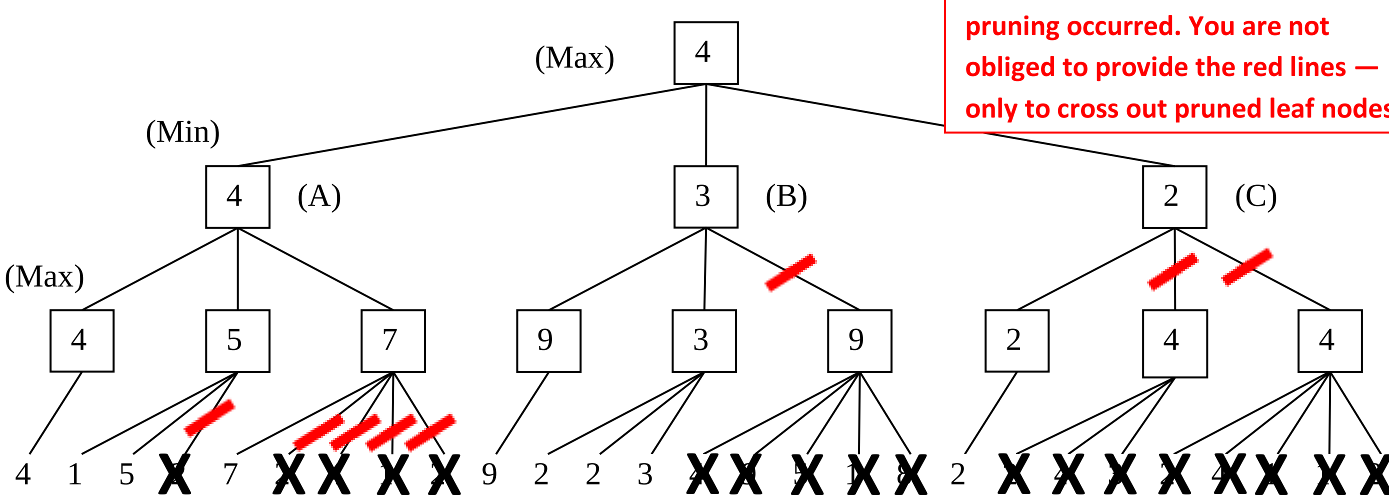
See Section 5.2-3.

(1) Perform Mini-Max search and label each branch node with its value.

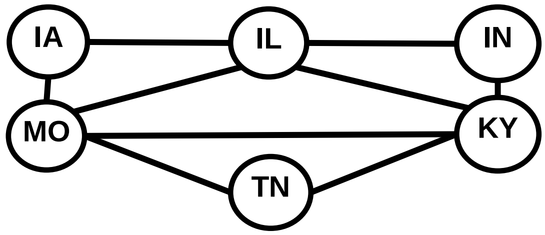
(2) Cross out each leaf node that would be pruned by alpha-beta pruning.

(3) What is Max's best move (A, B, or C)? A

Red lines indicate where in the tree pruning occurred. You are not obliged to provide the red lines — only to cross out pruned leaf nodes.



4. (10 points total, 2 pts each) Constraint Satisfaction Problems. See Chapter 6.



TN=Tennesse
IN=Indiana
IA=Iowa
IL=Illinois
MO=Missouri
KY=Kentucky

You are a map-coloring robot assigned to color this Midwest USA map. Adjacent regions must be colored a different color (R=Red, B=Blue, G=Green). The constraint graph is shown.

4a. (2pts total, -1 each wrong answer, but not negative) FORWARD CHECKING. Cross out all values that would be eliminated by Forward Checking, after variable TN has just been assigned value R as shown:

TN	IN	IA	IL	MO	KY
R	R G B	R G B	R G B	~~R~~ G B	~~R~~ G B

4b. (2pts total, -1 each wrong answer, but not negative) ARC CONSISTENCY. TN and MO have been assigned values, but no constraint propagation has been done. Cross out all values that would be eliminated by Arc Consistency (AC-3 in your book).

TN	IN	IA	IL	MO	KY
R	~~R~~ G ~~B~~	~~R~~ ~~G~~ B	R ~~G~~ ~~B~~	G	~~R~~ ~~G~~ B

4c. (2pts total, -1 each wrong answer, but not negative) MINIMUM-REMAINING-VALUES HEURISTIC. Consider the assignment below. IA is assigned and constraint propagation has been done. List all unassigned variables that might be selected by the Minimum-Remaining-Values (MRV) Heuristic:______**IL, MO**______.

TN	IN	IA	IL	MO	KY
R G B	R G B	G	R B	R B	R G B

4d. (2pts total, -1 each wrong answer, but not negative) DEGREE HEURISTIC. Consider the assignment below. (It is the same assignment as in problem 4c above.) IA is assigned and constraint propagation has been done. List all unassigned variables that might be selected by the Degree Heuristic:.______**KY**______.

TN	IN	IA	IL	MO	KY
R G B	R G B	G	R B	R B	R G B

4e. (2pts total) MIN-CONFLICTS HEURISTIC. Consider the complete but inconsistent assignment below. IA has just been selected to be assigned a new value during local search for a complete and consistent assignment. What new value would be chosen below for IA by the Min-Conflicts Heuristic?.______**R**______.

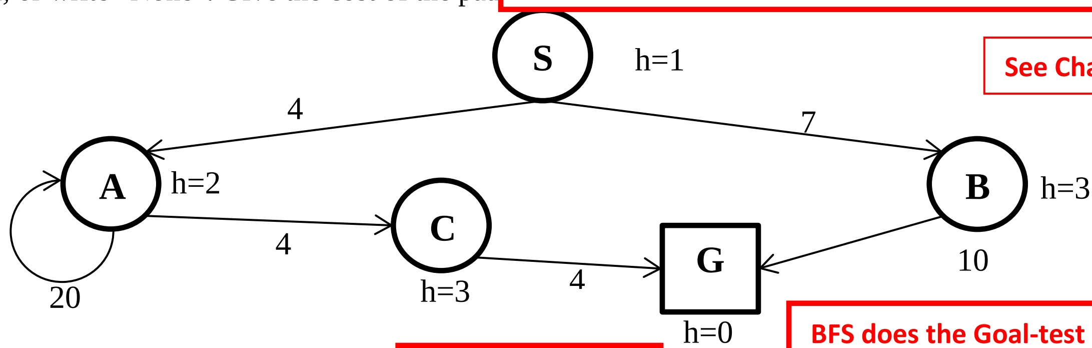
TN	IN	IA	IL	MO	KY
B	G	**?**	G	B	B

5. (10 pts total, 2 pts each) Execute Tree Search through this graph (i.e., do not remember visited nodes). Step costs are given next to each arc. Heuristic values are next to each node (as h=x). The successors of each node are indicated by the arrows out of th

For each search strategy, show the ord means that its children are generated), ending goal, or write "None". Give the cost of the pat

Please see the lecture slides for Uninformed Search, topic "When to do Goal-Test? When generated? When popped?" for clarification about exactly what to do in practical cases.

See Chapter 3.



5.a. BREADTH FIRST SEARCH:

See Section 3.4.1 and Fig. 3.11.

BFS does the Goal-test before the child is pushed onto the queue. The goal is found when B is expanded.

Order of node expansion: S A B G

Path found: S B G Cost of path found: 17

5.b. (*2* pts) DEPTH FIRST SEARCH:

See Section 3.4.3 and Fig. 3.17.

DFS can get caught in loops during Tree Search (= do not remember visited nodes).

Order of node expansion: S A A A A A A etc.

Path found: None Cost of path found: None

5.c. (*2* pts) UNIFORM COST SEARCH:

UCS does goaltest when node is popped off queue.

Order of node expansion: S A B C G

See Section 3.4.2 and Fig. 3.14.

Path found: S A C G Cost of path found: 12

5.d. (*2* pts) GREEDY (BEST-FIRST) SEARCH:

A always has lower h(=2) than any other node on queue.

Order of node expansion: S A A A A A A etc.

See Section 3.5.1 and Fig. 3.23.

Path found: None Cost of path found: None

5.e. (*2* pts) ITERATED DEEPENING SEARCH:

Order of node expansion: S S A B G

IDS does the Goal-test before the child is pushed onto the queue. The goal is found when B is expanded.

Path found: S B G

See Sections 3.4.4-5 and Figs. 3.18-19.

17

5.f. (*2* pts) A* SEARCH:

Order of node expansion: S A B C G

Path found: S A C G

See Section 3.5.2 and Figs. 3.24-25.

A* does goaltest when node is popped off queue.

Is the heuristic admissible (Yes or No)? Yes

Is the heuristic consistent (Yes or No)? Yes

6. (10 pts total) The Horned And Magical Unicorn.

If the unicorn is mythical, then it is immortal, but if it is not mythical, then it is a mortal mammal. If the unic … *ned. The unicorn is magical if it is horned. Prove th* …

Use these pr … ”):

Y = … al **M** = unicorn is a ma**M**mal

H = … cal

You have tr … ,” into (H ^ G), so the negated goal is:

(¬ I …

You have t … tional logic Knowledge Base (KB):

(¬ … ∨ M)

(R … I ∨ G)

Produce a r … l, that the unicorn is horned and magical. Repeatedly … lank space on a line, and the other clause in the second. … se in the third blank space, and insert it into the knowled … cannot produce ( ), then you have made a mistake. The shortest proof I know of is only six lines. It is OK to use more lines, if your proof is correct.

A bright and clever student has constructed a shorter proof than I was able to find:

Resolve (¬H ¬G) and (¬H G) to give (¬H)

Resolve (¬Y ¬R) and (Y M) to give (¬R M)

Resolve (¬R M) and (R H) to give (M H)

Resolve (M H) and (¬M H) to give (H)

Resolve (¬H) and (H) to give ()

See Section 7.5.2

Resolve (¬ H ∨ ¬ G) and (¬ H ∨ G) to give (¬ H).

Resolve (¬ M ∨ H) and (¬ H) to give (¬ M).

Resolve (Y ∨ M) and (¬ M) to give (Y).

Resolve (¬ Y ∨ ¬ R) and (Y) to give (¬ R).

Resolve (R ∨ H) and (¬ R) to give (H).

Resolve (¬ H) and (H) to give ( ).

Resolve ____________ and ____________ to give ____________.

It is OK if you used abbreviated CNF, i.e., (¬ H ¬ G) instead of (¬ H ∨ ¬ G). It is OK to omit the parentheses.

Resolve ____ and ____ to give ____.

Resolve ____

Other proofs are OK as long as they are correct. For example, another proof is:
Resolve (¬ H ∨ ¬ G) and (¬ H ∨ G) to give (¬ H).
Resolve (R ∨ H) and (¬ H) to give (R).
Resolve (¬ Y ∨ ¬ R) and (R) to give (¬ Y).
Resolve (Y ∨ M) and (¬ Y) to give (M).
Resolve (¬ M ∨ H) and (M) to give (H).
Resolve (H) and (¬ H) to give ( ).

See Chapter 12. The preferred encoding is given first, followed by acceptable syntactic variants.

7. (10 pts total, 1 pt each) ENGLISH TO FOPC CONVERSION.

For each English sentence below, write the FOPC sentence that best expresses its intended meaning.

7.a. (1pt) "All persons are mortal." [Use: Person(x), Mortal (x)]

$\forall x\ Person(x) \Rightarrow Mortal(x)$

$\forall x\ \neg Person(x) \lor Mortal(x)$

Common Mistakes:

$\forall x\ Person(x) \land Mortal(x)$

7.b. (1pt) "Fifi has a sister who is a cat." [Use: Sister(Fifi, x), Cat(x)]

$\exists x\ Sister(Fifi, x) \land Cat(x)$

Common Mistakes:

$\exists x\ Sister(Fifi, x) \Rightarrow Cat(x)$

7.c. (1pt) "For every food, there is a person who eats that food."
[Use: Food(x), Person(y), Eats(y, x)]

$\forall x\ \exists y\ Food(x) \Rightarrow [\ Person(y) \land Eats(y, x)\ ]$

$\forall x\ Food(x) \Rightarrow \exists y\ [\ Person(y) \land Eats(y, x)\ ]$

$\forall x\ \exists y\ \neg Food(x) \lor [\ Person(y) \land Eats(y, x)\ ]$

$\forall x\ \exists y\ [\ \neg Food(x) \lor Person(y)\ ] \land [\neg Food(x) \lor Eats(y, x)\ ]$

$\forall x\ \exists y\ [\ Food(x) \Rightarrow Person(y)\ ] \land [\ Food(x) \Rightarrow Eats(y, x)\ ]$

Common Mistakes:

$\forall x\ \exists y\ [\ Food(x) \land Person(y)\ ] \Rightarrow Eats(y, x)$

$\forall x\ \exists y\ Food(x) \land Person(y) \land Eats(y, x)$

7.d. (1pt) "Every person eats every food." [Use: Person (x), Food (y), Eats(x, y)]

$\forall x\ \forall y\ [\ Person(x) \land Food(y)\ ] \Rightarrow Eats(x, y)$

$\forall x\ \forall y\ \neg Person(x) \lor \neg Food(y) \lor Eats(x, y)$

$\forall x\ \forall y\ Person(x) \Rightarrow [\ Food(y) \Rightarrow Eats(x, y)\ ]$

$\forall x\ \forall y\ Person(x) \Rightarrow [\ \neg Food(y) \lor Eats(x, y)\ ]$

$\forall x\ \forall y\ \neg Person(x) \lor [\ Food(y) \Rightarrow Eats(x, y)\ ]$

Common Mistakes:

$\forall x\ \forall y\ Person(x) \Rightarrow [Food(y) \land Eats(x, y)\ ]$

$\forall x\ \forall y\ Person(x) \land Food(y) \land Eats(x, y)$

7.e. (2 pts) "All greedy kings are evil." [Use: King(x), Greedy(x), Evil(x)]

$\forall x\ [\ Greedy(x) \land King(x)\ ] \Rightarrow Evil(x)$

$\forall x\ \neg Greedy(x) \lor \neg King(x) \lor Evil(x)$

$\forall x\ Greedy(x) \Rightarrow [\ King(x) \Rightarrow Evil(x)\ ]$

Common Mistakes:

$\forall x\ Greedy(x) \land King(x) \land Evil(x)$

7.f. (1pt) "Everyone has a favorite food." [Use: Person(x), Food(y), Favorite(y, x)]

$\forall x\ \exists y\ Person(x) \Rightarrow [\ Food(y) \land Favorite(y, x)\ ]$

$\forall x$ Person(x) $\Rightarrow \exists y$ [Food(y) $\land$ Favorite(y, x)]
$\forall x \exists y \neg$Person(x) $\lor$ [Food(y) $\land$ Favorite(y, x)]
$\forall x \exists y$ [$\neg$Person(x) $\lor$ Food(y)] $\land$ [$\neg$Person(x) $\lor$ Favorite(y, x)]
$\forall x \exists y$ [Person(x) $\Rightarrow$ Food(y)] $\land$ [Person(x) $\Rightarrow$ Favorite(y, x)]

Common Mistakes:

$\forall x \exists y$ [Person(x) $\land$ Food(y)] $\Rightarrow$ Favorite(y, x)
$\forall x \exists y$ Person(x) $\land$ Food(y) $\land$ Favorite(y, x)

7.g. (1pt) "There is someone at UCI who is smart." [Use: Person(x), At(x, UCI), Smart(x)]

$\exists x$ Person(x) $\land$ At(x, UCI) $\land$ Smart(x)

Common Mistakes:

$\exists x$ [Person(x) $\land$ At(x, UCI)] $\Rightarrow$ Smart(x)

7.h. (1pt) "Everyone at UCI is smart." [Use: Person(x), At(x, UCI), Smart(x)]

$\forall x$ [Person(x) $\land$ At(x, UCI)] $\Rightarrow$ Smart(x)
$\forall x \neg$[Person(x) $\land$ At(x, UCI)] $\lor$ Smart(x)
$\forall x \neg$Person(x) $\lor \neg$At(x, UCI) $\lor$ Smart(x)

Common Mistakes:

$\forall x$ Person(x) $\land$ At(x, UCI) $\land$ Smart(x)
$\forall x$ Person(x) $\Rightarrow$ [At(x, UCI) $\land$ Smart(x)]

7.i. (1pt) "Every person eats some food." [Use: Person (x), Food (y), Eats(x, y)]

$\forall x \exists y$ Person(x) $\Rightarrow$ [Food(y) $\land$ Eats(x, y)]
$\forall x$ Person(x) $\Rightarrow \exists y$ [Food(y) $\land$ Eats(x, y)]
$\forall x \exists y \neg$Person(x) $\lor$ [Food(y) $\land$ Eats(x, y)]
$\forall x \exists y$ [$\neg$Person(x) $\lor$ Food(y)] $\land$ [$\neg$Person(x) $\lor$ Eats(x, y)]

Common Mistakes:

$\forall x \exists y$ [Person(x) $\land$ Food(y)] $\Rightarrow$ Eats(x, y)
$\forall x \exists y$ Person(x) $\land$ Food(y) $\land$ Eats(x, y)

7.j. (1pt) "Some person eats some food." [Use: Person (x), Food (y), Eats(x, y)]

$\exists x \exists y$ Person(x) $\land$ Food(y) $\land$ Eats(x, y)

Common Mistakes:

$\exists x \exists y$ [Person(x) $\land$ Food(y)] $\Rightarrow$ Eats(x, y)

See Chapter 7.

8. Logic Concepts (6 pts total, 1 pt each).

A .	Logic	A	Formal symbol system for representation and inference
C	Valid	B	The idea that a sentence follows logically from other sentences
G	Complete	C	True in every possible world
E	Conjunctive Normal Form	D	True in at least one possible world
F	Sound	E	A sentence expressed as a conjunction of clauses (disjuncts)
D	Satisfiable	F	Inference system derives only entailed sentences
B	Entailment	G	Inference system can derive any sentence that is entailed

9. Probability concepts and formulae (9 pts total, 1 pt each).

See Chapter 13.

A .	Probability Theory	A	Assigns each sentence a degree of belief ranging from 0 to 1
H	Conditional independence	B	Degree of belief accorded without any other information
G	Independence	C	Degree of belief accorded after some evidence is obtained
J	Product rule (chain rule)	D	Gives probability of all combinations of values of all variables
C	Conditional probability	E	Takes values from its domain with specified probabilities
B	Unconditional probability	F	A possible world is represented by variable/value pairs
F	Factored representation	G	$P(a \wedge b) = P(a)\, P(b)$
E	Random variable	H	$P(a \wedge b \mid c) = P(a \mid c)\, P(b \mid c)$
I	Bayes' rule	I	$P(a \mid b) = P(b \mid a)\, P(a) / P(b)$
D	Joint probability distribution	J	$P(a \wedge b \wedge c) = P(a \mid b \wedge c)\, P(b \mid c)\, P(c)$

****** THIS IS THE END OF THE FINAL EXAM ******