Midterm 2 Exam
CSCI 561: Artificial Intelligence

- DO NOT OPEN EXAM until given a start signal. Opening exam earlier will result in filing of a case with the office of student judicial affairs.
- STOP WRITING immediately when given the stop signal. Continuing to write after the stop signal will result in filing of a case with the office of student judicial affairs.
- Date: 11/3/2014 from 5:00pm to 6:20pm
- Credits/points for each question is indicated in the brackets [ ] before the question.
- No books or any other material and no devices are allowed.
- Your exam will be scanned and uploaded online.
- Do NOT write on the 2D barcode. You could lose all the points for that page!
- Do NOT write within a 1" margin of the edges of the page. You could lose points if the scanner cuts off the margins slightly.
- Use back side as scratch paper if needed. It will not be scanned. Do not write answers on the back side.
- Write down name and student ID on first sheet.
- NO questions during exam. If something is unclear to you, write that in your exam.
- Be brief: a few words are often enough if they are precise and use the correct vocabulary studied in class.
- When finished raise completed exam sheets until approached by proctor. Do not leave your seat until you are instructed to do so.
- Adhere to the Academic Integrity code.

Student ID: ____

Last Name: ____

First Name: ____

USC email: ____

	Problem	Max score (%)
1.	General AI knowledge	15
2.	Propositional logic	25
3.	First-order logic	30
4.	Fuzzy logic	20
5.	Knowledge engineering	10
	Total score	100%

1. [15%] True/False and Fill in the blank Questions

For 1.1) - 1.10), write T if the statement is always and unconditionally true, and write F otherwise. For 1.11) - 1.15), write down your answer below each question:

1.1)_**T**__ In FOL, constant symbols refer to objects, while predicate symbols refer to relations.

1.2)_**F**__ A sentence in Horn form is usually a disjunction of Horn clauses.

1.3)_ **F** __ ∀ is often used with ∧

1.4)_ **F** __ Fuzzy logic can only model linear functions.

1.5)_ **F** __ Promotion and demotion in planning are used to guide the choice of which new operator (step) should next be added to the plan.

1.6)__ **T** _The truth table method of inference is complete for propositional logic.

1.7)_ **F** __ Not every sentence of propositional logic can be converted to CNF.

1.8)_ **F** __ The Frame Problem is the problem of how to find an elegant way to handle non-change.

1.9)_ **F** __ Propositional logic has quantifiers ∀ and ∃.

1.10) In STRIPS, a plan where every precondition is achieved is __**COMPLETE**____.

1.11) Given that whenever KB $\vdash_i \alpha$ then KB $\models \alpha$ is true, an inference algorithm is said to be _**SOUND**__.

1.12) Generalized Modus Ponens requires sentences to be in ____**HORN**_____ form.

1.13) In plans, $A \xrightarrow{c} B$ means A __**ACHIEVES PRECONDITION c FOR**______ B. (OK if say achieves c for B)

1.14) In logic, __**SEMANTICS**______ define the meaning of sentences; i.e. define truth of a sentence in a world.

1.15) Prolog does inference using ____**BACKWARD**______ chaining.

2. [25%] Propositional Logic

2.1. [6%] CNF

Convert the following propositional sentence into CNF. Your answer must be simplified as much as possible and must exactly match the CNF form. You will lose points for any part of your final converted sentence that is not CNF.

$$\neg[((A \wedge B) \vee (D \wedge E)) \Rightarrow ((Q \Rightarrow R) \vee \neg S)]$$

Using $A \Rightarrow B$ is equivalent to $(\neg A \vee B)$, we get $\neg(\neg A \vee B)$ which is $A \wedge \neg B$

$$((A \wedge B) \vee (D \wedge E)) \wedge \neg ((Q \Rightarrow R) \vee \neg S)$$
$$((A \wedge B) \vee D) \wedge ((A \wedge B) \vee E)) \wedge (\neg(Q \Rightarrow R) \wedge S)$$
$$(D \vee A) \wedge (D \vee B) \wedge (E \vee A) \wedge (E \vee B) \wedge \neg(\neg Q \vee R) \wedge S$$
$$(D \vee A) \wedge (D \vee B) \wedge (E \vee A) \wedge (E \vee B) \wedge (Q \wedge \neg R) \wedge S$$

There can also be longer derivation where students will eliminate →, then move ¬ inwards using de Morgan, then eliminate double negative, distribute ∧ over ? and get the correct result.

-2 for any member that is not fully in CNF, e.g., ((A^B) v D) needs to be further developed into (A v D)^(B v D) using distributivity rule

several answers exist given commutativity of operators but they all should be strictly equivalent to the given final answer

2.2 [1%] Horn Clause

Is the following formula in Horn form? Justify your answer.

Rich V Happy V ¬ Unemployed

No. This formula contains disjunction with more than one positive subclause.

2.3. [18%] Inference

Consider the following knowledge base that describes when a car should brake:

PersonInFrontOfCar ⇒ Brake	***(1)***
((YellowLight ∧ Policeman) ∧ (¬Slippery)) ⇒ Brake	***(2)***
Policecar ⇒ Policeman	***(3)***
Snow ⇒ Slippery	***(4)***
Slippery ⇒ ¬Dry	***(5)***
RedLight ⇒ Brake	***(6)***
Winter ⇒ Snow	***(7)***

Given the following observations:

YellowLight ∧ ¬RedLight ∧ Dry ∧ Policecar ∧ ¬PersonInFrontOfCar (8)

Using the various propositional logic inference rules studied in class, show how each of the following conclusions can be inferred: In each case, mention which inference rule is used [1%], and to which sentence(s) it was applied [1%]:

[2%] Policecar	(9)	_____And elimination (AE) on 8_____________
[2%] Policeman	(10)	_____modus ponens (MP) on 3 & 9_________
[2%] Dry	(11)	______AE on 8____________________________
[2%] ¬Slippery	(12)	______modus tollens (MT) on 5 & 11_________
[2%] YellowLight	(13)	_______AE on 8______________________
[2%] YellowLight ∧ Policeman ∧ ¬Slippery	(14)	__and introduction (AI) on 10, 12 & 13___
[2%] Brake	(15)	____MP on 2 & 14_________________________
[2%] ¬Snow	(16)	_____MT on 4 & 12_________________
[2%] ¬Winter	(17)	______MT on 7 & 16______________

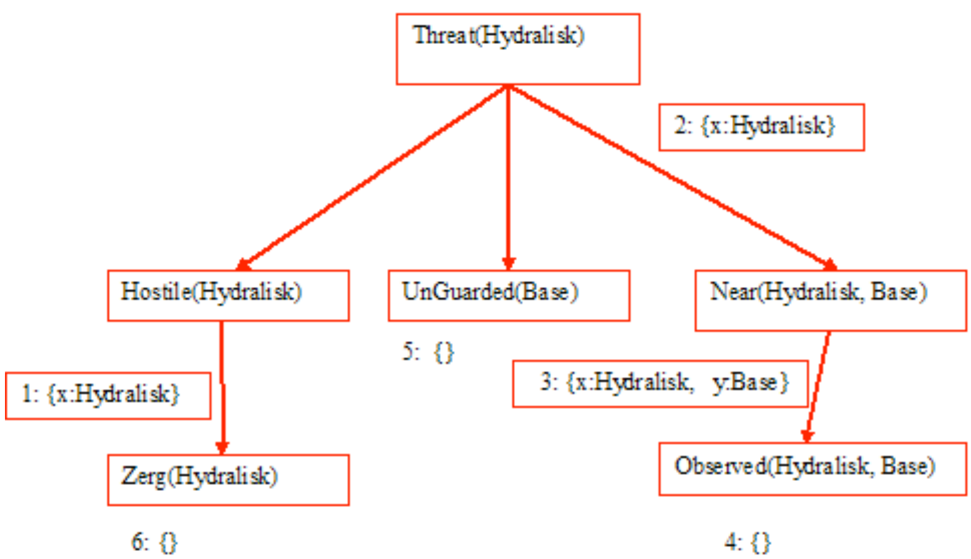
3. [30%] First Order Logic

3.1. [10%] Backward Chaining

Consider the following six sentences that are added to a knowledge base in turn. Zerg, Hostile, Guarded, Threat and Near are predicates, Hydralisk and Base are constants, x and y are variables. Assume all sentences are implicitly universally quantified over all variables.

1. ***Zerg (x) → Hostile (x)***
2. ***Hostile (x) ∧ UnGuarded (Base) ∧ Near (x, Base) → Threat (x)***
3. ***Observed (x, y) → Near (x, y)***
4. ***Observed (Hydralisk, Base)***
5. ***UnGuarded (Base)***
6. ***Zerg (Hydralisk)***

Given the KB above, show how **backward chaining with GMP** can be used to infer whether Hydralisk is a Threat (i.e., ***Threat(Hydralisk)***). Draw a backward-chaining inference tree. Be sure to show (1) a backward-chaining inference tree as studied in class, (2) all the substitutions used in unification at each stage, as relevant. You will lose marks for any missing node or substitution. You will get 0 of you use any other method than backward chaining with GMP.



0 if using any other rule than MP anywhere in answer or not using backward chaining
-1 each missing substitution
if proof is not complete, 1 point for each node + substitution that are correct in partial proof

3.2. [10%] Logic Sentences

Consider the following predicates:

I(x): x is an integer
E(x): x is even
O(x): x is odd
P(x): x is prime

Using the above predicates, translate the following English sentences into logic expressions:

1. [2%] There exists an even integer.

$\exists x \quad I(x) \wedge E(x)$

2. [2%] Every integer is even or odd.

$\forall x \quad I(x) \Rightarrow E(x) \vee O(x)$

3. [2%] Some odd integer numbers are prime.

$\exists x \quad I(x) \wedge O(x) \wedge P(x)$

4. [2%] The only even prime integer is 2.

$\forall x \quad I(x) \wedge E(x) \wedge P(x) \Rightarrow (x=2)$

5. [2%] Not all integers are odd.

$\exists x \quad I(x) \wedge \sim O(x)$
or
$\sim\forall x \quad I(x) \Rightarrow O(x)$

3.3. [10%] Resolution

Use resolution and a proof by contradiction to prove **D(Bob)** from the following knowledge base:

~A(x) v B(x)

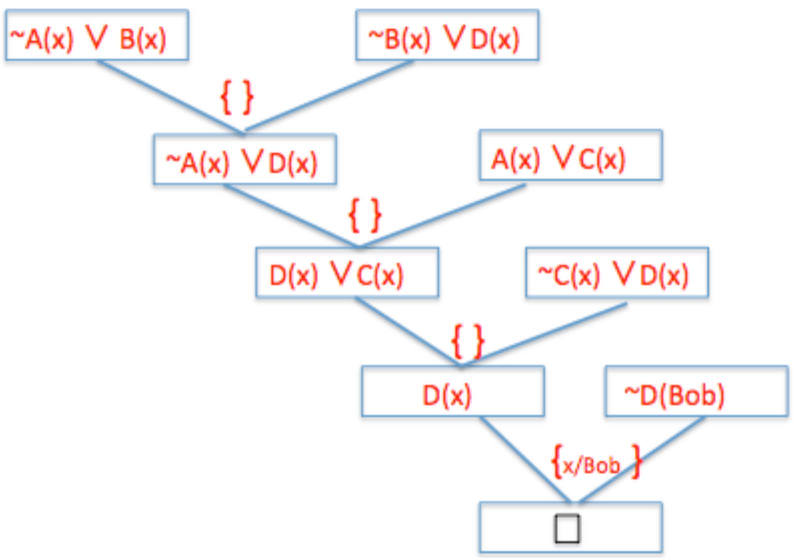
A(x) v C(x)

~B(x) v D(x)

~C(x) v D(x)

Please show the complete resolution proof, including all substitutions used (you will lose points for any missing step or substitution). You will get 0 if you use any other method than resolution and proof by contradiction.

To prove D(Bob), we first add ~D(Bob) to the knowledge base.



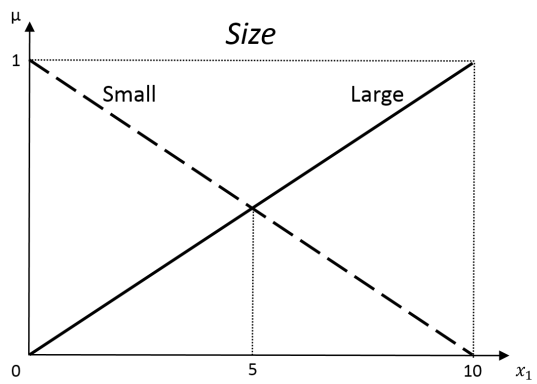
0 if any inference rule other than resolution used anywhere in proof, or not proof by contradiction

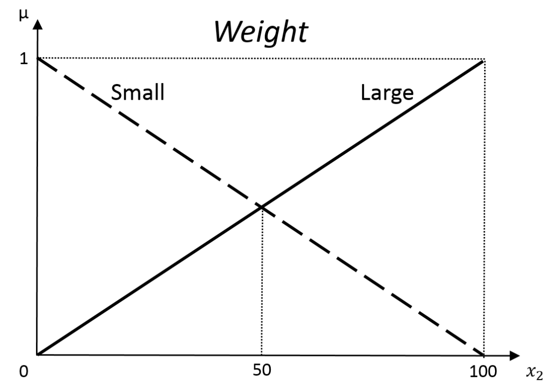
-1 for each missing substitution

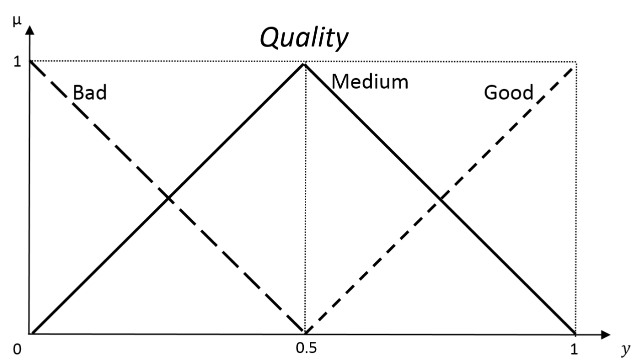
if proof is not complete, 1 point for each node + substitution that are correct in partial proof

4. [20%] Fuzzy Logic

We want to use fuzzy inference system to inspect the quality of apples. In the inspection, we check **Size** and **Weight** to decide **Quality**. Thus, consider three variables, x_1 for size, x_2 for weight, and y for quality. The membership functions for Small or Large size, Small or Large weight, and Bad, Medium or Good quality have triangular shapes as shown below:







Rule 1: If Size is Small and Weight is Small, then Quality is Bad.

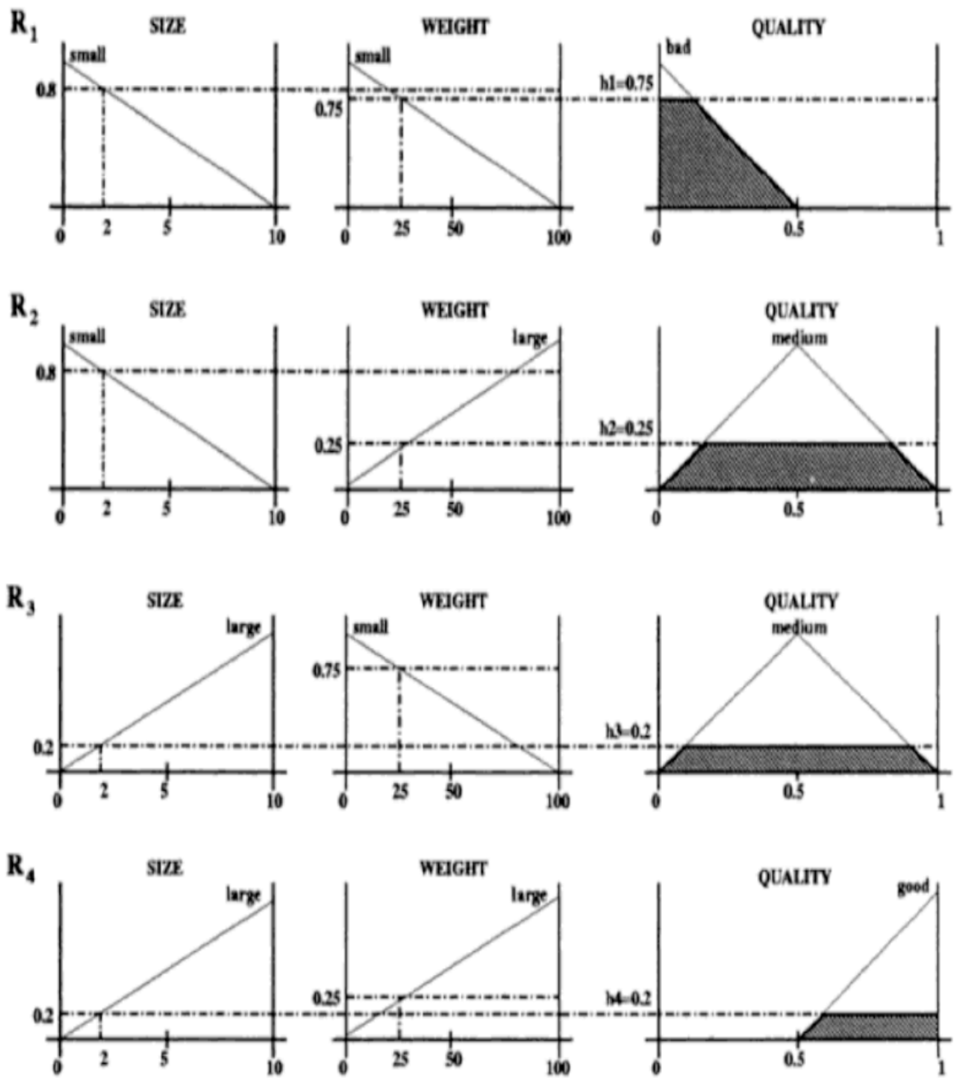
Rule 2: If Size is Small and Weight is Large, then Quality is Medium.

Rule 3: If Size is Large and Weight is Small, then Quality is Medium.

Rule 4: If Size is Large and Weight is Large, then Quality is Good.

Now, given an apple such that **$\{ x_1, x_2 \} = \{ 2, 25 \}$,**

4.1. [16%] Draw graphs to show how the fuzzy rules evaluate for the given apple.
Show the clipped values in your drawings. <u>You do not have to be very precise as long as you indicate the important values on your axes. Be sure to label your axes!</u>



4 points each rule, as follows:

1pt fuzzify size=2 onto correct size fuzzy concept mentioned in rule (small or large). 1 vertical line at size=2 and 1 horizontal line at intersect between the vertical line and the concept

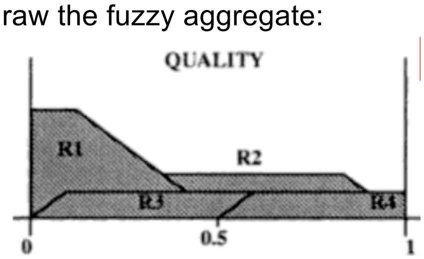
1pt to fuzzify weight=25 in the same way

1pt to take the AND (min) between both horizontal lines

1pt to clip the correct quality concept (bad, medium or good) mentioned in the rule

4.2. [3%] From the clipped values, approximately draw the fuzzy aggregate:

Shape should roughly look like the union of the 4 clipped areas from 4.1



4.3. [1%] Using center of mass, approximately show the defuzzified quality of the given apple, on the same graph as you draw for question 4.2. **Using COG, the quality is about 0.37 (accuracy not important if reasoning is correct)**.

5. [10%] Knowledge Engineering.

Circle the best choice for each question:

(a) [2%] In the debate between Neats vs Scruffies, this is Scruffy:

a. Doug Lenat's Cyc
b. Rodney Brooks' robots
c. MIT's ELIZA chatbot
d. All of the above
e. None of the above

(b) [2%] MYCIN, one of the first expert systems, is

a. An example of *Knowledge Engineering as a transfer process*
b. An example of *Knowledge Engineering as a modeling process*
c. An example of *Knowledge Engineering as a reification process*
d. All of the above
e. None of the above

(c) [2%] Propositional logic does not handle implicature because:

a. An implicature is like entailment in that it can not be cancelled
b. An implicature may not be necessarily logically implied
c. Propositional logic is not monotonic
d. All of the above
e. None of the above

(d) [2%] This is **not** a knowledge sharing research effort:

a. KQML
b. KIF
c. KRSS
d. All of the above
e. None of the above

(e) [2%] A problem that can occur for multiple inheritance is:

a. Objects inherit from only one class yielding conflicting answers
b. Objects inherit from both a superclass and a subclass yielding conflicting answers
c. Objects inherit from multiple classes yielding conflicting answers
d. All of the above
e. None of the above