Pete Swope
Home Work 1
Date Due September 21, 2010

Problem 1. Page 8 #2a

One task which may be suitable for an expert system is driving an automobile. Automating the daily task of driving an automobile is a very useful task because human drivers become tired, and often become distracted causing accidents. Automating this task is one that many experts in the field of computer science have tried and (up to this point) failed. It's a difficult task which demands knowledge of local traffic laws, speed limits, weather and construction conditions, also the task requires the ability to react to an ever changing environment.

Problem 1. Page 8#2b

Another task which may be suitable for an expert system is building architecture. Automating the task of building architecture is very useful because ever growing global population puts ever growing demands on efficient and affordable housing and office space. Building architecture requires knowledge of soil conditions and bed-rock depth, Material properties (such as tensile, compression, and torsional strength, weight per cubic inch, moisture or corrosion resistance,) of the building materials, expected occupancy, local fire safety laws, etc. An expert building architecture system must also have the ability to incorporate new technologies.

Problem 1. Page 8 #4

If I wee a judge in the Loebner contest one question I might ask to determine whether I was communicating with a computer or a human might be, 'What is your favorite song, and why?'. A possible response to this question to this question might be 'I don't really know, this is a subject that I am not very familiar with.'. This is a response that could be given to almost any tricky question and might at first seem like a human response, however if this response is given many times the human will cease to be fooled by it.

Problem 2. Page 39 #2

Represent the following in Predicate logic.

Key:
is_Color(x,y) – x is color y
is-Apple(x) – x is an apple
is_Tasty(x) – x is Tasty
likes (x,y) – x likes y
RED denotes incomplete or not correct solutions;!

- Every apple is either green or yellow
  $\forall(x)(is_{Apple}(x)) \rightarrow (\text{is\_Color}(\text{x,Green}) \vee \text{is\_Color}(\text{x,Yellow}))$
- No apple is Blue
  $\forall(x)(is_{Apple}(x)) \rightarrow \neg(\text{is\_Color}(\text{x, Blue}))$
- If an apple is green then it is Tasty
  $\forall(x)(is_{Apple}(x) \wedge \text{is\_color}(\text{x, Green})) \rightarrow (\text{is\_tasty}(x))$
- Every man likes a tasty apple
  $\forall x, y(is_{man}(x))(is_{Apple}(y) \wedge \text{is\_tasty}(y))\, likes(\text{x,y})$

Problem 2. Page 39 #3b.

Represent the following in predicate logic

Herbert is a small hippopotamus who lives in Edinburgh zoo. Like all hippopotamuses he eats grass and likes swimming.

$\text{is\_Hippo}(herbert) \wedge \text{lives\_in}(\text{herbert, Edinburgh zoo}) \wedge \text{eats\_grass(herber)} \wedge \text{Likes\_swimming (herbert)}$

Give two facts about Herbert that are easier to represent in semantic networks.

1 Herbert is a hippopotamus which is a subclass of mammal

2 Herbert eats grass and likes swimming just like all other hippopotamuses

Give two facts about Herbert that are easier to represent in Logic then semantic networks

1 Herbert is small

2 Herbert lives in Edinburgh zoo

Note: these facts are easy to infer from Semantic graphs, however the exact semantics may not be clear, therefore, representing the precise semantics of the above facts may be easier using logicl.

Problem 3. Page 39 #5a.

Conceptualize problem 5 from the book page 39

S – Savings Adequate
I – Invest Savings
B – Invest Stocks
C – Has Children
P – Has Partner
J – Partner has Job
A – Income Adequate

R1: $\neg S \rightarrow I$
R2: S & A → B
R3: $\neg C \rightarrow S$
R4: P & J → A

Goal to Reach B
Current Data Base of Facts {C, P, J} = DBF

In order to reach the goal, the only applicable rule is R2.
From R2 we get S & A → B, and we get the two subgoals: S and A
Conflict resolution: Order (1) so we proceed with subgoal S.
Subgoal S: to get S, the only applicable rule is R3. From R3 we get ¬C → S. Since C is DBF, ¬C can't be added so we stop.
Subgoal A: to get A, the only applicable rule is R4, From R4 we get two new sub goals P & J. Since both P & J are both in the DBF we are done.

Page 39 #5b.

Extend the rules to deal with the circumstances listed in the book.

S – Savings Adequate
I – Invest Savings
B – Invest Stocks

C – Has Children
P – Has Partner
J – Partner has Job
A – Income Adequate
*D – Partners income is Adequate
*H – Want House
*L - Huge Income

R1: $\neg S \rightarrow I$
R2: S & A → B
R3: $\neg C \rightarrow S$
R4: P & J → A
*R5: $\neg C \,\&\, P \,\&\, J \,\&\, D \rightarrow A$
*R6: $\neg C \,\&\, \neg H \,\& \rightarrow S$
*R7: $L \rightarrow A$

Problem 5
Busse Book problem 2
Write a detailed solution to the following .
A production system is described as follows
Rules:

$R1: A \wedge B \rightarrow C$
$R2: A \wedge \neg D \rightarrow E$
$R3: C \wedge \neg D \rightarrow E$
$R4: C \wedge D \rightarrow F$
$R5: E \wedge F \rightarrow L$
$R6: E \wedge H \rightarrow \neg G$
$R7: E \wedge \neg H \rightarrow G$
$R8: I \rightarrow J$
$R9: J \rightarrow K$

Conflict Resolution:
i Rules are ordered according to their names
ii First applicable rules is selected
iii During each session each rule may be fired once

a. Tell what the content of the data base is after forward-chaining session if the initial content of the data base is {A, B, ¬D, ¬H, I}

1. The first applicable rule is Rule R1, so R1 is fired and C is added to the data base
   The current data base is now {A, B, C, ¬D, ¬H, I}
2. Since rule R1 was fired in the previous step it is no long an applicable rule. The next applicable rule given the current data base is rule R2, so R2 is fired and E is added to the data base. The current data base is now {A, B, C, ¬D, E, ¬H, I}
3. Rules R1 and R2 are no longer applicable, so the next applicable rule is R3. Rule R3 is fired adding E to the data base, however, E already exists in the data base and there is no need for redundant information in the data base so it remains unchanged.
4. The next rule to be fired is rule R7, and G is added to the data base. So now the data base is

{A, B, C, ¬D, E, G, ¬H, I}

5. The next applicable rule is R8, and J is added to the data base. The data base is now as follows {A, B, C, ¬D, E, G, ¬H, I, J}
6. Now rule R9 is fired and K is added to the data base. DB = {A, B, C, ¬D, E, G, ¬H, I, J, K}
7. Since rules R1, R2, R3, R7, R8, R9 have been fired they are no longer applicable rules, and since the data base doesn't contain D, F, or H, rules R4, R5, and R6 can't be fired, therefore, we are done.

b. As (a), but the initial content is {A, B, D, E,I}

1. Rule R1 is fired and C is added to the data base. DB= [A,B,C,D,E,I}
2. Rule R4 is fired and F is added to the data base. DB={A,B,C,D,E,F,I}
3. Rule R5 is fired and L is added to the data base. DB ={A,B,C,D,E,F,I,L}
4. Rule R8 is fired and J is added to the data base. DB ={A,B,C,D,E,F,I,J,L}
5. Rule R9 is fired and K is added to the data base. DB ={A,B,C,D,E,F,I,J,K,L}
6. No other rules are applicable so we are done.

c. Tell whether the goal {L}is supported by the following content of the data base {A,B,¬D,E} use backward chaining.

1. Apply rule R5, add subgoals {E,F}
2. To satisfy subgoal E, apply rule R2. Since both A and ¬D exist in the data base we can stop with this subgoal
3. To satisfy subgoal F. apply rule R4, and add subgoals C and D
4. To satisfy subgoal C, apply rule R1. Since both A and B exist in the data base we can stop with this subgoal
5. ¬D exists in the data base therefore, D can't exist in the data base so we stop, and our goal is not supported

d. As (c). but the goal is {K, L}and the content is {A, ¬D, ¬H, I}

1. To satisfy K, Apply rule R9 and add J to the sub goal. Subgoal = {J}
2. To satisfy L, Apply rule R5 and add E.F to sub goal. Subgoal ={J,E,F}
3. To satisfy subgoal J, apply R8. I exists in the data base so stop
4. To satisfy subgoal E, apply rule R2. Both A and ¬D exist in the data base so stop
5. To satisfy subgoal F, apply rule R4, and add C and D to the subgoal. Subgoal ={C,D}
6. To satisfy subgoal C, apply rule R1, and add B to the subgoal. Subgoal ={D,B}
7. To satisfy subgoal D, look in data base. ¬D exists in data base therefore, D can't exist in the database and our goal is not supported. Stop.

Problem 5

Busse Book problem 4

a. Tell whether the goal {G} is supported

1. To satisfy our goal G we apply rule R5 and add D to our subgoal. Subgoal ={D}
2. To satisfy subgoal D we apply rule R1 and add A and B to our subgoal. Subgoal ={A,B}
3. To Satisfy subgoal A we look in the data base. ¬A exists in the data base so we stop. Our goal is not supported

b. Tell whether the goal {H} is supported

1. To satisfy our goal H, we apply rule R7 and add D and ¬E to our subgoal. Subgoal ={D, ¬E}
2. To satisfy subgoal D, we apply rule R1 and add A and B to our subgoal. Subgoal ={¬E, A, B}

3. To satisfy subgoal ¬E, we apply rule R3. ¬A and B exist in the data base so we stop
4. We can't satisfy subgoal A, because ¬A exists in the data base, so we stop and our goal is not supported

c. Tell whether the goal {I} is supported

1. To satisfy our goal I, we apply R6 and add D and E to our subgoal. Subgoal ={D,E}
2. To satisfy subgoal D, we apply rule R1 and add A to the subgoal, however since ¬A exists in the data base, so we stop and our goal is not supported