Menoufia University

Knowledge Based System

Faculty of Computers & information

Department: CS & IT

1st semester Final Exam

3 pages

Subject:

Year: 4th year

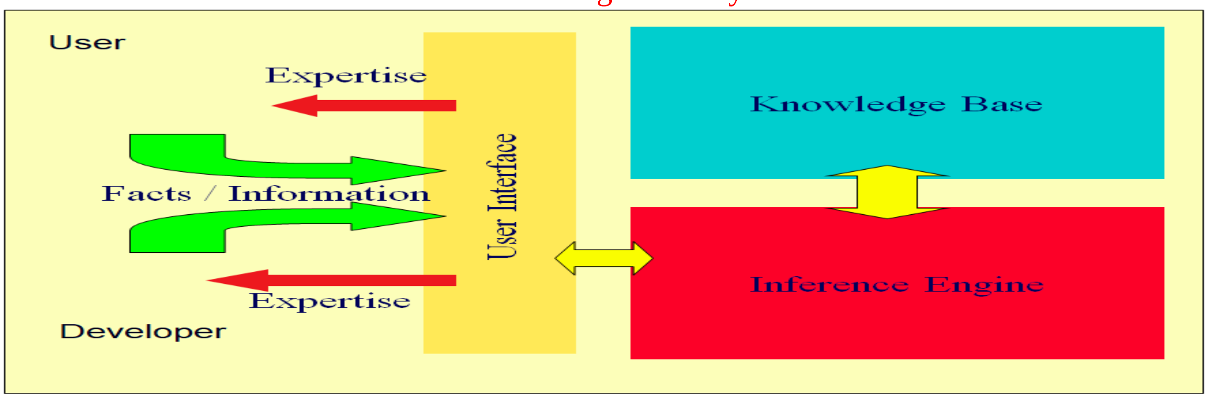
Time Allowed: Three Hours

Full Mark: 70

Please Answer All Questions;

Question 1 ____ 12 Marks ________

1. Illustrate the basic architecture of Knowledge based system.



"""اشرحها"

2. List the main characteristics of an expert system?
   - Expertise:
     - Exhibit expert performance
     - Have high level of skill
   - Symbolic reasoning
     - Represent knowledge symbolically
   - Depth
     - Handle difficult problem domains
     - Use complex rules
   - Self-knowledge
     - Examine its own reasoning
     - Explain its operation
3. List expert systems advantages and problems?
   - economical
     - lower cost per user
   - availability
     - accessible anytime, almost anywhere
   - response time
     - often faster than human experts
   - reliability
     - can be greater than that of human experts
     - no distraction, fatigue, emotional involvement, …
   - explanation
     - reasoning steps that lead to a particular conclusion

   - limited knowledge
     - "shallow" knowledge

        - no “deep” understanding of the concepts and their relationships
    - no “common-sense” knowledge
    - no knowledge from possibly relevant related domains
    - “closed world”
        - the ES knows only what it has been explicitly “told”
        - it doesn’t know what it doesn’t know
- mechanical reasoning
    - may not have or select the most appropriate method for a particular problem
    - some “easy” problems are computationally very expensive
- lack of trust
    - users may not want to leave critical decisions to machines

4. List the different types of Knowledge?
    - a priori knowledge
        - considered to be universally true
    - a posteriori knowledge
        - knowledge verifiable through the senses
        - may not always be reliable
    - procedural knowledge
        - knowing how to do something
    - declarative knowledge
        - knowing that something is true or false
    - tacit knowledge
        - knowledge not easily expressed by language

5. **What are the difficulties of developing the KBS?**
        - conventional algorithms are known and efficient
        - the main challenge is computation, not knowledge
        - knowledge cannot be captured easily
        - users may be reluctant to apply an expert system to a critical task

6. Use truth tables to determine whether each of the following is a valid sentence in propositional logic. $(P \rightarrow Q) \leftrightarrow (\neg P \vee Q)$

P	Q	P -> Q	~P ∨ Q	WFF
F	F	T	T	T
F	T	T	T	T
T	F	F	F	T
T	T	T	T	T

Since the formula is always satisfied, we say that the formula is **valid.**

<u>Question 2</u> **<u>16 Marks</u>**

1. Suppose we have a working memory consisting of the following assertions,
    a. Use the Forward-chaining technique to diagnose the following symptoms.
    b. Use the Backward-chining technique to ensure your previous diagnosis.

1: runny nose
2: temperature = 104°F
3: headache
4: cough
<u>The rules in our rule base are:</u>
R1: if (nasal congestion and viruses), then diagnose (influenza) exit
R2: if (runny nose), then assert (nasal congestion)
R3: if (body aches), then assert (itchiness)
R4: if (temp > 100), then assert (fever)
R5: if (headache), then assert (itchiness)
R6: if (fever and itchiness and cough), then assert (viruses)

a. forward

- First R2 fires, so (nasal congestion) is added to working memory.
- R4 fires, so (fever) is added to working memory.
- R5 fires, so (itchiness) is added to working memory.
- R6 fires, so (virusis) is added to working memory.
- R1 fires, diagnosing the ailment as (influenza) and exits.

b. Backward

- Initially, R1 fires: Since the goal, diagnosis (influenza), goes with the conclusion of the rule R1, new goals are created: (nasal congestion) and (virosis), and back chaining is recursively used with these recent goals.
- R2 fires, matching the goal nasal congestion. A new goal is created: (runny nose). Back chaining is recursively used. Since (runny nose) is in working memory, it returns true.
- R6 fires, matching the goal virosis. Back-chaining recursion takes place with new goals: (fever), (itchiness), and (cough)
- R4 fires, adding the goal (temperature > 100). Since (temperature = 104) is in working memory, it returns true.
- R3 fires, adding the goal (body aches).
- On recursion, there is no information in working memory or rules that match this goal. Therefore, it returns false and the next matching rule is chosen.
- Here, the rule is R5, which fires, so the goal (headache) is added. Since (headache) is in working memory, it returns true.
- The goal (cough) is in working memory, so that returns true.
- Finally, the entire recursive procedures have returned true. This proves the result.

2- Give one predicate calculus representation for each of the following English sentences.

a) Not all people have a cell phone
   - ∃p person(p) ^ ~hasCellPhone(p)

b) Students who took CS101 but did not take CS102 must take Math103.
   - ∀s student(s) ^took(s,CS101)^ ~took(s,CS102) → requiredToTake(s,Math103)

c) All Germans speak the same languages
   - ∀ *x,y,l German(x)^German(y)^Speaks(x,l)=> Speaks(y,l)*

Question 3 16 Marks

1- Give one (1) predicate calculus representation for each of the following English sentences. If you feel a sentence is ambiguous, provide a more detailed sentence that better captures the version represented by your FOPC. Choose reasonable constants, predicates, and functions

a. There are no green Martians.
   - ~(∃x green(x) ^Martian(x)) -- also correct to say ∀x~green(x) ∨~Martian(x)

b. Everything painted by Picasso is valuable.
   - ∀x painting(x) ^ paintedBy(x, Picasso) → valuable(x)

c. Not all people have a cell phone.
   - ∃p person(p) ^ ~hasCellPhone(p)

d. Everyone who owns a violin knows someone that likes music written by Mozart.
   - ∀x person(x) ^owns(x, violin) →(∃y person(y) ^knows(x,y)^ likesMozartMusic(y))

e. Students who took CS101 but did not take CS102 must take Math103.
   - ∀s student(s) ^took(s,CS101)^ ~took(s,CS102) → requiredToTake(s,Math103)

f. Every student at UW-Madison knows someone who likes dogs.
   - ∀x studentAt(x,UW_Madison) → (∃y person(y) ^knows(x, y) ^likesDogs(y))

2- Translate the following into idiomatic English.

$$\exists\, z\,.P(z\,,\,x\,)\wedge S(z\,,\,y)\wedge W(y)$$

where P(z , x) means z is a parent of x , S(z , y) means z and y are siblings, W(y) means y is a woman, and x , y, and z range over people

- Some parents have all their siblings as women.

3- Which of the following are semantically and syntactically correct translations of "Everyone's zipcode within a state has the same first digit"?

a) ∀x, s, z1 (State(s) ^ LivesIn(x, s) ^ Zip(x)=z1)→ ∀ y, z2 LivesIn(y, s) ^ Zip(y)=z2) Digit(1, z1)=Digit(1,z2)].

b) x, s [State(s) ^ LivesIn(x, s) ^ ∃ z1 Zip(x)=z1])→ [∀ y, z2 LivesIn(y, s) ^ Zip(y)=z2 ^ Digit(1, z1)=Digit(1, z2)].

c) ∀x, y, s State(s) ^ LivesIn(x, s) ^ LivesIn(y, s) →Digit(1,Zip(x)=Zip(y)).

d) ∀x, y, s State(s) ^ LivesIn(x, s) ^ LivesIn(y, s) → Digit(1,Zip(x))=Digit(1,Zip(y)).

- Ans is A & D

Question 4 16 Marks

1- Represent each of the following sentences in propositional logic. Use reasonable propositions. For example, the proposition *spot_is_sleeping_dog* is not a good answer for the first question. Since we are using propositional logic and not predicate logic in this question, it is ok to use *dogSpot* and *sleepingSpot* here, which in predicate logic would be *dog(spot)* and *sleeping(spot)*.

a. Spot is a dog who is sleeping.

- dogSpot ^ sleepingSpot

b. Felix is a cat who is not sleeping.

- CatFelix ^ ~sleepingFelix

c. If the TV is on, neither Spot nor Felix are sleeping.

- TV on → (~ sleepingSpot ^ ~sleepingFelix)

d. One of Spot and Felix is awake.

- We can interpret this statement in two different ways. This issue arises because English is an inherently ambiguous language like all natural human languages. If it is true that exactly one of them is sleeping, then we can represent that as:
  - **(~sleepingSpot ^ sleepingFelix) ∨(sleepingSpot ^ ~sleepingFelix)**

  If instead it is true that at least one of them is awake, we can represent that as:
  - **sleepingSpot∨sleepingFelix**

e. Whenever Spot is sleeping Felix isn't and whenever Felix is sleeping Spot isn't.

- (sleepingSpot → ~sleepingFelix) ^ (sleepingFelix→ ~sleepingSpot)

2- Russia was a superior power, and either France was not strong or Napoleon made an error. Napoleon did not make an error, but if the army did not fail, then France was strong. Hence the army failed and Russia was a superior power. Prove this argument is valid !

R: Russia was a superior power.
F: France was strong.
N: Napoleon made an error.
A: The army failed.

$$R \wedge [\neg F \vee N]$$

(a)

$$\neg N \wedge [\neg A \rightarrow F]$$

$$A \wedge R$$

(b) First,

$$R \wedge [\neg F \vee N] \rightarrow R$$

$$R \wedge [\neg F \vee N]$$

R --- (1)

Next,

$$R \wedge [\neg F \vee N] \rightarrow [\neg F \vee N]$$

$$R \wedge [\neg F \vee N]$$

$[\neg F \vee N]$ --- (2)

Also

$$\neg N \wedge [\neg A \rightarrow F] \rightarrow \neg N$$

$$\neg N \wedge [\neg A \rightarrow F]$$

$\neg N$ -- (3)

Hence from (2) and (3),

$$[\neg F \vee N]$$

$$\neg N$$

$\neg F$ -- (4)

Now,

$$\neg N \wedge [\neg A \rightarrow F] \rightarrow [\neg A \rightarrow F]$$

$$\neg N \wedge [\neg A \rightarrow F]$$

$[\neg A \rightarrow F]$ -- (5)

From (4) and (5)

$$[\neg A \rightarrow F]$$

$$\neg F$$

A -- (6)

Hence from (1) and (6) $A \wedge R$ is deduced. Hence the argument is correct. --->

<u>شرح للسؤال دا ل زميل معانا</u>

Question 5 16 Marks

1. Write down logical representations for the following sentences, suitable for use with Generalized Modus Ponens:

   a. Horses, cows, and pigs are mammals.
      - Horse (x) => Mummal (x)
      - Cow (x) => Mummal (x)
      - Pig (x) => Mummal (x)

   b. An offspring of a horse is a horse.
      - Offspring(x,y) ^ Horse (y)=> Horse (x)

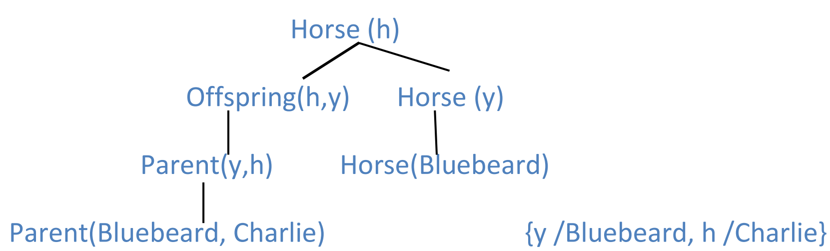
   c. Bluebeard is a horse.
      - Horse(Bluebeard)

   a. Bluebeard is Charlie's parent.
      - Parent(Bluebeard, Charlie)

   a. Offspring and parent are inverse relations.
      - Offspring(x,y) => Parent(y,x)
      - Parent(x,y) => Offspring (y.x)
      - (Note we couldn't do Offspring (x,y) <=> Parent (y,x) because that is not in the form expected by generalized Modus Ponens)

   b. Every mammal has a parent.
      - Mummal(x) => Parent (G(x),x) (here G is a Skolem function)

2. Draw the proof tree generated by an exhaustive backward-chaining algorithm for the query **h Horse *(h)*** where clauses are matched in the order given.



With my best wishes