**CONFIDENTIAL**



## UNIVERSITI TEKNOLOGI MALAYSIA

# **MID TERM TEST**

# **SEMESTER II 2017/2018**

|  |  |
| --- | --- |
| **SUBJECT CODE** | **: SCSJ3553 / SCJ3553** |
| **SUBJECT NAME** | : ARTIFICIAL INTELLIGENCE |
| **YEAR/COURSE** | **:** |
| **TIME** | **: 2 ½ HOUR** |
| **DATE** | **:** |
| **VENUE** | **:** |

**INSTRUCTIONS TO THE STUDENTS:**

This test book consists of 2 sections:

Part A: Short Explanation [25 Marks]

Part B: Theory and Applications [50 marks]

**ANSWER ALL QUESTIONS IN THIS QUESTION PAPER.**

|  |  |
| --- | --- |
| **Name** |  |
| **I/C No.** |  |
| **Year/Course** |  |
| **Section** |  |
| **Lecturer Name** |  |

**SECTION A TOTAL 25 MARKS**

***Short Explanation Questions: Answer each question in the space provided.***

1. Describe TWO characteristics of think humanly and act rationally. Provide ONE example for each category. [ 5 marks]

|  |  |  |
| --- | --- | --- |
|  | **Think Humanly** | **Act Rationally** |
| **Characteristics** | Adapting cognitive behavior of human | Reacting on input based on reasoning and analysis |
| Need to study about neuron activities | Generally considered as highest intelligence for machine |
| **Example** | Recognizing human face | Suggesting / predicting similar products for future shopping |

1. a) Artificial Intelligence is the study of knowledge representation and search through which activity can be enacted on a mechanical device. Name two knowledge representation schemes and briefly explain each. [ 4 marks]

* Propositional Logic/Calculus

It is a system of [symbolic logic](http://dictionary.reference.com/browse/symbolic%20logic) using symbols to stand for whole [propositions](http://dictionary.reference.com/browse/propositions) and [logical connectives](http://dictionary.reference.com/browse/logical%20connectives).

* Predicate Logic/Calculus

It is a formal language for describing knowledge and objects, their properties and relations.

1. Define briefly what these terms mean in logic reasoning  [ 2 marks]
2. Quantifiers and its significance

A key element of FOL are predicates, which are used to describe objects, properties, and relationships between objects. The use of Quantifiers allows FOL to handle infinite domains, while propositional logic can only handle finite domains.

1. Inference

The process of deriving new sentences from old one is called **inference**.

1. a) Resolution is a technique for proving theorems in predicate calculus. The unification is one of the important practical application in resolution theorem proving. Briefly explain how the unification is applied in resolution theorem proving. [2 marks]

A solution of a unification problem is denoted as a substitution, that is, a mapping assigning a symbolic value to each variable of the problem's expressions. To resolve the problem, just substitute the variable with suitable constant according the facts.

1. What are the main steps during the refutation resolution process? [2 marks]
2. Assume that our fact base is consistent
3. Add the *negation* of the thing we want to prove to the fact base
4. Show that the fact base is now inconsistent
5. Conclude the thing we want to prove
6. a) Identify whether the pair of PROLOG clauses given below can be unified or fails to do so. If unification is possible, give the unifier(s). Otherwise, say why it fails. [3 marks]

|  |  |  |
| --- | --- | --- |
| Clause 1 | Clause 2 | Unifier |
| vacation\_at (person, place)  friend(x,y)  fav\_subj(2, ai, prolog) | vacation\_at (Ahmad, Osaka)  friend(Ali, neighbour(Ali))  fav\_subj(x, y, c++) | **{person/Ahmad, place/Osaka}**  **{ x/ Ali, y/neighbour(Ali) }**  **FAILS, x/2, y/ ai, but prolog cannot unify with c++** |

b) How is a recursive rule formulated and written in PROLOG? [1 mark]

A recursive rule will consist of the recursive step whereby the rule is defined by calling itself. Eventually it will match a fact in the knowledge base to terminate the recursive procedure

c) State whether the following rules use recursion ? State your answer with Yes or No. [2 marks]

go\_home(X) :- get\_next\_house(X,Y), home(Y). No

on\_route(P):-move(P,M,N), on\_route(N). Yes

search(X):- path(X,Y), search(Y). Yes

no\_friends(X):- totally\_lonely(X). No

1. Graphsare generally used as an abstract way of representing search problems so that general-purpose algorithms may be applied, without having to develop a new method for each new problem. Given the following map, find a path from a *start* *node*, Entrance (*E*) to a *target* *node*, Deer park (*Dp*), represent the possible route as search *graph*. Follow the arrow sign as possible route. [ 4 marks]

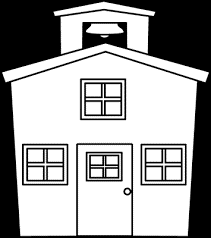
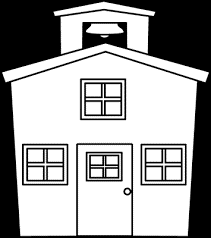
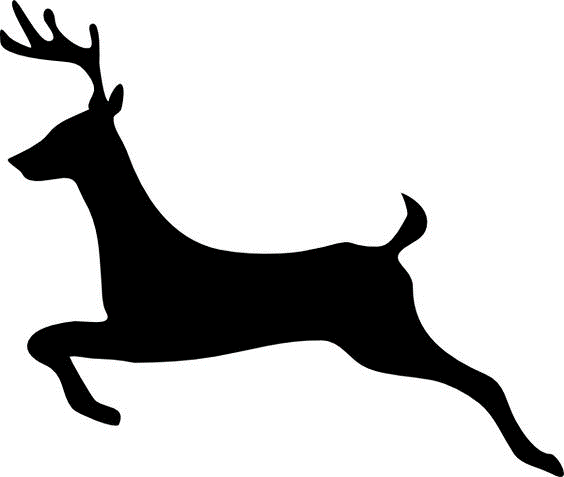
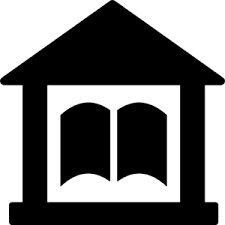
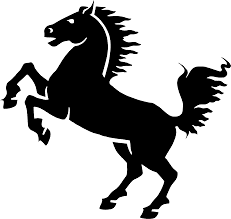
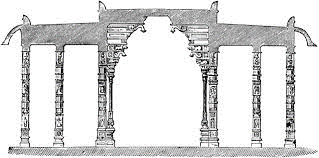
**Deer park (Dp)**

**Canoe Centre (CC)**

**Equine Centre (EC)**

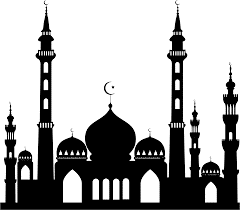
**Library (L)**

**Masjid (M)**

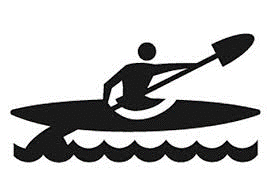


**Entrance (E)**

**School (S)**



Lake



**Entrance (E)**

**School (S)**

**Equine Centre (EC)**

**Masjid (M)**

**Canoe Centre, (CC)**

**Library (L)**

**Deer Park (Dp)**

**SECTION B TOTAL 50 MARKS**

***Structured Questions: Answer questions in the space provided.***

1. Translate the following English sentences into propositional logic formulae using the symbols given. [5 marks]

|  |  |
| --- | --- |
| A: Ali will make dinner | E: Ela exercises regularly |
| B: Bedah will make dinner | F: Faiz is getting fat |
| C: Chin will make lunch | G: Faiz jogs regularly |
| D: Daud will make lunch |  |

1. Ela will exercise regularly on the condition that Faiz jogs regularly.

G -> E

1. Either Ali will make dinner while Chin will make lunch or Bedah will make dinner while Daud will make lunch.

(A ˄ C) ˅ (B ˄ D)

1. Assuming that Faiz does not jog regularly, he is getting fat.

!G -> F

1. Bedah will make dinner and either Chin or Daud will make lunch.

B ˄ (C ˅ D)

1. Ali will make dinner if and only if Bedah will.

A ⬄ B

1. Convert the English sentences in parts (i) – (v) into predicate logic sentences using the predicates indicated. [10 marks]

player, tall, businessman, trader, investor, mouse, zigzag, smart, cat, chase, catch, frustrated, offspring, parent

1. Not all basketball players are tall.

∀x ¬ (player(x) → tall(x)

1. Every businessman is either a trader or an investor.

∀x (businessman(x) trader(x) v investor(x))

1. Every mouse who runs ``zig-zag” is smart.

∀y(mouse(y) ∧ zigzag(y) smart(y))

1. Any cat who chases some mouse but does not catch it is frustrated.

∀x∃ y(cat(x) ∧ mouse(y) ∧ chase(x,y) ∧ ¬catch(x,y) frustrated(x))

1. Offspring and Parent are inverse relations.

ParentOf(x,y) ⇔ Offspring(y,x) 2m

OR

ParentOf(x,y)  Offspring(y,x)

2 m

Offspring(y,x) ParentOf(x,y)

1. Given the following simplified predicate logic sentences. [12 marks]
   * + Students(Amar)
     + Student(Sofi)
     + Student(Hafiz)
     + Student() Poor () V Good (
     + Good() Likes(chatting)
     + Poor() Likes(playing)
     + Likes(Hafiz,) Likes(Amar,)
     + Likes(Amar, chatting)
     + Likes(Amar, playing)

Using the above sentences construct a proof by refutation resolution of the sentence

Goal : Good()  Likes (, playing)

[This page is intentionally left blank for Question 3. ]

**Answer:**

1. Student ( Amar)
2. Students ( Sofi)
3. Student (Hafiz)
4. Student () Poor () V Good (

Student () V Poor () V Good ( 1m

1. Good() Likes(chatting)

Good() V Likes(chatting) 1m

1. Poor() Likes(playing)

Poor() V Likes(playing) 1m

1. Likes ( Hafiz,) Likes (Amar,)

7a. Likes ( Hafiz,) V Likes (Amar,) 1m

7b. Likes ( Hafiz,) V Likes (Amar,) 1m

1. Likes (Hafiz, chatting)
2. Likes (Hafiz, playing)

Goal : Good(x) Likes (, playing)

Negate the goal: Good(x) ∧ Likes (, playing)

(Good(x) ∧ Likes (, playing))

10. Good(x ) V Likes (, playing) 1m

Resolve 9 & 7a. : {x/playing }

11. Likes ( Amar,playing) 1m

Resolve 11 & 10 : {x/Amar }

12. Good(Amar) 1m

Resolve 11 & 6: {x/ Amar }

13. Poor(Amar) 1m

Resolve 1 & 4 : {x/Amar }

14. Poor (Amar) V Good (Amar) 1m

Resolve 13 & 14 15 Good (Amar) 1m

Resolve 12 & 15 16. Nil. 1m

1. Consider the following scenario about students and their relations or properties.

Burhan, Cindy and Dinaz are students of Faculty of Computing (FC). Alia, Lim, Jo and Zita are students of Faculty of Civil Engineering (FA). Cindy and Dinaz are final year students. Burhan and Jo are third year students. Lim is a second year student. On the other hand, Alia and Zita are first year students. FC and FA are faculties in UTM.

Students of a faculty are students of UTM. Students of FC are friends with students of FA. Third year students goes for industrial training (LI). First year students must take Introductory Programming (PT). Second year students should take Foreign Language (FL). A final year student studies in the fourth year at UTM.

Hint: use these predicates / clauses : studentOf, year, facOf, take, goes, friend.

Based on the above relations:

1. Write a PROLOG program to exhibit its knowledge base (KB) which consists of both facts and rules. [12.5 marks]
2. Code queries in PROLOG to answer the following questions. [2.5 marks]
3. Who are students in the Faculty of Computing?
4. Who are friends of students in the Faculty of Civil Engineering?
5. List the students studying in the fourth year?
6. List the students studying Introductory Programming?
7. Who are students that are going for industrial training?

[This page is intentionally left blank for Question 4 (a) and 4 (b). ]

4 a)

% facts 8 marks worth 0.5 each

studentOf(burhan, fc).

studentOf(cindy, fc).

studentOf(dinaz, fc).

studentOf(alia, fa).

studentOf(lim, fa).

studentOf(Jo, fa).

studentOf(zita, fa).

year(cindy, final).

year(dinaz, final).

year(burhan, 3).

year(jo, 3).

year(lim, 2).

year(alia, 1).

year(zita, 1).

facOf(fc, utm).

facOf(fa, utm).

%rules 4.5 marks

take(X, pt) :- year(X, 1).

take (X, fl):- year(X,2).

year(X, 4) :- year(X, final).

goes(X, li) :- year(X, 3).

friend(X, Y) :- studentOf(X, fc), studentOf(Y, fa).

studentOf(X, utm) :- studentOf(X, Y), facOf(Y, utm).

b)

1. Who are students in the Faculty of Computing?

? - studentOf(X, fc).

1. Who are friends of students in the Faculty of Civil Engineering?

? - friend(X, fa).

1. List the students studying in the fourth year?

? - year(X, 4).

1. List the students studying Introductory Programming?

? - take(X, pt).

1. Who are students that are going for industrial training?

? - goes(X, li).

1. A *farmer (F)* wishes to carry a *wolf (W)*, a *duck (D)* and *corn (C)* across a river, from the *Riverbank 1* to the *Riverbank 2*. The *farmer* is the proud owner of a small rowing boat called *Fajar Samudera* which he feels is easily up to the job. Unfortunately the boat is only large enough to carry at most the *farmer* and one other item. Worse again, if left unattended the *wolf* will eat the *duck* and the *duck* will eat the *corn*.”

Riverbank 1

F,W,D,C

**Figure 1**

River

F,W,D,C

Riverbank 2

Using the river problem as illustrated in Figure 1, show the transition made between states by a sequence of legal moves in order to solve the farmer’s problem (to carry a wolf, a duck and corn across a river)

Below are the the list of rules that you may apply to show the possible moves in solving this problem:

R1 : Farmer takes duck to riverbank 2

R2: Farmer takes wolf to riverbank 2

R3: Farmer takes corn to riverbank 2

R4: Farmer returns with duck to riverbank 1

R5: Farmer returns with wolf to riverbank 1

R6: Farmer returns with corn to riverbank 1

R7: Farmer returns alone to riverbank 1

In this configuration , the status of Farmer (F), Wolf (W), Duck(D) or Corn (C) is either located on the riverbank 1, **R1** or on the riverbank 2, **R2**.

For example, the start state can be visualized as where Farmer (F), Wolf (W), Duck(D) and Corn (C) are located on the riverbank1 while for , Wolf (W), Duck(D) are located on riverbank 1 and Corn has been moved to riverbank 2.

Draw the full search tree to get from the start state until a goal configuration  is found. Avoid cycles (dead ends) by not generating children of states already explored.

[ 8 marks]



Legend



**Appendix A**

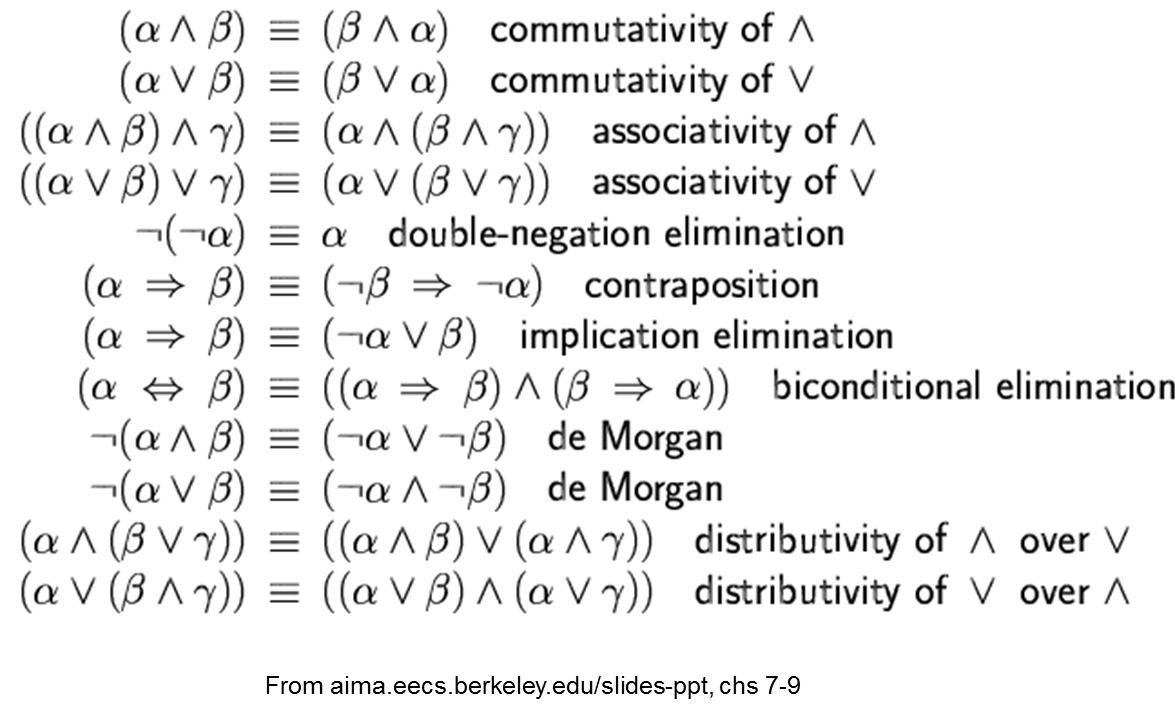
¬(¬p) ≡ p

¬(a ∧ b) ≡ (¬a ∨ ¬b)

¬(a ∨ b) ≡ (¬a ∧ ¬b)

¬∀x, p(x) ≡ ∃x, ¬p(x)

¬∃x, p(x) ≡ ∀x, ¬p(x)

****