Roll No. | | | | | | | | |

35

B.E / B.Tech (Full Time) DEGREE END SEMESTER EXAMINATIONS, APRIL / MAY 2013

Computer Science and Engineering

V Semester

CS 9304 Artificial Intelligence

(Regulation 2008)

Time : 3 Hours | Answer ALL Questions | Max. Marks 100

PART-A (10 x 2 = 20 Marks)

1. What is a Rational Agent?
2. Develop a PEAS description of the task environment for a Movie Booking agent.
3. Explain admissibility of a heuristic function
4. Discuss Games with an element of chance and explain how to represent such games.
5. Tom flipped a coin of unknown bias 100 time under the same conditions. He told you that trials 2 to 90 are all heads, and trials 91 to 99 are all tails. But he didn't tell you the outcome of trail 1 or trial 100. Given what he told you, is trial 1 or trial 100 more likely to be heads?
6. Define the predicates *Before*, *During* and *Overlap* using the predicate *Meet* and the functions *Start* and *End*, but not the function *Time* or the predicate <.
7. Draw a Decision Tree for the problem of deciding whether to move forward at a road intersection, given that the light has just turned green.
8. Explain the concept of Relevance Based Learning?
9. Natural Language is ambiguous. Justify this Statement.
10. Discuss the concept of Biometric Recognition with an example.

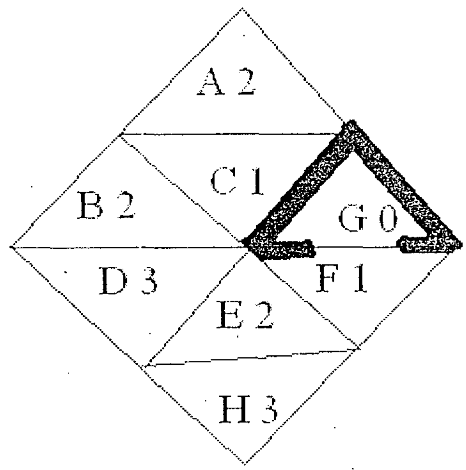
PART-B (5 X 16 = 80 Marks)

11. (i) Differentiate between Search Trees and Game Trees. (4)
    (ii) Explain the difference between uninformed and informed search. Compare the five uninformed search strategies explaining how they are distinguished by the order in which the nodes are expanded. (8)
    (iii) What are contingency problems. Discuss (4)
12. (a) Consider the following path-finding problem (Figure 1). One can move from one small triangle to another if they share a vertex (e.g., *A* can go to *B* and *C*). However, the goal *G* can only be accessed from *F*. The number after the letter is the heuristic function value for that state. The actual cost of each move is as follows:

    - A move down one level (e.g. *A* → *C* or *B* → *E*) costs 1
    - A move sideways on the same level (e.g. *C* → *B* or *E* → *F*) costs 2
    - A move up one level (e.g. *B* → *A* or *C* → *A*) costs 3

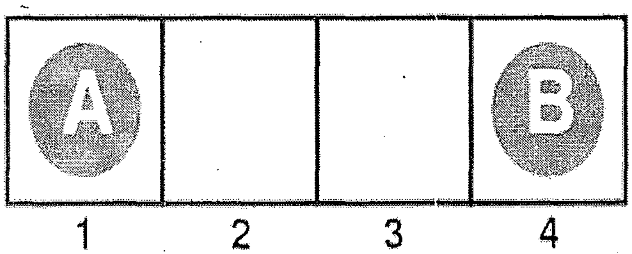
(i) Perform Depth-First Search, starting from *A*, using path-checking to avoid repeated states if they occur on the path back to the root in the search tree. Expand successors in alphabetical order. Show your search tree, and *circle* states that are expanded. What is the *cost* of your solution path? (8)

(ii) Perform A* Search, starting from *A*. Break ties alphabetically. Show the *expanded states* and the *priority queue* contents at each step. What is the *cost* of your solution path? (8)



(Figure 1)

12 (b) Consider the two-player game described below (Figure 2).



This is the starting position of a simple game. Player *A* moves first. The two players take turns moving, and each player must move his token to an open adjacent space *in either direction*. If the opponent occupies an adjacent space, then a player may jump over the opponent to the next open space if any (for example, if *A* is on 3 and *B* is on 2, then *A* may move back to 1). The game ends when one player reaches the opposite end of the board. If player *A* reaches space 4 first, then the value of the game to *A* is +1; if player *B* reaches space 1 first, then the value of the game to *A* is -1.

(Figure 2)

(ii) Draw the complete game tree, using the following conventions:

- Write each state as (s_A, s_B) where s_A and s_B denote the token locations.
- Put each terminal state in a square boxes and write its game value in a circle next to it.
- Put *loop states* (states that already appear on the path to the root) in double square boxes. Since it is not clear how to assign values to loop states, annotate each with a "?" in a circle. (8)

(ii) Now mark each node with its backed-up minimax value (also in a circle). Explain how you handle the "?" values and why. (4)

(iii) What is Alpha –Beta Pruning? Discuss. (4)

13 (a) Tony, Shi-Kuo and Ellen belong to the Hoofers Club.
Every member of the Hoofers Club is either a skier or a mountain climber or both.
No mountain climber likes rain, and all skiers like snow.
Ellen dislikes whatever Tony likes and likes whatever Tony dislikes.
Tony likes rain and snow.
Prove via the resolution method and unification (where needed) that
"Ellen is a mountain climber but not a skier". (16)

(OR)

13 (b) (i) Decide whether each of the following sentences is valid, unsatisfiable or neither.
Verify your decisions using Truth Tables.

- $(Smoke \Rightarrow Fire) \Rightarrow (\neg Smoke \Rightarrow \neg Fire)$
- $Smoke \vee Fire \vee \neg Fire$
- $((Smoke \wedge Heat) \Rightarrow Fire) \Leftrightarrow ((Smoke \Rightarrow Fire) \vee (Heat \Rightarrow Fire))$
- $(Smoke \Rightarrow Fire) \Rightarrow ((Smoke \wedge Heat) \Rightarrow Fire)$ (8)

(iii) The following Prolog Code defines a predicate P:

```
P(X, [X|Y]).
P(X, [Y|Z]) :- P(X,Z)
```

Show proof trees and solutions for the queries P(A, [1,2,3]) and P(2, [1,A,3]). What standard list operation does P represent? (8)

14. (a) (i) Consider the family tree given in Figure 3.

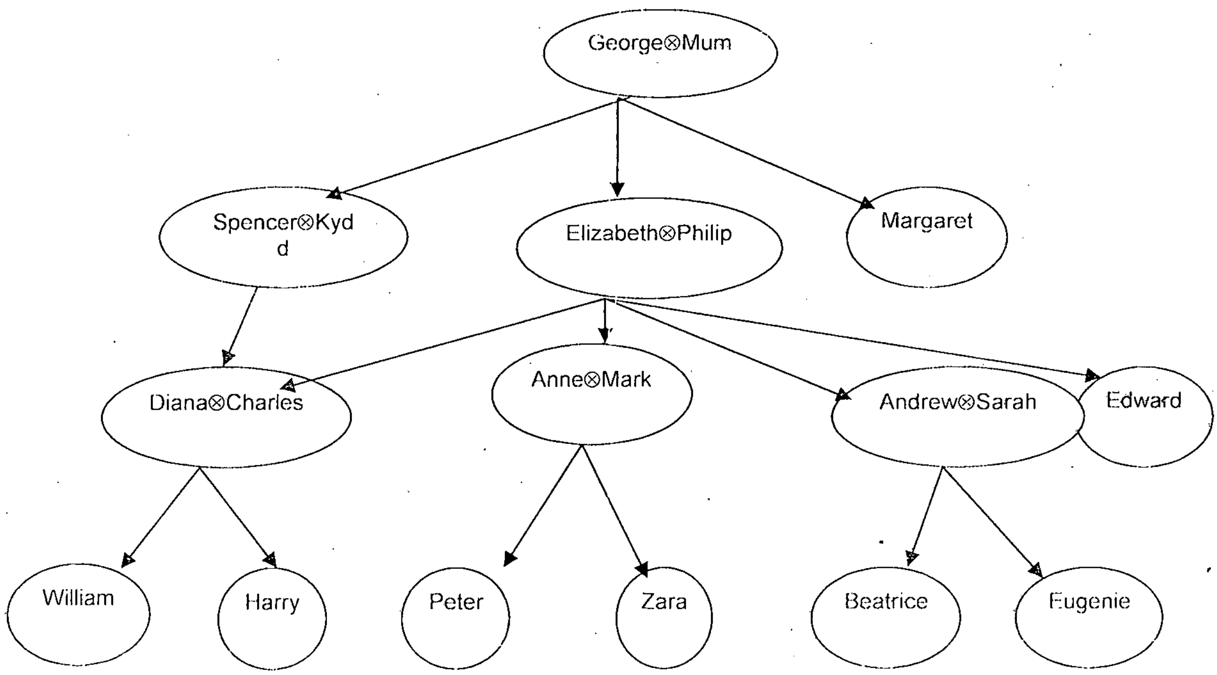


Figure 3

Assume ***Father*** and ***Mother*** Predicates are given. Explain clearly how Inductive Logic Programming can be used to learn the ***Ancestor*** relationship. (8)

(ii) Explain the EM algorithm in detail. (8)

(OR)

14 (b) (i) Describe Bayesian learning with an example. Mention the approximations in *Max A Posteriori* (MAP) and *Max Likelihood* (ML) hypothesis? (8)

(ii) Explain the concept of Reinforcement Learning. Compare and contrast passive and active reinforcement learning. (8)

15 (a) (i) Explain the Seven processes involved in communication using a typical example (8)

(ii) Discuss the Chart Parsing algorithm in detail with an example. Write down the necessary grammar rules required.

(OR)

15 (b) (i) Write a detailed note on Object Recognition (8)

(ii) How is three dimensional information extracted? Explain (8)