CS6659 – ARTIFICIAL INTELLIGENCE

UNIT I- INTRODUCTION TO Al AND PRODUCTION SYSTEMS 9

Introduction to AI-Problem formulation, Problem Definition -Production systems, Control strategies, Search strategies. Problem characteristics, Production system characteristics -Specialized production system- Problem solving methods -Problem graphs, Matching, Indexing and Heuristic functions – Hill Climbing-Depth first and Breath first, Constraints satisfaction - Related algorithms, Measure of performance and analysis of search algorithms.

UNIT II -REPRESENTATION OF KNOWLEDGE 9

Game playing - Knowledge representation, Knowledge representation using Predicate logic, Introduction to predicate calculus, Resolution, Use of predicate calculus, Knowledge representation using other logic-Structured representation of knowledge.

UNIT III- KNOWLEDGE INFERENCE 9

Knowledge representation -Production based system, Frame based system. Inference – Backward chaining, Forward chaining, Rule value approach, Fuzzy reasoning - Certainty factors, Bayesian Theory-Bayesian Network-Dempster– Shafer theory.

UNIT IV -PLANNING AND MACHINE LEARNING 9

Basic plan generation systems - Strips -Advanced plan generation systems – K strips –Strategic explanations -Why, Why not and how explanations. Learning- Machine learning, adaptive Learning.

UNIT V -EXPERT SYSTEMS 9

Expert systems - Architecture of expert systems, Roles of expert systems - Knowledge Acquisition – Meta knowledge, Heuristics. Typical expert systems - MYCIN, DART, XOON, Expert systems shells

TEXT BOOKS:

1. Kevin Night and Elaine Rich, Nair B., ¯Artificial Intelligence (SIE)‖, McGraw Hill- 2008. (Units-I,II,VI& V)

2. Dan W. Patterson, ¯Introduction to AI and ES‖, Pearson Education, 2007. (Unit-III)

REFERENCES:

1. Peter Jackson, ¯Introduction to Expert Systems‖, 3rd Edition, Pearson Education, 2007.

2. Stuart Russel and Peter Norvig ¯AI – A Modern Approach‖, 2nd Edition, Pearson Education 2007.

3. Deepak Khemani ¯Artificial Intelligence‖, Tata McGraw Hill Education 2013.

4. http://nptel.ac.in

COURSE OUTCOME:

Ability to identify problems that are amenable to solution by AI methods.
Ability to analyze appropriate AI methods to solve a given problem.
Ability to formalize a given problem in the language/framework of different AI methods.
Ability to create basic and advanced plan generation systems and to understand the concepts learning methods.
An ability to understand the concepts of expert systems

UNIT – 1: INTRODUCTION TO AI AND PRODUCTION SYSTEMS

TWO MARKS

1. **What is ridge?(MAY/JUNE2016)**
   Ridge: special kind of local maximum. It is an area of the search space that is higher than the surrounding areas and that itself has the slope.

2. **How much knowledge would be required by a perfect program for the problem of playing chess? Assume that unlimited computing power is available.**
   (MAY/JUNE2016) Knowledge would be required by a perfect by a perfect program.
   - Rules for determining legal moves
   - Control mechanism that implements an appropriate search procedure
   - Good strategy and tactics

3. **What is artificial intelligence?**

The exciting new effort to make computers think machines with minds in the full and literal sense. Artificial intelligence systemizes and automates intellectual tasks and is therefore potentially relevant to any sphere of human intellectual activities.

4. Define Turing test.

The Turing test proposed by Alan Turing was designed to provide a satisfactory operational definition of intelligence. Turing defined intelligent behaviour as the ability to achieve human-level performance in all cognitive tasks, sufficient to fool an interrogator.

5. List the capabilities that a computer should possess for conducting a Turing Test?

The capabilities that a computer should possess for conducting a Turing Test are,

Natural Language Processing;
Knowledge Representation;
Automated Reasoning;
Machine Language.

6. Define an agent.

An agent is anything that can be viewed as perceiving its environment through sensors and acting upon the environment through effectors.

7. Define rational agent. (DEC 2011) (APRIL/MAY 2015)

A rational agent is one that does the right thing. Here right thing is one that will cause agent to be more successful. That leaves us with the problem of deciding how and when to evaluate the agent's success.

8. Define an Omniscient agent.

An omniscient agent knows the actual outcome of its action and can act accordingly; but omniscience is impossible in reality.

9. What are the factors that a rational agent should depend on at any given time?

The factors that a rational agent should depend on at any given time are,

The performance measure that defines criterion of success;
Agent's prior knowledge of the environment;
Action that the agent can perform;
The agent's percept sequence to date.

10. List the measures to determine agent's behaviour.

The measures to determine agent's behaviour are,

Performance measure,
Rationality,
Omniscience, Learning and Autonomy.

11. List the various types of agent programs. (DEC 2012)

The various types of agent programs are,

Simple reflex agent program;
Agent that keep track of the world;
Goal based agent program;
Utility based agent program.

12. What is depth-limited search?

Depth-limited avoids the pitfalls of DFS by imposing a cut off of the maximum depth of a path. This cut-off can be implemented by special depth limited search algorithm or by using the general search algorithm with operators that keep track of the depth.

13. Define breadth-first search.

The breadth-first search strategy is a simple strategy in which the root-node is expanded first, and then all the successors of the root node are expanded, then their successors and so on. It is implemented using TREE-SEARCH with an empty fringe that is a FIFO queue, assuring that the nodes that are visited first will be expanded first.

14. Define problem formulation.

Problem formulation is the process of deciding what actions and states to consider for a goal that has been developed in the first step of problem solving.

15. List the four components of a problem?

The four components of a problem are,

An initial state;
Actions;
Goal test;
Path cost.

16. Define iterative deepening search.

Iterative deepening is a strategy that sidesteps the issue of choosing the best depth limit by trying all possible depth limits: first depth 0, then depth 1, then depth 2& so on.

17. Mention the criteria's for the evaluation of search strategy.(MAY/JUNE 2014)

The criteria's for the evaluation of search strategy are,

Completeness;
Time complexity;
Space complexity;
Optimality.

18. Define the term percept.

The term percept refers to the agents perceptual inputs at any given instant. An agent's percept sequence is the complete history of everything that the agent has perceived.

19. Define Constraint Satisfaction Problem (CSP).

A constraint satisfaction problem is a special kind of problem satisfies some additional structural properties beyond the basic requirements for problem in general. In a CSP, the states are defined by the values of a set of variables and the goal test specifies a set of constraint that the value must obey.

20. List some of the uninformed search techniques.

Some of the uninformed search techniques are,

Breadth-First Search(BFS);
Depth-First Search(DFS);
Uniform Cost Search;
Depth Limited Search; Bidirectional Search.
Iterative Deepening Search;

21. Define Abstraction ? (May 2012)

Abstraction is the process by which data and programs are defined with a representation similar in form to its meaning (semantics), while hiding away the implementation details. Abstraction tries to reduce and factor out details so that the programmer can focus on a few concepts at a time. A system can have several abstraction layers whereby different meanings and amounts of detail are exposed to the programmer. For example, low-level abstraction layers expose details of the computer hardware where the program run, while high-level layers deal with the business logic of the program.

22. What does Software Agent mean? (NOV/DEC 2013)

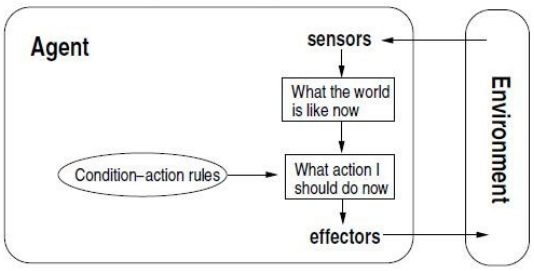
A software agent is an piece of software that functions as an agent for a user or another program, working autonomously and continuously in a particular environment. It is inhibited by other processes and agents, but is also able to learn from its experience in functioning in an environment over a long period of time.

23. Define the effect of heuristic accuracy on performance (DEC 2011)(NOV/DEC 2013)

A heuristic is a method that might not always find the best solution but is guaranteed to find a good solution in reasonable time.

By sacrificing completeness it increases efficiency. Useful in solving tough problems which could not be solved any other way. Solutions take an infinite time or very long time to compute. The classic example of heuristic search methods is the travelling salesman problem.

24.Give the structure of an agent. (MAY/JUNE 2014)



25.Why problem formulation should follow goal formulation ?(APRIL/MAY 2015)

In goal formulation, we decide which aspects of the world we are interested in, and which can be ignored or abstracted away. Then in problem formulation we decide how to manipulate the important aspects (and ignore the others). If we did problem formulation first we would not know what to include and what to leave out. That said, it can happen that there is a cycle of iterations between goal formulation, problem formulation, and problem solving until one arrives at a sufficiently useful and efficient solution.

26. List the components of a learning agent?

The components of a learning agent are,
Learning element;
Performance element;
Critic;
Problem generator.

27. **List out some of the applications of Artificial Intelligence.**

Some of the applications of Artificial Intelligence are,
Autonomous planning and scheduling;
Game playing;
Autonomous control;
Diagnosis; Logistics
planning; Robotics.

PART - B

1. i) Explain the heuristic functions with examples. **(MAY/JUNE2016)**

   ii) Write the algorithm for Generate and test and simple Hill Climbing**. (MAY/JUNE2016).**

2. Solve the given problem. Describe the operators involved in it. **(MAY/JUNE2016)**

Consider a water jug problem: You are given two jugs, a 4-gallon one a 3-gallon one. Neither have any measuring markers on it. There is a pump that can be used to fill the jugs with water. How can you get exactly 2 gallons of water into the 4-gallon jug? Explicit Assumptions: A jug can be filled from the pump, water can be poured out a jug onto the ground, water can be poured from one jug to another and that there are no other measurements and that there are no other measuring devices available.

3. Explain in detail about uninformed search strategies. **(APRIL/MAY 2015)**
4. Write in detail about any two informed search strategies. **(MAY/JUNE 2009). (APRIL/MAY 2015)**
5. Explain AO* algorithm. **(APR/MAY11) (DEC 2012)**
6. Define CSP and Discuss about backtracking search for CSPs. **(May 2013)(APRIL/MAY 2015)**
7. How searching is used to provide solutions and also describe some real world problems? **(May/june2009) (May 2013)**

8. Explain in detail on the characteristics and applications of learning agents. **(APR/MAY11)**
9. Prove that breadth first search is special case of uniform cost search. **(MAY 2012)**
10. Explain in detail learning agent. **(MAY 2013)**
11. Explain Depth-First search. **(MAY 2012)**
12. Describe the various properties of the task environment in AI. **(MAY 2012)**
13. Explain in detail any of the four agent structure. **(DEC 2012)**
14. Describe alpha-beta pruning and its effectiveness. **(MAY/JUNE2009)**
15. Describe briefly the various problem characteristics?
16. Identify the problems encountered during hill climbing and list the ways available to deal with these problems?
17. Describe the process of simulated annealing with example?
18. Illustrate in detail about the constraint satisfaction procedure with example?
19. Show how the steepest accent hill climbing works?
20. Explain in detail about the mean end analysis procedure with example?
21. Prepare the merits and demerits of depth-first and breadth-first search with the algorithm?
22. Evaluate a problem as a state space search with an example?

UNIT – 2: REPRESENTATIOEN OF KNOWLEDGE

TWO MARKS

1. **What is alpha-beta pruning? (MAY/JUNE2016)**

   To handle both maximizing and minimizing players, it is also necessary to modify the branch-and-bound strategy to include two bounds, one for each of the players. Lower bound on the value that a maximizing node ultimately assigned as alpha.
   Upper bound on the value that a minimizing node ultimately assigned as beta.

2. **For the given sentence "All Pompeian's were Romans" write a well formed formula in predicate logic. (MAY/JUNE2016)**

Well formed formula in predicate logic.
All Pompeian's were Romans

$\forall$ **x: Pompeian's(x) => Romans(x)**

3. What are the standard quantifiers of First Order Logic?

The First Order Logic contains two standard quantifiers. They are:

Universal Quantifiers $\exists$

Existential Quantifiers $\forall$

4. Define Universal Quantifier with an example.

To represent ¯All elephants are mammal‖ ¯Raj is an elephant‖ is represented by Elephant(Raj) and ¯Raj is a mammal‖. The first order logic is given by

$\forall$ **X Elephant(x) => Mammal(x)**

$\forall$ Refers to ¯For all‖. P is any logical expression, which is equivalent to the conjunction(i.e. the) of all sentences obtained by substituting the name of an object for the variable x where if appears in p. The above sentence is equivalent to

Elephant(Raj)=>Mammal(Raj)
Elephant(John)=>Mammal(John)

Thus it is true if and only if, all the above sentences are true that is if p is true forall objects x in the universe. Hence, $\forall$ is called universal quantifier.

5. Define Existential Quantifier with an example.

Universal quantification makes statements about every object. Similarly, we can make statement about some object in the universe without naming it, by using an existential quantifier.

To say, for example, that king john has a crown on his head, we write

$\exists$ **X Crown(x) ^ OnHead(x,John)**

X is pronounced ¯There exists an x such that ….‖ Or ¯For some x…‖

The sentence says that P is true for at least one object x. Hence, is called existential quantifier.

6. Define Nested Quantifier with an example.

The Nested Quantifier is to express the more complex sentences using multiple quantifiers. For example, ¯Brothers are siblings‖ can be written as

$\forall x \forall y$ **Brother(x,y)=>Sibling(x,y)**

Consecutive quantifiers of the same type can be written as one quantifier with several variables. For example, to say that siblinghood is a symmetric relationship, we can write

$\forall x,y$ **Sibling(x,y)□Sibling(y,x)**

7. Explain the connections between $\forall$ and $\exists$

The two quantifiers can be connected with each other through negation. It can be explained through negation. It can be explained with the following example.

Eg: $\forall$x Likes(x, Ice Cream) is equivalent to $\exists x \forall$Likes(x, Ice cream)

This means ¯Everyone likes ice cream‖ is equivalent to ¯there is no one who does not like ice cream‖.

8. What is the use of equality symbol?

The equality symbol is used to make the statements more effective that two terms refer to the same object.

Eg: Father (John)=Henry

9. Define Higher Order Logic.

The Higher Order Logic allows quantifying over relations and functions as well as over objects.

Eg: The two objects are equal if and only if, all the properties to them are equivalent.

$\forall$x, y(x=y)□($\forall$p p(x) □p(y))

10. Define First Order Logic.

First Order Logic, a representation language that is far more powerful than propositional logic. First Order Logic commits to the existence of objects and relations.

Eg: One plus two equals three
Objects – one,two& three Relations-
equals Functions-plus

11. What is called declarative approach?

The representation language makes it easy to express the knowledge in the form of sentences. This simplifies the construction problem enormously. This is called as declarative approach.

12. State the aspects of a knowledge representation language.

A knowledge representation language is defined in two aspects:

i) **Syntax:** The syntax of a language describes the possible configuration that canconstitute sentences.

ii)Semantics: It determines the facts in the world to which the sentences refer.

13. What is called entailment?

The generations of new sentences that are necessarily true given the old sentences are true. This relation between sentences is called **entailment.**

14. What is meant by tuple?

A tuple is a collection of objects arranged in a fixed order and is written with angle brackets surrounding the objects.

{<Richard the Lion heart, King John>,<King John, Richard the Lion heart>}

15. What is Prepositional Logic?

Propositional Logic is a declarative language because its semantics is based on a truth relation between sentences and possible worlds. It also has sufficient expressive power to deal with partial information, using disjunction and negation.

16. What is compositionality in propositional logic?

Propositional Logic has a third property that is desirable in representation languages, namely compositionality. In a compositionality language, **the meaning of sentences is a function of the meaning of its parts**. For example, ¯S1^S2‖ is related to the meanings of ¯S1 and S2‖.

17. Define Symbols.

The basic syntactic elements of first order logic are the symbols that stand for objects, relations and functions. The symbols are in three kinds. Constant symbols which stand for objects, Predicate symbols which stand for relations and Function symbol which stand for functions.

18. Define ground term, Inference.

The term without variables is called ground term.
The task of deriving the new sentence form the old is called Inference.

19. Define Data log.

The set fo first order definite clauses with no function symbols is called datalog.

Eg: "The country Nono, an enemy of America" Enemy(Nono, America)

The absence of function symbols makes inference much easier.

20. What is Pattern Matching?

The ¯inner loop‖ of the algorithm involves finding all possible unifiers such that the premise of a rule unifies with a suitable set of facts in the knowledge base. This is called Pattern Matching.

21. What is Data Complexity?

The complexity of inferences is a function of the number of ground facts in the database is called data complexity.

22. Define Prolog.

Prolog programs are sets of definite clauses written in a notation somewhat different

from standard first-order logic.

23. Define conjunctive normal form.

First Order resolution requires that sentences be in conjunctive normal form that is, a conjunction of clauses, where each clause is a disjunction of literals. Literals can contain variables, which are assumed to universally quantified.

For ex, the sentence

 **American(x) ^ Weapon(y) Sells(x,y,z) ^ Hostile(z)=> Criminal(x)**

Becomes, in CNF,

¬ American(x)v¬ Weapon(y)v¬ Sells(x,y,z)v¬ Hostile(z)vCriminal(x)

24. Define Skolemization.

Skolemization is the process of removing existential quantifiers by elimination.

25. What is the other way to deal with equality?

Another way to deal with an additional inference rule is Demodulation Para modulation

26. Define the ontology of situation calculus.

Situations which denote the states resulting from executing actions. This approach is called Situation Calculus.

Situations are logical terms consisting of the initial situation and all situations that are generated by applying an action to a situation.

Fluent are functions and predicates that vary from one situation to the next, such as the location of the agent.

A temporal or **eternal** predicates and functions are also allowed.

27. Define unification. (Dec 2012) (May 2012)

Lifted inference rule require finding substitutions that make different logical expressions look identical (same).this is called unification

28. Distinguish between predicate and propositional logic? (Dec 2011)

Propositional logic (also called sentential logic) is the logic the includes sentence letters (A,B,C) and logical connectives, but not quantifiers. The semantics of propositional logic uses truth assignments to the letters to determine whether a compound propositional sentence is true.

Predicate logic is usually used as a synonym for first-order logic. Syntactically, first-order logic has the same connectives as propositional logic, but it also has variables for individual objects, quantifiers, symbols for functions, and symbols for relations. The semantics include a domain of discourse for the variables and quantifiers to range over, along with interpretations of the relation and function symbols.

29. With an example show objects properties functions and relations. (Dec 2012)

Example Evil king john brother of Richard rules England in 1200‖

Objects: john, Richard, England, 1200

Relation: Brother of

Properties: evil, king

Functions: ruled

30. Define synchronic and diachronic sentence (May 2012)

Sentences dealing with same time are called synchronic sentences, sentences that allow reasoning ¯a cross time‖ are called diachronic sentence

31. **Define Modus Ponen's rule in Propositional logic? (MAY/JUNE 2014)**

The standard patterns of inference that can be applied to derive chains of conclusions that lead to the desired goal is said to be Modus Ponen's rule.

32. What is the significance in using the unification algorithm? (Nov/Dec 2012)

Unification is an algorithmic process of solving equations between symbolic expressions.

PART – B

1. Convert the following well formed formula into clause form with sequence of steps.
   ∀x : [Roman(x) & known(x,Marcus)] → [hate(x, Caeser) v (y: z : hate(y,z)
   ∀ ∃→thinkcrazy(x,y))] **(MAY/JUNE2016)**
2. i) Write the resolution procedure for prepositional logic. **(MAY/JUNE2016)**
   ii) Explain the iterative Deepening Algorithm . **(MAY/JUNE2016)**
3. Discuss forward and backward chaining in detail**. (MAY 2012)(NOV/DEC 2014)(APRIL/MAY 2015)**
4. Discuss in detail about FOL and inferences in FOL**. (APRIL/MAY 2015)(MAY 2012)**
5. What are the steps to convert first order logic sentence to normal form? Explain each step. **(MAY/JUNE 2014)**
6. Differentiate propositional logic with FOL .List the inference rules along with suitable examples for first order logic. **(MAY/JUNE 2014)**
7. Write the algorithm for deciding entailment in propositional logic. **(NOV/DEC 2014)**

8. Explain the completeness proof of resolution. **(NOV/DEC 2014).**
9. Give the Syntax and Semantics of a first order logic in detail with an eg. **(MAY 2013)**

(DEC 2012) (MAY 2013)

10. Explain predicate logic. **(DEC 2012)**
11. Explain unification algorithm used for reasoning under predicate logic with an example. **(APR/MAY11)**
12. Describe in detail the steps involved in the knowledge engineering process. **(APR/MAY11)**
13. Give Syntax and Semantics of a first order logic for a family domain.
14. Give the Syntax and Semantics of a first order logic for Numbers, Sets, Lists domain.
15. Elaborate upon the process of knowledge engineering with electronic circuit's domain.
16. How facts are represented using prepositional logic? Give an example.

UNIT – 3: KNOWLEDGE INFERENCE

TWO MARKS

1. **What is Bayesian Networks? (MAY/JUNE2016)**
   - Graphical model for reasoning under uncertainty
   - Nodes represents variables
   - Arc represents direct connections between variables.

2. **Write the properties of fuzzy sets. (MAY/JUNE2016)**
   - Properties:
   - Distributivity
   - Commutativity
   - Idempotency

3. **What are representations types of knowledge?**
   Four General Representation Types
   - Logical Representations
   - Semantic

   Networks o Production
   Rules o Frames

4. **What is a frame?**
   Frames represent an alternative way to structure and organise knowledge. A frame system is a hierarchy of frames. Each Frame has:
   • a name

• slots: properties of the entity that has the name,and their values.

5. Define Uncertainty.

Uncertainty means that many of the simplifications that are possible with deductive inference are no longer valid.

6. State the reason why first order, logic fails to cope with that the mind like Medical diagnosis.

Three reasons:
Laziness: It is hard to lift complete set of antecedents of consequence, needed to ensure and exception less rule.
Theoretical Ignorance: Medical science has no complete theory for the domain.
Practical ignorance: Even if we know all the rules, we may be uncertain about a particular item needed.

7. What is the need for probability theory in uncertainty?

Probability provides the way of summarizing the uncertainty that comes from our laziness and ignorance. Probability statements do not have quite the same kind of semantics known as evidences.

8. What is the need for utility theory in uncertainty?

Utility theory says that every state has a degree of usefulness, or utility to In agent, and that the agent will prefer states with higher utility. The use utility theory to represent and reason with preferences.

9. What is called As Decision Theory?

Preferences As Expressed by Utilities Are Combined with Probabilities in the General Theory of Rational Decisions Called Decision Theory. Decision Theory = Probability Theory + Utility Theory.

10. Define conditional probability?

Once the agents has obtained some evidence concerning the previously unknown propositions making up the domain conditional or posterior probabilities with the notation
p(A/B) is used. This is important that p(A/B) can only be used when all be is known.

11. When probability distribution is used?

If we want to have probabilities of all the possible values of a random variable probability distribution is used.

Eg:

P(weather) = (0.7,0.2,0.08,0.02). This type of notations simplifies many equations.

12. What is an atomic event?

An atomic event is an assignment of particular values to all variables, in other words,

the complete specifications of the state of domain.

13. Define joint probability distribution.

Joint probability distribution completely specifies an agent's probability assignments to all propositions in the domain. The joint probability distribution p(x1,x2,--------xn) assigns probabilities to all possible atomic events; where x1,x2------xn=variables.

14. What is meant by belief network?

- A belief network is a graph in which the following holds
- A set of random variables
- A set of directive links or arrows connects pairs of nodes. The conditional probability table for each node
- The graph has no directed cycles.

15. What are called as Poly trees?

The algorithm that works only on singly connected networks known Poly trees. Here at most one undirected path between any two nodes is present.

16. What is a multiple connected graph?

A multiple connected graph is one in which two nodes are connected by more than one path.

17. List the three basic classes of algorithms for evaluating multiply connected graphs.

The three basic classes of algorithms for evaluating multiply connected graphs

1. Clustering methods;
2. Conditioning methods;
3. Stochastic simulation methods.

18. What is called as principle of Maximum Expected Utility (MEU)?

The basic idea is that an agent is rational if and only if it chooses the action that yields the highest expected utility, averaged over all the possible outcomes of the action. This is known as MEU

19. What is meant by deterministic nodes?

A deterministic node has its value specified exactly by the values of its parents, with no uncertainty.

20. Give the Baye's rule equation

W.K.T $P(A \wedge B) = P(A/B)\ P(B)$---------- 1

$P(A \wedge B) = P(B/A)\ P(A)$---------- 2

DIVIDING BYE P(A) ; WE GET

$P(B/A) = P(A/B)\ P(B)$

$$\frac{\text{---------}}{P(A)}$$

21. Define Dynamic Belief Network.

A Belief network with one node for each state and sensor variable for each time step is called a Dynamic Belief Network.(DBN)

22. Define Dynamic Decision Network?

A decision network is obtained by adding utility nodes, decision nodes for action in DBN. DDN calculates the expected utility of each decision sequence.

PART-B

1. i) Briefly explain how reasoning is done using fuzzy logic. **(MAY/JUNE 2016)**
   ii) Explain Dempster-Shafer Theory. **(MAY/JUNE 2016)**
2. What is forward chaining and how does it work? Explain the forward chaining algorithm with an example**. (MAY/JUNE2016)**
3. Explain in detail about (1)Temporal models(2)Probabilistic reasoning.**(APRIL/MAY 2015)**
4. How to get the exact inference form Bayesian network. **(MAY/JUNE2014)(APRIL/MAY 2015)**
5. Explain variable elimination algorithm for answering queries on Bayesian networks? **(NOV/DEC 2014)**
6. Define uncertain knowledge, prior probability and conditional probability. State the Bayes' theorem. How it is useful for decision making under uncertainty? Explain belief networks briefly? **(MAY/JUNE 2014)**
7. Explain the method of handling approximate inference in Bayesian networks. **(APR/MAY11)**
8. What is Bayes' rule ?explain how Bayes' rule can be applied to tackle uncertain Knowledge. **(JUNE 07) (MAY 2013)**
9. How to construct Bayesian network. **(MAY 2013)**
10. How to get the approximate inference from Bayesian network. **(DEC 2012)**

UNIT – 4: PLANNING & MACHINE LEARNING

TWO MARK

1. **What is rote learning? (MAY/JUNE2016)**
   Cache has been used in A.I. programs to produce surprising performance improvement. Such

Caching is rote learning. Most basic learning activity.

2. **Brief frame problem. (MAY/JUNE2016)**
   The problem of how to determine which things change and which do not becomes increasingly important as the complexity of the problem state increases.

3. **Define state-space search.**
   The most straightforward approach is to use state-space search. Because the descriptions of actions in a planning problem specify both preconditions and effects, it is possible to search in either direction: either forward from the initial state or backward from the goal

4. **What are the types of state-space search?**
   The types of state-space search are,
   Forward state space search;
   Backward state space search.

5. **What is Partial-Order Planning? (April/May 2015)(Nov/Dec 2013)**
   A set of actions that make up the steps of the plan. These are taken from the set of actions in the planning problem. The ¯empty‖ plan contains just the Start and Finish actions.
   Start has no preconditions and has as its effect all the literals in the initial state of the planning problem. Finish has no effects and has as its preconditions the goal literals of the planning problem.

6. **What are the advantages and disadvantages of Partial-Order Planning?**
   Advantage: Partial-order planning has a clear advantage in being able to decompose problems into sub problems.
   Disadvantage: Disadvantage is that it does not represent states directly, so it is harder to estimate how far a partial-order plan is from achieving a goal.

7. **What is a Planning graph?**
   A Planning graph consists of a sequence of levels that correspond to time steps in the plan where level 0 is the initial state. Each level contains a set of literals and a set of actions.

8. **What is Conditional planning?**
   Conditional planning is also known as contingency planning, conditional planning deals with incomplete information by constructing a conditional plan that accounts for each possible situation or contingency that could arise

9. **What is action monitoring?**
   The process of checking the preconditions of each action as it is executed, rather than checking the preconditions of the entire remaining plan. This is called action monitoring.

10. **Define planning.**
    Planning can be viewed as a type of problem solving in which the agent uses beliefs about actions and their consequences to search for a solution.

11. **List the features of an ideal planner?**
    The features of an ideal planner are,

The planner should be able to represent the states, goals and actions; The planner should be able to add new actions at any time;
The planner should be able to use Divide and Conquer method for solving very big problems.

12. **What are the components that are needed for representing an action?**

The components that are needed for representing an action are,

- Action description
- Precondition
- Effect.

13. **What are the components that are needed for representing a plan?**

The components that are needed for representing a plan are,

- A set of plans steps;
- A set of ordering constraints;
- A set of variable binding constraints; A set of casual link protection.

14. **What are the different types of planning? (May/June 2014)**

The different types of planning are,

- Situation space planning;
- Progressive planning;
- Regressive planning;
- Partial order planning;
- Fully instantiated planning.

15. Define a solution.

A solution is defined as a plan that an agent can execute and that guarantees the achievement of goal.

16. **Define complete plan and consistent plan.(Nov/Dec 2014) (May/June 2013)**

A complete plan is one in which every precondition of every step is achieved by some other step.

A consistent plan is one in which there are no contradictions in the ordering or binding constraints.

17. What are Forward state-space search and Backward state-space search?

Forward state-space search: It searches forward from the initial situation to the goal situation.

Backward state-space search: It searches backward from the goal situation to the initial situation.

18. What is Induction heuristics? What are the different types of induction heuristics?

Induction heuristics is a method, which enable procedures to learn descriptions from positive and negative examples.

There are two different types of induction heuristics. They are:

Require-link heuristics.
Forbid-link heuristics.

19. Define Reification.

The process of treating something abstract and difficult to talk about as though it were concrete and easy to talk about is called as reification.

20. What is reified link?

The elevation of a link to the status of a describable node is a kind of reification. When a link is so elevated then it is said to be a reified link.

21. Define action monitoring.

The process of checking the preconditions of each action as it is executed, rather than checking the preconditions of the entire remaining plan. This is called action monitoring.

22. What is meant by Execution monitoring?

Execution monitoring is related to conditional planning in the following way. An agent that builds a plan and then executes it while watching for errors is, in a sense, taking into account the possible conditions that constitute execution errors.

23. What is meant by learning?

Learning is a goal-directed process of a system that improves the knowledge or the knowledge representation of the system by exploring experience and prior knowledge.

24. Define informational equivalence and computational equivalence.

A transformation from on representation to another causes no loss of information; they can be constructed from each other.

The same information and the same inferences are achieved with the same amount of effort.

25. Define knowledge acquisition and skill refinement.

knowledge acquisition (example: learning physics) — learning new symbolic information coupled with the ability to apply that information in an effective manner

skill refinement (example: riding a bicycle, playing the piano) — occurs at a subconscious level by virtue of repeated practice

26. What is Explanation-Based Learning?

The background knowledge is sufficient to explain the hypothesis of Explanation-Based Learning. The agent does not learn anything factually new from the instance. It extracts general rules from single examples by explaining the examples and generalizing the explanation.

27. Define Knowledge-Based Inductive Learning.

Knowledge-Based Inductive Learning finds inductive hypotheses that explain set of observations with the help of background knowledge.

28. What is truth preserving?

An inference algorithm that derives only entailed sentences is called sound or truth preserving.

29. Define Inductive learning. How the performance of inductive learning algorithms can be

measured?

Learning a function from examples of its inputs and outputs is called inductive learning.

It is measured by their learning curve, which shows the prediction accuracy as a function of the number of observed examples.

30. List the advantages of Decision Trees. The advantages of Decision Trees are,

It is one of the simplest and successful forms of learning algorithm. It serves as a good introduction to the area of inductive learning and is easy to implement.

31. What is the function of Decision Trees?

A decision tree takes as input an object or situation by a set of properties, and outputs a yes/no decision. Decision tree represents Boolean functions.

32. What is the task of reinforcement learning?

The task of reinforcement learning is to use rewards to learn a successful agent function.

33. Define Passive learner and Active learner.

A passive learner watches the world going by, and tries to learn the utility of being in various States.

An active learner acts using the learned information, and can use its problem generator to suggest explorations of unknown portions of the environment.

34. Define supervised learning & unsupervised learning. (Nov/Dec 2014)(May/June 2013)

Any situation in which both inputs and outputs of a component can be perceived is called supervised learning.

Learning when there is no hint at all about the correct outputs is called unsupervised learning.

35. What is reinforcement learning? (Dec 2012) (May 2012)

Reinforcement learning refers to a class of problems in machine learning which postulate an agent exploring an environment in which the agent perceives its current state and takes actions. The environment, in return, provides a reward (which can be positive or negative). Reinforcement learning algorithms attempt to find a policy for maximizing cumulative reward for the agent over the curse of the problem.

.

PART-B

1. i) Describe the components of a planning system. **(May/June 2016)**
   ii)What is ID3? Write the drawback of ID3? **(May/June 2016)**
2. i)Describe the hierarchical planning method with an example. **(May/June 2016)**

ii) Describe the Learning with Marco-Operators. **(May/June 2016)**

3. Explain in detail about active and passive reinforcement learning**. (April/May 2015)**
4. Explain Planning with state space search with an example. **(May 2013)(April/May 2015)**
5. Explain with example learning in decision trees. **(May/June2009)**
6. Explain partial order planning with example. **(Dec 2012) (May 2012) (May 2013)**
7. Explain Graph Plan algorithm with the example. **(Dec 2012) (May 2013)**
8. What is STRIPS explain in detail with the example.
9. How we plan and act in non deterministic domains.
10. What is conditional planning **(May 2013).**
11. Give an example for partial order planning.
12. For Blocks World explain STRIPS. **(May 2013)**
13. Compare STRIPS and ADL language.
14. Explain the concept of planning with state space search using suitable examples.
15. Explain the use of planning graphs in providing better heuristic estimates with examples.
16. Explain the various forms of learning.
17. List the machine learning algorithms in detail.
18. Explain adaptive learning in detail.
19. Write short notes on the following

    i.Conditional planning

    ii.Execution monitoring and replanning

    iii. Continuous planning

    iv. Multiagent planning

UNIT-5 –EXPERT SYSTEMS

1. **What is meta knowledge? How meta knowledge is represents in rule-based expert systems? (MAY/JUNE2016)**
   Meta Knowledge: Knowledge about knowledge
   Meta Knowledge is represented in rule-based expert systems as meta rules.

2. **Write any four earliest expert systems. (MAY/JUNE2016)**
   Four earliest expert systems:
   DENDRAL, MYCIN, PROSPECTOR, XCON

3. **Define an expert system.**

   An expert system is a computer program that attempts to mimic human experts by the system's capability to render advice, to teach and execute intelligent tasks.

4. **Define natural language processing.**

   Natural language processing is a program that permits (to a certain degree) a human-computer dialogue in a conversational, day-to-day language (a natural language like English, French, or Dutch).

5. **Define speech recognition and understanding. Why is it useful?**

Speech or voice recognition is a data input method. For example, the computer recognizes and understands one (or a few) word commands. Speech understanding on the other hand is the computer's ability to understand a spoken language. That is, the computer understands the meaning of sentences and paragraphs through syntax and semantics.

6. **Define an intelligent agent. Why is it useful?**

An intelligent agent is a program that runs in the background and learns your patterns, like any other agent working for you. It learns your needs to serve you better. The little paperclip guy that shows up in Microsoft Word is an example of an intelligent agent.

7. **List the major benefits of intelligent computer-aided instruction.**

The major benefits are:
Individualized, self-adjusted level of instruction; immediate feedback; portability; consistency; and better control of updating and variety of presentations.

8. **Define the ES development environment and contrast it with the consultation environment.**

The development environment includes the activities and support that are necessary to acquire and represent the knowledge as well as to make inferences and provide explanations. The major players in this environment are the knowledge engineer and the domain expert who act as builders. Once the system is completed it is used for consultation by the nonexpert user via the consultation environment.

9. **List and define the major components of an ES.**

The major components are:

a. **Knowledge base**--the software that represents the knowledge.
b. **Inference engine**--the reasoning mechanism.
c. **User interface**--the hardware and software that provide the dialogue between people and the computer.

Domain expert--the individual who is considered an expert.
Knowledge engineer--the individual who acquires and represents the knowledge.

d. **Explanation facility**--the software that answers questions such as "Why" and How." **Blackboard**--a workplace for storing and working on intermediate information.
e. **Reasoning improvement**--a facility (not available commercially) for improving the reasoning capabilities of an ES.
f. **User**--the non-expert who uses the machine for consultation. **Hardware**--the hardware that is needed to support the ES.

10. **What is the role of a knowledge engineer?**

Major duties are to acquire and represent the knowledge. Some knowledge engineers do the computer programming as well.

11. Describe how expert systems perform inference.

The brain of an expert system is the inference engine that provides a methodology for reasoning about information in the knowledge base. Inference can be performed using semantics networks, production rules, and logic statements.

12. What are the major activities performed in the ES blackboard (workplace)?

The blackboard records intermediate hypotheses and decisions, devises a plan of how to attack a problem, provides an agenda of actions awaiting execution, and lists the candidate solutions to be examined.

13. Describe generic categories of ES applications.

Genetic categories of ES applications are:
Rule-based ES. Knowledge is represented by a series of rules.
Frame-based systems. Knowledge is represented as a series of frames (an object-oriented approach).
Hybrid systems. Involve several approaches such as fuzzy logic and neural networks.
Model-based systems. Structured around a model that simulates the structure and function of thesystem under study.
Ready-made systems. Utilize prepackaged software.
Real-time systems. Systems designed to produce a just-in-time response.

14. Describe some of the limitations of ES.

Some of the limitations are:
Knowledge is not always readily available.

a. It can be difficult to extract expertise from humans. There are frequently multiple correct assessments. Time pressures.

Users have cognitive limits.
ES works well only within a narrow domain of knowledge.

b. Most experts do not have an independent means to validate results. Vocabulary is often limited and difficult to understand.
c. Help from knowledge engineers is difficult to obtain and costly. Potential for lack of trust on the part of the end-users.

Knowledge transfer is subject to biases.

15. Describe the success factors of ES.

Success factors are:

- Level of knowledge must be sufficiently high.
- Expertise must be available from at least one expert.
- The problem to be solved must by fuzzy.
- The problem must be narrow in scope.

- The shell must be of high quality and naturally store and manipulate the knowledge. o The user interface must be friendly to novice users.
- The problem to be solved must be difficult and important enough to justify the development of a system.
- Knowledgeable developers with good people skills are needed. The impact of the ES must be considered.
- The impact should be favourable. Management support is needed.

16. What is a ready-made (off-the-shelf) ES?

Ready-made systems are sold in computer stores or via the Web to people who want to buy general expertise in a certain area. For example, the system WINE advises a user what wine is most appropriate for certain types of meals.

17. What is a real-time ES?

In real-time ES the conclusions (recommendations) are derived fast so a process can be impacted immediately. They are used in quality control and robotics (e.g., to correct a malfunction).

18. What are the benefits of deploying an ES on the Web?

The expertise is made available to a wider audience, use of a common interface, etc.

19. How can an ES help a decision maker in Web use?

An ES can advise a user on how to proceed in doing his/her work (Web searches, database access, etc.).

PART-B

1. Explain the basic components of an expert system. **(MAY/JUNE 2016)**
2. Write any six applications of expert systems. **(MAY/JUNE 2016)**
3. Explain about the knowledge acquisition. **(MAY/JUNE 2016)**
4. Write the characteristic features of expert systems. **(MAY/JUNE 2016)**
5. Explain the various stages of expert system development?
6. Explain the roles of an expert system?
7. Explain expert system work at knowledge engineering companies?
8. What is meant by high performance expert system? How is it used in research and in business?
9. Write notes on MYCIN, DART, and XCON.
10. Write notes on expert systems shells.