## Lesson Plan & Schedule (CS6659 – ARTIFICIAL INTELIGENCE)

<table>
<thead>
<tr>
<th>Unit</th>
<th>Task</th>
<th>No of Hours Required</th>
<th>Actual hours Taken, Reason for the deviation</th>
<th>Teaching Method BB/PP/Video</th>
<th>Book</th>
<th>Dependencies (page no.)</th>
<th>Remarks</th>
<th>No of Students Present</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIT I (9 hours)</td>
<td>Introduction about the subject, Concepts of AI, Concepts of Expert Systems</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>R4</td>
<td><a href="http://nptel.ac.in">http://nptel.ac.in</a></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Introduction to AI, Problem formulation, Production systems, Control strategies, Search strategies</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>4 – 11,25 – 30</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Problem characteristics</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>36 – 43</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Production system characteristics, Specialized production system</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>43 – 45</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Problem solving methods, Problem graphs</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>47 – 48</td>
<td>Assignment-I</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Matching</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>138 – 142</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Indexing and Heuristic functions, Hill Climbing, Depth first and Breath first</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>138, 52 – 57, 31 – 36</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Constraints satisfaction, Related algorithms, Measure of performance</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>68 – 72</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tutorial (Test)</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Indexing and Heuristic functions</td>
<td></td>
</tr>
</tbody>
</table>
### Lesson Plan & Schedule (CS6659 – ARTIFICIAL INTELLIGENCE)

<table>
<thead>
<tr>
<th>Unit</th>
<th>Task</th>
<th>No of Hours Required</th>
<th>Actual hours Taken, Reason for the deviation</th>
<th>Teaching Method</th>
<th>Book</th>
<th>Dependencies (page no.)</th>
<th>Remarks</th>
<th>No of Students Present</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIT II</td>
<td>Game playing: The Minimax Search Procedure, Adding Alpha-Beta</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>231-239</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Additional Refinements, Iterative Deepening</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>240-246</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Knowledge representation</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>79 – 97</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Knowledge representation using Predicate logic</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>98 – 125</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Introduction to predicate calculus</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>56</td>
<td>Assignment-II</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Resolution</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>108 – 121</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Use of predicate calculus</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>56</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Knowledge representation using other logic</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>113</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Structured representation of knowledge</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>212</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Lesson Plan & Schedule (CS6659 – ARTIFICIAL INTELLIGENCE)

<table>
<thead>
<tr>
<th>Unit</th>
<th>Task</th>
<th>No of Hours Required</th>
<th>Actual hours Taken, Reason for the deviation</th>
<th>Teaching Method</th>
<th>Book</th>
<th>Dependencies (page no.)</th>
<th>Remarks</th>
<th>No of Students Present</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Knowledge representation</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>79 – 97</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Production based system, Frame based system</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>136 – 139</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Inference</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>65 – 66</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Backward chaining, Forward chaining</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>R2</td>
<td>343 – 346</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rule value approach</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>R2</td>
<td>75 – 76</td>
<td>Assignment-III</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fuzzy reasoning - Certainty factors</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>121</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bayesian Theory-Bayesian Network</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>112 – 113</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dempster-Shafer Theory: Classical Probability Theory</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>114 – 116</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dempster -Shafer Theory: Belief Function</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>115-116</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Lesson Plan & Schedule (CS6659 – ARTIFICIAL INTELLIGENCE)

<table>
<thead>
<tr>
<th>Unit</th>
<th>Task</th>
<th>No of Hours Required</th>
<th>Actual hours Taken, Reason for the deviation</th>
<th>Teaching Method</th>
<th>Book</th>
<th>Dependencies (page no.)</th>
<th>Remarks</th>
<th>No of Students Present</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit IV (9 Hours)</td>
<td>Basic plan generation systems</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>240-246</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Strips</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>WS1</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Advanced plan generation systems</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>WS2</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>K strips</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>WS7</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Strategic explanations</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>WS3</td>
<td>*</td>
<td>Assignment-IV</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Why, Why not and how explanations</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>WS8</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Learning- Machine learning</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>WS4</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rote Learning</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>WS5</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Adaptive Learning</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>WS6</td>
<td>*</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Lesson Plan & Schedule (CS6659 – ARTIFICIAL INTELLIGENCE)

<table>
<thead>
<tr>
<th>Unit</th>
<th>Task</th>
<th>No of Hours Required</th>
<th>Actual hours Taken, Reason for the deviation</th>
<th>Teaching Method</th>
<th>Book</th>
<th>Dependencies (page no.)</th>
<th>Remarks</th>
<th>No of Students Present</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit V</td>
<td>Expert systems - Architecture of expert systems</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>326 – 337</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Roles of expert systems</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>326 – 337</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Knowledge Acquisition</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>347 – 349</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Meta knowledge</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>12, 336</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Heuristics</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T1</td>
<td>138, 52 – 57, 31 – 36</td>
<td>Assignment-V</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Typical expert systems – MYCIN</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>76, 120, 328</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Typical expert systems – DART</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Typical expert systems - XOON</td>
<td>1</td>
<td></td>
<td>BB</td>
<td>T2</td>
<td>212, 328</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Name of the Department: Computer Science and Engineering
Year/Semester: III/VI

Name of the Faculty: Miss. Adlin.S

Lesson Plan & Schedule (CS6659 – ARTIFICIAL INTELLIGENCE)

<table>
<thead>
<tr>
<th>Lesson Plan</th>
<th>T1</th>
<th>T2</th>
<th>424</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expert systems shells</td>
<td>1</td>
<td>BB</td>
<td>T2</td>
</tr>
<tr>
<td>Tutorial (Test)</td>
<td>1</td>
<td></td>
<td>Roles of expert systems</td>
</tr>
</tbody>
</table>

(Note: BB= Black Board, PP=Power Point Presentation, VIDEO = Video Lecture, WS=Web Sources)

Text Book:


Reference Book:

4. [http://nptel.ac.in/](http://nptel.ac.in/)
Lesson Plan & Schedule (CS6659 – ARTIFICIAL INTELLIGENCE)

Web sources:

5. www.cs.columbia.edu/~kathy/cs4701/documents/Learning1-print.ppt
8. www.cs.uu.nl/docs/vakken/exp/Slides/XS1ExpertSystemsAndAI.ppt
St. Joseph College of Engineering
Sriperumbudur, Chennai – 602 117
EVEN SEMESTER (2017 - 2018)

Name of the Department: Computer Science and Engineering
Year /Semester: III / VI

Lesson Plan & Schedule (CS6659 – ARTIFICIAL INTELLIGENCE)

Signature of the Staff

Signature of the HOD

Signature of the Principal
CS6659-ARTIFICIAL INTELLIGENCE

UNIT 1
INTRODUCTION TO AI AND PRODUCTION SYSTEMS

PART – A

1. What is AI?
   Artificial Intelligence is the branch of computer science concerned with making computers behave like humans.
   - Systems that think like humans
   - Systems that act like humans
   - Systems that think rationally
   - Systems that act rationally

2. Define an agent.
   An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.

3. What is a task environment? How it is specified?
   Task environments are essentially the "problems" to which rational agents are the "solutions". A Task environment is specified using PEAS (Performance, Environment, Actuators, and Sensors) description.

4. List the properties of task environments.
   - Fully observable vs. partially observable.
   - Deterministic vs. stochastic.
   - Episodic vs sequential
   - Static vs dynamic.
   - Discrete vs. continuous.
   - Single agent vs. multiagent.

5. What are the four different kinds of agent programs?
   - Simple reflex agents;
   - Model-based reflex agents;
   - Goal-based agents; and
   - Utility-based agents.
6. **Explain goal based reflex agent.**
Knowing about the current state of the environment is not always enough to decide what to do. For example, at a road junction, the taxi can turn left, turn right, or go straight on. The correct decision depends on where the taxi is trying to get to. In other words, as well as a current state description, the agent needs some sort of goal information that describes situations that are desirable—for example, being at the passenger’s destination.

7. **What are utility based agents?**
Goals alone are not really enough to generate high-quality behavior in most environments. For example, there are many action sequences that will get the taxi to its destination (thereby achieving the goal) but some are quicker, safer, more reliable, or cheaper than others. A utility function maps a state (or a sequence of states) onto a real number, which describes the associated degree of happiness.

8. **Define the problem solving agent.**
A Problem solving agent is a goal-based agent. It decides what to do by finding sequence of actions that lead to desirable states. The agent can adopt a goal and aim at satisfying it. Goal formulation is the first step in problem solving.

9. **List the steps involved in simple problem solving agent.**
   - Goal formulation
   - Problem formulation
   - Search
   - Search Algorithm
   - Execution phase

10. **Define search and search algorithm.**
The process of looking for sequences actions from the current state to reach the goal state is called search. The search algorithm takes a problem as input and returns a solution in the form of action sequence. Once a solution is found, the execution phase consists of carrying out the recommended action.

11. **What are the components of well-defined problems?**
   - The initial state that the agent starts in. The initial state for our agent of example problem is described by $In(Arad)$
   - A Successor Function returns the possible actions available to the agent. Given a state $successor-FN(x)$ returns a set of \{action, successor\} ordered pairs where each action is one of the legal actions in state $x$, and each successor is a state that can be reached from $x$ by applying the action.
   - For example, from the state $In(Arad)$, the successor function for the Romania problem would return
     \[
     \{ [Go(Sibiu), In(Sibiu)], [Go(Timisoara), In(Timisoara)], [Go(Zerind), In(Zerind)] \}
     \]
   - The goal test determines whether the given state is a goal state.
   - A path cost function assigns numeric cost to each action. For the Romania problem the cost of path might be its length in kilometers.
12. Give examples of real world problems.
   - Touring problems
   - Travelling Salesperson Problem (TSP)
   - VLSI layout
   - Robot navigation
   - Automatic assembly sequencing
   - Internet searching

13. List the criteria to measure the performance of different search strategies.
   - Completeness: Is the algorithm guaranteed to find a solution when there is one?
   - Optimality: Does the strategy find the optimal solution?
   - Time complexity: How long does it take to find a solution?
   - Space complexity: How much memory is needed to perform the search?

    Best-first search is an instance of the general TREE-SEARCH or GRAPH-SEARCH algorithm in which a node is selected for expansion based on the evaluation function \( f(n) \). Traditionally, the node with the lowest evaluation function is selected for expansion.

15. What is a heuristic function?
    A heuristic function or simply a heuristic is a function that ranks alternatives in various search algorithms at each branching step basing on available information in order to make a decision which branch is to be followed during a search.

    For example, for shortest path problems, a heuristic is a function, \( h(n) \) defined on the nodes of a search tree, which serves as an estimate of the cost of the cheapest path from that node to the goal node. Heuristics are used by informed search algorithms such as Greedy best-first search and A* to choose the best node to explore.

16. What are relaxed problems?
    - A problem with fewer restrictions on the actions is called a relaxed problem
    - The cost of an optimal solution to a relaxed problem is an admissible heuristic for the original problem
    - If the rules of the 8-puzzle are relaxed so that a tile can move anywhere, then \( hoop(n) \) gives the shortest solution
    - If the rules are relaxed so that a tile can move to any adjacent square, then \( hmd(n) \) gives the shortest solution

17. What is greedy best-first-search?
    Greedy best-first-search tries to expand the node that is closest to the goal, on the grounds that is likely to lead to a solution quickly. For example, it evaluates nodes by using just the heuristic function: \( f(n) = h(n) \)

18. What is A* search?
    A* search is the most widely-known form of best-first search. It evaluates the
nodes by combining \( g(n) \), the cost to reach the node, and \( h(n) \), the cost to get from the node to the goal:

\[
f(n) = g(n) + h(n)
\]

Where \( f(n) \) = estimated cost of the cheapest solution through \( n \).

\( g(n) \) is the path cost from the start node to node \( n \).

\( h(n) \) = heuristic function

\( A^* \) search is both complete and optimal.

19. What is Hill-climbing search?

The Hill-climbing algorithm is simply a loop that continually moves in the direction of increasing value—that is uphill. It terminates when it reaches a “peak” where no neighbor has a higher value. The algorithm does not maintain a search tree so the current node data structure need only record the state and its objective function value. Hill-climbing does not look ahead beyond the immediate neighbors of the current state.

20. What are local search algorithms?

Local search algorithms operate using a single current state (rather than multiple paths) and generally move only to neighbors of that state. The local search algorithms are not systematic. The key two advantages are (i) they use very little memory—usually a constant amount, and (ii) they can often find reasonable solutions in large or infinite (continuous) state spaces for which systematic algorithms are unsuitable.

PART - B

1. Explain brute force’s algorithm.
2. Explain AI application areas in detail.
3. Explain depth limited search algorithm in detail.
4. Explain Water jug problem with an example.
5. Explain various search strategies.
6. What is an agent? Explain the basic kinds of agents program (Nov/Dec 2014)
7. Explain the components necessary to define a problem (Nov/Dec 2014)
8. What is depth limited search? Give the implementation of depth limited search (Nov/Dec 2014)
10. Explain in detail the structure of different intelligent agents. (Nov/Dec 2012)
12. What are the five uninformed search strategies? Explain any two in detail with Eg. (Nov/Dec 2013)
13. Explain the approach of formulation for constraint satisfaction problems with Eg. (Nov/Dec 2013)
15. Discuss about constraint satisfaction problem. (April/May 2015)
   a. Depth first search
   b. Iterative Deepening Depth First Search
c. Bidirectional Search

17. Explain the Heuristic functions with examples. (May/June 2016)
18. Write the algorithm for Generate and Test and Simple Hill Climbing. (May/June 2016)
19. Solve the given problem. Describe the operators involved in it.
20. Consider a water jug problem: you are given two jugs, a 4 gallon one and a 3-gallon one. Neither has 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 of a jug onto the ground, water can be poured from one jug to another and that there are no other measuring devices available (May/June 2016).

UNIT II

REPRESENTATION OF KNOWLEDGE

PART – A

1. What are the limitations of simple reflex agents?
   Reflex agents are also unable to avoid infinite loops in Wumpus world. A pure reflex agent cannot know for sure when to climb, because neither having the gold nor being in the start square is part of the percept; they are things the agent knows by forming a representation of the world.

2. What is Situation Calculus?
   A situation is a snapshot of the world at an interval of time during which nothing changes:
   • Every true or false statement is made with respect to a particular situation.
   – Add situation variables to every predicate.
   – at(hunter,1,1) becomes at(hunter,1,1,s0): at(hunter,1,1) is true in situation (i.e., state) s0.
   • Example: The action agent-walks-to-location-y could be represented by
     \[(x)(y)(s) \text{ at(Agent,x,s)} \land \lnot \text{onbox(s)} \Rightarrow \text{at(Agent,y, \text{result(walk(y),s)}})\]

3. What are Diagnostic rules?
   Diagnostic rules infer the presence of hidden properties directly from the percept-derived information. We have already seen two diagnostic rules:
   \[(A,l,s) \text{ at(Agent,l,s)} \land \text{Breeze(s)} \Rightarrow \text{Breezy(l)} \quad (A,l,s) \text{ at(Agent,l,s)} \land \text{Stench(s)} \Rightarrow \text{Smelly(l)}\]

4. Give an example rule for Goal Based Agent.
   Once the gold is found, it is necessary to change strategies. So now we need a new set of action values. We could encode this as a rule:
   a. (s) Holding (Gold,s) \Rightarrow GoalLocation([1,1],s)

5. What are the components of Propositional Logic?
   a. Logical constants: true, false
   b. Propositional symbols: P, Q, S, ... (atomic sentences)
   c. Wrapping parentheses: (...) 
   d. Sentences are combined by connectives:
   e. and [conjunction]
   ...or [disjunction]
   ...implies [implication / conditional]
   ..is equivalent [biconditional]
6. **What is Universal Quantification? Universal quantification**

   \[(x)P(x)\] means that \(P\) holds for all values of \(x\) in the domain associated with that variable \(b\). E.g., \((x)\) dolphin(x) mammal(x)

7. **What is Existential Quantification?**
   a. \((x)\) \(P(x)\) means that \(P\) holds for some value of \(x\) in the domain associated with that variable
   b. E.g., \((x)\) mammal(x) lays-eggs(x)
   c. Permits one to make a statement about some object without naming it Connections between all and Exists.

8. **Define a knowledge Base**

   Knowledge base is the central component of knowledge base agent and it is described as a set of representations of facts about the world.

9. **Define a Sentence?**

   Each individual representation of facts is called a sentence. The sentences are expressed in a language called as knowledge representation language.

10. **Define an inference procedure**

    An inference procedure reports whether or not a sentence is entitled by knowledge base provided a knowledge base and a sentence. An inference procedure ‘\(i\)’ can be described by the sentences that it can derive.
    If \(i\) can derive from knowledge base, we can write. \(\text{KBAlpha is derived from KB or } i\) derives alpha from KB.

11. **Define Modus Ponen’s rule in Propositional logic?**

    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.

    In propositional logic, modus ponendo ponens (Latin for "the way that affirms by affirming"; generally abbreviated to MP or modus ponens) or implication elimination is a valid, simple argument form and rule of inference. It can be summarized as "\(P\) implies \(Q\); \(P\) is asserted to be true, so therefore \(Q\) must be true."

12. **Define atomic sentence and complex sentence (Nov/Dec2014)**

    i. An atomic sentence (which has value true or false) is an \(n\)-place predicate of \(n\) terms
    ii. A complex sentence is formed from atomic sentences connected by the logical connectives:
    iii. \(\Box P, P \Box Q, P \Box Q, P \Box Q, P \Box Q\) where \(P\) and \(Q\) are sentences


    The idea of unification is to describe values by logical equations which can be resolved automatically by some unification algorithm. First-order unification is a form of unification whose values are trees. There are several kinds of first-order unification which differ in the choice of the notion of trees. Trees may be ground terms (tuples) or feature trees (records), they may be finite or infinite. First-order unification for feature trees is called feature unification.
14. Distinguish between predicate and propositional logic. (November/December 2011)
(November/December 2015)

Propositional logic (also called sentential logic) is the logic that 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, but sometimes it is used to refers to other logics that have similar syntax. 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.

15. What factors justify whether the reasoning is to be done in forward or backward reasoning?
(November/December 2011)

Selection of forward reasoning or backward reasoning depends on which direction offers less branching factor and justifies its reasoning process to the user. Most of the search techniques can be used to search either forward or backward. One exception is the means-ends analysis technique which proceeds by reducing differences between current and goal states, sometimes reasoning forward and sometimes backward.


Alpha–beta pruning is a search algorithm that seeks to decrease the number of nodes that are evaluated by the minimax algorithm in its search tree. It is an adversarial search algorithm used commonly for machine playing of two-player games (Tic-tac-toe, Go, etc.).

17. Define Syntax?

Syntax is the arrangement of words. Syntax of knowledge describes the possible configurations that can constitute sentences. Syntax of the language describes how to make sentences.

18. Define Semantics

The semantics of the language defines the truth of each sentence with Respect to each possible world. With this semantics, when a particular configuration exists within an agent, the agent believes the corresponding sentence.

19. Define Logic

Logic is one which consist of
i. A formal system for describing states of affairs, consisting of a) Syntax b) Semantics.
ii. Proof Theory – a set of rules for deducing the entailment of a set sentences.

20. What is entailment?

The relation between sentences is called entailment. The formal definition of entailment is this: if and only if in every model in which is true, is also true or if is true then must also be true.

PART- B
1. Explain alpha-beta pruning algorithm with its procedure.
2. Explain the resolution for first order logic and inference rule
3. Illustrate the use of first-order-logic to represent the knowledge.
4. Explain the unification algorithm with an example.
5. Explain the unification algorithm with an example.
6. Write the algorithm for deciding entailment in propositional logic (Nov/Dec 2014)
7. Explain standard quantifiers of first order logic with an example (Nov/Dec 2014)
8. Explain the forward chaining algorithm with the help of the pseudo code (Nov/Dec 2014)
9. Give the completeness proof of resolution (Nov/Dec 2014)
10. Consider the following facts and represent them in predicate form (Nov/Dec 2012)

   (Nov/Dec 2015)
   - F1 There are 500 employees in ABC Company
   - F2 Employees earning more than Rs.5000 pay tax
   - F3 John is a manager in ABC Company
   - F4 Manager earns Rs 10,000

   Convert the facts in predicate form to clauses and then prove by resolution “John pays tax”
11. Explain with an example the use of unification algorithm to prove the concept of resolution (Nov/Dec 2012)
12. Explain the forward chaining process and efficient forward chaining with example. State its usage (Nov/Dec 2013)
13. State and explain the various steps in knowledge engineering process (Nov/Dec 2013)
15. Illustrate the use of first order logic to represent knowledge (April/May 2015)

UNIT III
KNOWLEDGE INFERENCE
PART-A

1. What is knowledge based agents?
   The central component of a knowledge-based agent is its knowledge base, or KB. Informally, a knowledge base is a set of sentences. Each sentence is expressed in a language called a knowledge representation language and represents some assertion about the world. It takes a percept as input and returns an action. The agent maintains a knowledge base, KB, which may initially contain some background knowledge. Each time the agent program is called, it does three things. First, it TELLS the knowledge base what it perceives. Second, it ASKS the knowledge base what action it should perform. In the process of answering this query, extensive reasoning may be done about the current state of the world, about the outcomes of possible action sequences, and so on.

2. Define First order Logic?
   Whereas propositional logic assumes the world contains facts,
   First-order logic (like natural language) assumes the world contains
Objects: people, houses, numbers, colors, baseball games, wars,

Relations: red, round, prime, brother of, bigger than, part of, comes between,

Functions: father of, best friend, one more than, plus,

3. **What are the steps associated with the knowledge Engineering process?**

Discuss them by applying the steps to any real world application of your choice.

Knowledge Engineering: The general process of knowledge base construction a process is called knowledge engineering.

A knowledge engineer is someone who investigates a particular domain, learns what concepts are important in that domain, and creates a formal representation of the objects and relations in the domain. We will illustrate the knowledge engineering process in an electronic circuit domain that should already be fairly familiar.

The steps associated with the knowledge engineering process are:

- Identify the task.
- Assemble the relevant knowledge.
- Decide on a vocabulary of predicates, functions, and constants.
- Encode general knowledge about the domain.
- Encode a description of the specific problem instances.
- Pose queries to the inference procedure and get answers.
- Debug the knowledge base.

4. **Give examples on usage of First Order Logic.**

The best way to find usage of First order logic is through examples. The examples can be taken from some simple domains. In knowledge representation, a domain is just some part of the world about which we wish to express some knowledge.

5. **What is forward chaining?**

Using a deduction to reach a conclusion from a set of antecedents is called forward chaining. In other words, the system starts from a set of facts, and a set of rules, and tries to find the way of using these rules and facts to deduce a conclusion or come up with a suitable course of action. This is known as data driven reasoning.

6. **What is backward chaining?**

Forward chaining applies a set of rules and facts to deduce whatever conclusions can be derived.

In backward chaining, we start from a conclusion, which is the hypothesis we wish to prove, and we aim to show how that conclusion can be reached from the rules and facts in the data base.

The conclusion we are aiming to prove is called a goal and the reasoning in this way is known as goal-driven.
7. **What is Ontological Engineering?**
   Ontology refers to organizing everything in the world into hierarch of categories. Representing the abstract concepts such as Actions, Time, Physical Objects, and Beliefs is called Ontological Engineering.

8. **How categories are useful in Knowledge representation?**
   **Categories and Objects:** The organization of objects into categories is a vital part of knowledge representation. Although interaction with the world takes place at the level of individual objects, much reasoning takes place at the level of categories.

9. **What is taxonomy?**
   Subclass relations organize categories into a taxonomy, or taxonomic hierarchy. Taxonomies have been used explicitly for centuries in technical fields. For example, systematic biology aims to provide a taxonomy of all living and extinct species; library science has developed a taxonomy of all fields of knowledge, encoded as the Dewey Decimal system.

10. **Explain the Ontology of Situation calculus.**
    Situations are logical terms consisting of the initial situation (usually called $S_0$) and all situations that are generated by applying an action to a situation. The function $Result(a, s)$ (sometimes called $Do$) names the situation that results when action $a$ is executed in situation $s$. Figure 10.2 illustrates this idea.
    
    Fluenst are functions and predicates that vary from one situation to the next, such as the location of the agent or the aliveness of the wumpus. The dictionary says a fluent is something that flows, like a liquid. In this use, it means flowing or changing across situations. By convention, the situation is always the last argument of a fluent. For example, $Holdzng(G1, S_0)$ says that the agent is not holding the gold $G1$ in the initial situation $S_0$. $Age(Wumpus, S_0)$ refers to the wumpus's age in $S_0$. A temporal or eternal predicates and functions are also allowed. Examples include the predicate Gold ($G1$) and the function $Leg Of(Wumpus)$.

11. **What is event calculus?**
   Event calculus Situation calculus works well when there is a single agent performing instantaneous, discrete actions. When actions have duration and can overlap with each other, situation calculus becomes somewhat awkward. Therefore, we will cover those topics with an alternative for- EVENTCALCULUS malism known as event calculus, which is based on points in time rather than on situations.

12. **What are semantic networks?**
    Semantic networks are capable of representing individual objects, categories of objects, and relation among objects. Objects or Category names are represented in ovals and are connected by labeled arcs.

13. **Define consistent plan?** (Nov/Dec2014)
    Consistent Planning
    - Data is stored at the most detailed level of the planning hierarchy.
    - Planning levels are interdependent: changes made at one planning level immediately affect all other planning levels. The system performs aggregation and disaggregation at
Advantages of Consistent Planning
- Ease of use: you enter planning figures at one level and can rely on data consistency at all other levels.


An extension of critical path scheduling to a case where there are alternate methods for completing a project is introduced. Three solution methods are presented including one based on the heuristic search algorithm used in artificial intelligence.

15. Define the bi-directed search? (Nov/Dec 2013)

Bidirectional search is a graph search algorithm that finds a shortest path from an initial vertex to a goal vertex in a directed graph. It runs two simultaneous searches: one forward from the initial state and one backward from the goal, stopping when the two meet in the middle.

16. What are continuous random variables? (Nov/Dec 2013)

A continuous random variable is a random variable where the data can take infinitely many values. For example, a random variable measuring the time taken for something to be done is continuous since there are an infinite number of possible times that can be taken.

17. Define knowledge acquisition. (April/May 2015)

The knowledge engineer might already be an expert in the domain, or might need to work with real experts to extract what they know—a process called knowledge acquisition.

18. Define First Order Logic.

- First-order logic (FOL) models the world in terms of
  - Objects, which are things with individual identities
  - Properties of objects that distinguish them from other objects
  - Relations that hold among sets of objects
  - Functions, which are a subset of relations where there is only one "value" for any given "input"
- Examples:
  - Objects: Students, lectures, companies, cars...
  - Relations: Brother-of, bigger-than, outside, part-of, has-color, occurs-after, owns, visits, precedes.
  - Properties: blue, oval, even, large
  - Functions: father-of, best-friend, second-half, one-more-than...

19. What are the three levels in describing knowledge based agent?

- Logical level
- Implementation level
- Knowledge level or epistemological level
20. State the expressiveness extension (Nov/Dec 2013)

With Semantic Expressiveness it is possible to use simple references in form of a new syntax to place the same information in various places without writing redundant information.

PART-B

1. Explain about partial order planning with an example.
2. Explain about the different types of state space searches.
3. Explain about partial order planning algorithm.
4. Describe in detail about planning graphs.
5. Explain in detail about graph plan algorithm.
6. Explain in detail about conditional planning with an example.
7. Explain about re planning agent algorithm.
8. Explain the concepts of forward and backward state space search in detail. (Nov/Dec 2014)
9. Describe Graph plan algorithm in detail with example (Nov/Dec 2014)
10. Describe planning methods for handling indeterminacy. (Nov/Dec 2014)
11. Describe a planning method based on hierarchical task networks with an example (Nov/Dec 2014)
13. Explain planning graphs used in providing better heuristics estimates. (Nov/Dec 2012)
14. Explain the procedure of planning with state space search with example. (Nov/Dec 2013)
15. Explain the process of scheduling with resource constraints in detail. (Nov/Dec 2013)
16. Explain the concept of planning with state space search with examples(April/May 2015)
17. Explain the use of planning graph in providing better heuristic estimation (April/May 2015)
18. Explain the hidden markov model. (Nov/Dec 2013)
19. Explain the need of fuzzy set and fuzzy logic with example. (Nov/Dec 2013)
20. Explain the inference in temporal models. (April/May 2015)
23. (a) Briefly explain how reasoning is done using fuzzy logic (May/June 2016) (b) Explain Dempster–Shafer Theory (May/June 2016)
24. What is Forward Chaining and how does it work? Explain the forwarded Chaining algorithm with an example (May/June 2016)

UNIT-IV

PLANNING AND MACHINE LEARNING

PART-A

1. 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.

2. Define informational equivalence.

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

3. List the difference between 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

4. Define Explanation-Based Learning.
The background knowledge is sufficient to explain the hypothesis. 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

5. What is meant by Relevance-Based Learning?
• uses prior knowledge in the form of determinations to identify the relevant attributes
• generates a reduced hypothesis space

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

7. Differentiate between 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.

8. State the factors that play a role in the design of a learning system.
• Learning element
• Performance element
• Critic
• Problem generator

9. What is memorization?
The technique of memorization is used to speed up programs by saving the results of
computation. The basic idea is to accumulate a database of input/output pairs; when the
function is called, it first checks the database to see if it can avoid solving the problem from
scratch.

The agent learns an action-value function giving the expected utility of taking a given
action in a given state. This is called Q-Learning.

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.

12. Define Ockham’s razor.
Extracting a pattern means being able to describe a large number of cases in a concise
way. Rather than just trying to find a decision tree that agrees with example, try to find a
concise one, too.
13. Define Bayesian learning
Bayesian learning simply calculates the probability of each hypothesis, given the data, and makes predictions on that basis. That is, the predictions are made by using all the hypotheses, weighted by their probabilities, rather than by using just a single “best” hypothesis.

14. What is meant by hidden variables?
Many real-world problems have hidden variables (sometimes called latent variables) which are not observable in the data that are available for learning.

15. What is over fitting (Nov/Dec2014)?
Over fitting occurs when a statistical model describes random error or noise instead of the underlying relationship. Over fitting generally occurs when a model is excessively complex, such as having too many parameters relative to the number of observations. A model that has been over fit will generally have poor predictive performance, as it can exaggerate minor fluctuations in the data.

16. Distinguish between supervised learning and reinforcement learning (Nov/Dec 2012)
“Reinforcement learning (RL) and supervised learning are usually portrayed as distinct methods of learning from experience. RL methods are often applied to problems involving sequential dynamics and optimization of a scalar performance objective, with online exploration of the effects of actions. Supervised learning methods, on the other hand, are frequently used for problems involving static input-output mappings and minimization of a vector error signal, with no explicit dependence on how training examples are gathered.

17. 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.

18. What are all the various uses of a belief network?
- Making decisions based on probabilities in the network and on the agent’s utilities.
- Deciding which additional evidence variables should be observed in order to gain useful information.
- Performing sensitivity analysis to understand which aspects of the model have the greatest impact on the probabilities of the query variables (and therefore must be accurate).

19. List down applications of Bayesian network (April/May 2015)
- image processing
- gaming
- decision support systems
- information retrieval
- semantic search
20. What is rote learning? (MAY/JUNE 2016)
Rote learning is a memorization technique based on repetition. The idea is that one will be able to quickly recall the meaning of the material the more one repeats it. Some of the alternatives to rote learning include meaningful learning, associative learning, and active learning.

PART - B

1. Explain the learning decision tree with algorithm
2. Explain the explanation based learning?
3. Explain how learning with complete data is achieved?
4. Discuss learning with hidden variables?
5. Explain all the statistical learning method available in AI.
7. Explain decision tree learning algorithm (Nov/Dec 2014)
9. Explain the basic concept of support vector machine. (Nov/Dec 2014)
10. Give the complete agent design for an exploratory Q learning agent (Nov/Dec 2014)
11. Explain the concept of learning using Decision tree (Nov/Dec 2012)
12. Write short notes on (Nov/Dec 2012)
   i) Reinforcement learning
   ii) Explanation based learning
13. Explain the process of learning on action utility function (Nov/Dec 2013)
14. Explain the temporal difference learning with example (Nov/Dec 2013)
15. What are various approaches for instance based learning. Explain any one Eg. (Nov/Dec 2013)
16. The following table consists of training data from an employee database. The data have been generalized. Let status be the class label attribute. Construct Decision tree from the given data (April/May 2015)

<table>
<thead>
<tr>
<th>Department</th>
<th>Age</th>
<th>Salary</th>
<th>Count</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sales</td>
<td>31..35</td>
<td>46k..50k</td>
<td>30</td>
<td>Senior</td>
</tr>
<tr>
<td>Sales</td>
<td>26..30</td>
<td>26k..30k</td>
<td>40</td>
<td>Junior</td>
</tr>
<tr>
<td>Sales</td>
<td>31..35</td>
<td>31k..35k</td>
<td>40</td>
<td>Junior</td>
</tr>
<tr>
<td>Systems</td>
<td>31..35</td>
<td>31k..35k</td>
<td>42</td>
<td>Senior</td>
</tr>
<tr>
<td>Systems</td>
<td>42..45</td>
<td>36k..40k</td>
<td>40</td>
<td>Junior</td>
</tr>
<tr>
<td>Sales</td>
<td>31..35</td>
<td>31k..35k</td>
<td>40</td>
<td>Junior</td>
</tr>
</tbody>
</table>

17. Explain in detail about Active and Passive Reinforcement learning (April/May 2015)
18. Explain variable elimination algorithm for answering queries on Bayesian networks (Nov/Dec 2014)
19. Discuss forward-backward algorithm in details. (Nov/Dec 2014)
20. Discuss the different design issues to be solved to use HMM for real world applications. (Nov/Dec 2014)
22. Explain the concept of inference in temporal models. (Nov/Dec 2012)
23. How to handle uncertain knowledge with example . (Nov/Dec 2013)
24. How to represent knowledge in uncertain domain (Nov/Dec 2013)
25. (i) Describe the components of a planning system (MAY/JUNE 2016)
(ii) What is ID3? Write the drawback of ID3? (MAY/JUNE 2016)
26. (i) Describe the Hierarchical planning method with an example. (MAY/JUNE 2016)
   Describe the Learning with macro operators. (MAY/JUNE 2016)

UNIT V
EXPERT SYSTEMS
PART - A

1. What are Expert Systems?
   The expert systems are the computer applications developed to solve complex problems in a particular domain, at the level of extra-ordinary human intelligence and expertise.

2. What are the Characteristics of Expert Systems
   - High performance
   - Understandable
   - Reliable
   - Highly responsive

3. List out the Capabilities of Expert Systems
   The expert systems are capable of Instructing and assisting human in decision making
   - Demonstrating
   - Deriving a solution
   - Diagnosing
   - Explaining
   - Interpreting input
   - Predicting results
   - Justifying the conclusion
   - Suggesting alternative options to a problem

4. What are the Requirements of Efficient ES User Interface?
   - It should help users to accomplish their goals in shortest possible way.
   - It should be designed to work for user’s existing or desired work practices.
   - Its technology should be adaptable to user’s requirements; not the other way round.
   - It should make efficient use of user input.

5. Limitations of Expert Systems?
   No technology can offer easy and complete solution. Large systems are costly, require significant development time, and computer resources. ESs has their limitations which include
   - Limitations of the technology
   - Difficult knowledge acquisition
6. **List out the steps of Development of Expert Systems**
   The process of ES development is iterative. Steps in developing the ES include –
   - Identify Problem Domain
   - Design the System
   - Develop the Prototype
   - Test and Refine the Prototype
   - Develop and Complete the ES
   - Maintain the ES

7. **What are the Benefits of Expert Systems**
   - Availability – they are easily available due to mass production of software.
   - Less Production Cost – Production cost is reasonable. This makes them affordable.
   - Speed – they offer great speed. They reduce the amount of work an individual puts in.
   - Less Error Rate – Error rate is low as compared to human errors.
   - Reducing Risk – they can work in the environment dangerous to humans.
   - Steady response – they work steadily without getting motional, tensed or fatigued.

8. **List out the expert System Technology**
   - Expert System Development Environment
   - Tools
   - Shells

9. **Give an importance of Expert System Shells**
   The part of an expert system that does not contain any domain specific or case specific knowledge is the expert system shell. A single expert system shell can be used to build a number of different expert systems. An example of an expert system shell is CLIPS.

10. **Write notes on Knowledge Engineering of experts system?**
   i. Takes knowledge from experts and inputs it into the expert system.
   ii. Usually choose which expert system shell to use.
   iii. Responsible for entering meta-rules.

11. **What is CLIPS?**
    b. A CLIP uses a LISP-like notation to enter rules.

12. **Explain Backward Chaining in Expert Systems**
    Backward chaining is often used in expert systems that are designed for medical diagnosis:
• For each hypothesis, H:
  • If H is in the facts database, it is proved.
  • Otherwise, if H can be determined by asking a question, then enter the user’s answer in
    the facts database. Hence, it can be determined whether H is true or false, according to the
    user’s answer.

13. Write a Simple Medical Expert System?

Rules
• If headache then prescribe pain killer
• If headache and sore throat and coughing than diagnose flu
• If tired and headache then diagnose glandular fever
• If tired and sore throat then diagnose tonsillitis
If tired than diagnose stress.

14. List out all ES Components in AI

1. Knowledge Acquisition Subsystem
2. Knowledge Base
3. Inference Engine
4. User Interface
5. Blackboard (Workplace)
6. Explanation Subsystem (Justifier)
7. Knowledge Refining System
8. User
9. Most ES do not have a Knowledge Refinement Component

15. List out the two Basic Knowledge Base Elements

ii. Facts
iii. Special heuristics, or rules that direct the use of knowledge
iv. Knowledge is the primary raw material of ES
v. Incorporated knowledge representation


b. Development
c. Consultation
d. Improvement

17. What are the Problem Areas Addressed by Expert Systems

  o Interpretation systems
  o Prediction systems
  o Diagnostic systems
  o Design systems
  o Planning systems
  o Monitoring systems
  o Debugging systems
18. What is Meta knowledge? How Meta knowledge is represented in rule based expert systems? (May/June 2016)
In ES, Meta knowledge refers to knowledge about the operation of knowledge-based systems
- Meta knowledge is knowledge about knowledge and expertise.
- Most successful expert systems are restricted to as small a domain as possible.
- In an expert system, ontology is the Meta knowledge that describes everything known about the problem domain.
- Wisdom is the meta knowledge of determining the best goals of life and how to obtain them.

19. Write any four earliest expert systems? (May/June 2016)
- MYCIN
- DART
- XOON
- Expert systems shells.

20. Write MYCIN rule example:
1. IF the infection is meningitis
2. AND patient has evidence of serious skin or soft tissue infection
3. AND organisms were not seen on the stain of the culture
4. AND type of infection is bacterial
5. THEN There is evidence that the organism (other than those seen on cultures or smears) causin the infection is Staphylococcus coagpus.

PART-B

1. What is an expert system shell
2. What are common pitfalls in planning an expert system
3. What is knowledge acquisition? Explain in detail
4. Discuss briefly about meta knowledge
   i) Discuss briefly about the EMYCIN in detail
   ii) Illustrate Heuristics with an example.
   iii) Classify the XOON and DART in detail and write its application.
5. Draw the schematic diagram of an expert system. Explain all the relevant components.
6. Explain the various stages of expert system development?
7. (i) Explain about the knowledge acquisition (May/June 2016)
   (ii) Write the Characteristics features of Expert systems (May/June 2016)
8. (i) Explain the basic components of an expert systems (May/June 2016)
   (ii) Write any six applications of expert systems. (May/June 2016)