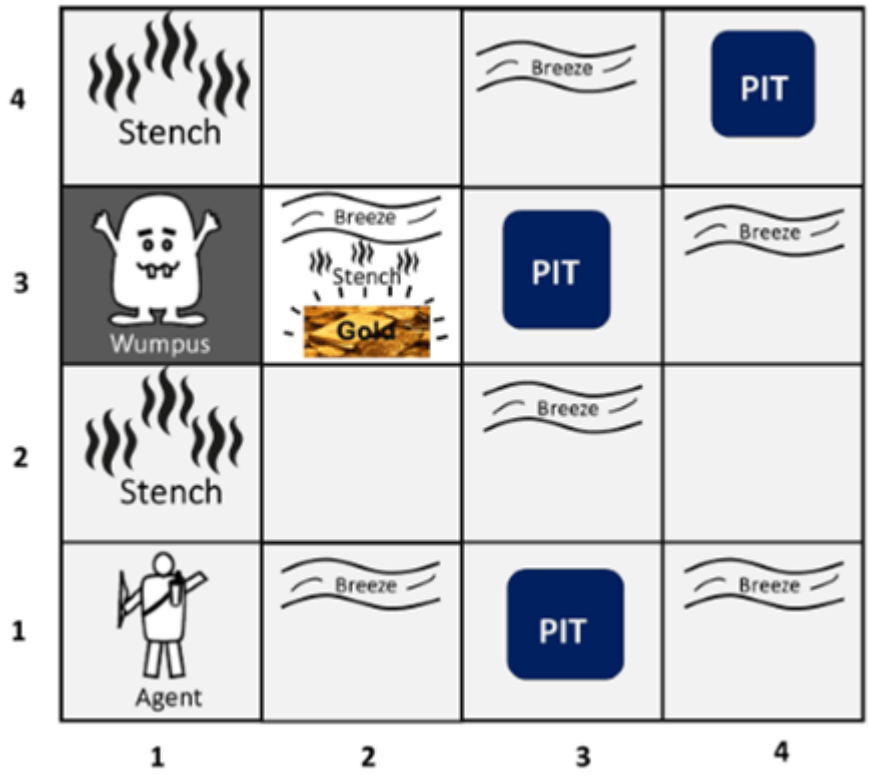
1. How would you solve the wumpus world environment problem using knowledge based agent.

Ans: The Wumpus world is a simple world example to illustrate the worth of a knowledge-based agent and to represent knowledge representation.

The Wumpus world is a cave which has 4/4 rooms connected with passageways. So there are total 16 rooms which are connected with each other. We have a knowledge-based agent who will go forward in this world.

Following is a sample diagram for representing the Wumpus world. It is showing some rooms with Pits, one room with Wumpus and one agent at (1, 1) square location of the world.



Sensors:

- The agent will perceive the **stench** if he is in the room adjacent to the Wumpus. (Not diagonally).
- The agent will perceive **breeze** if he is in the room directly adjacent to the Pit.

- The agent will perceive the **glitter** in the room where the gold is present.
- The agent will perceive the **bump** if he walks into a wall.
- When the Wumpus is shot, it emits a horrible **scream** which can be perceived anywhere in the cave.

Actuators:

- Left turn,
- Right turn
- Move forward
- Grab
- Release
- Shoot.

Environment :

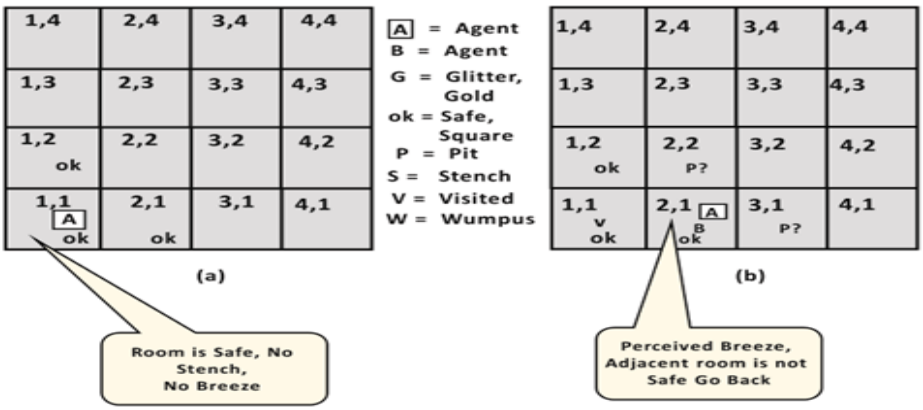
- A 4*4 grid of rooms.
- The agent initially in room square [1, 1], facing toward the right.
- Location of Wumpus and gold are chosen randomly except the first square [1,1].
- Each square of the cave can be a pit with probability 0.2 except the first square.

Now we will explore the Wumpus world and will determine how the agent will find its goal by applying logical reasoning.

Agent's First step:

Initially, the agent is in the first room or on the square [1,1], and we already know that this room is safe for the agent, so to represent on the below diagram (a) that room is safe we will add symbol OK. Symbol A is used to represent agent, symbol B for the breeze, G for Glitter or gold, V for the visited room, P for pits, W for Wumpus.

At Room [1,1] agent does not feel any breeze or any Stench which means the adjacent squares are also OK.

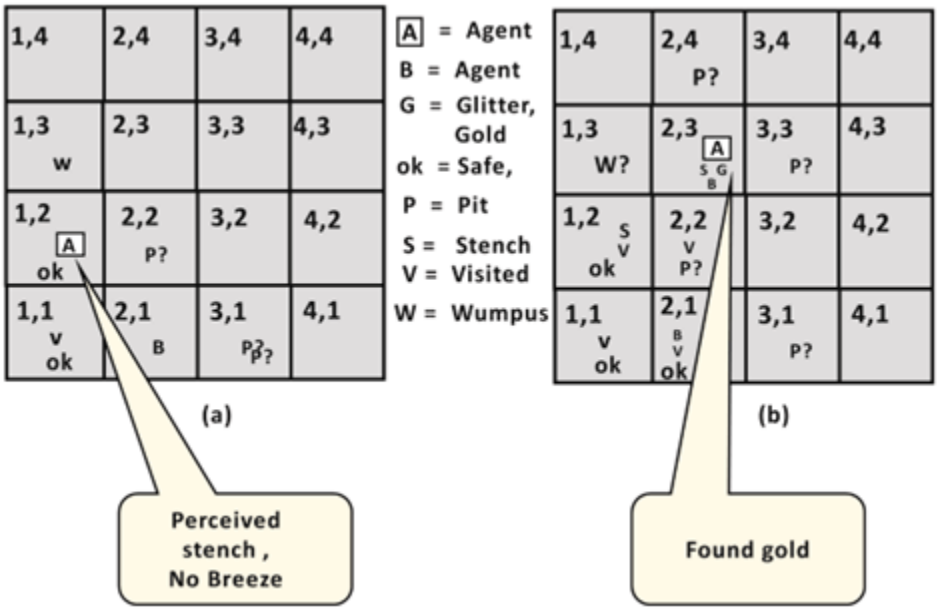


Agent's second Step:

Now agent needs to move forward, so it will either move to [1, 2], or [2,1]. Let's suppose agent moves to the room [2, 1], at this room agent perceives some breeze which means Pit is around this room. The pit can be in [3, 1], or [2,2], so we will add symbol P? to say that, is this Pit room?

Now agent will stop and think and will not make any harmful move. The agent will go back to the [1, 1] room. The room [1,1], and [2,1] are visited by the agent, so we will use symbol V to represent the visited squares.

Agent's third step:



At the third step, now agent will move to the room [1,2] which is OK. In the room [1,2] agent perceives a stench which means there must be a Wumpus nearby. But Wumpus cannot be in the room [1,1] as by rules of the game, and also not in [2,2] (Agent had not detected any stench when he was at [2,1]). Therefore agent infers that Wumpus is in the room [1,3], and in current state, there is no breeze which means in [2,2] there is no Pit and no Wumpus. So it is safe, and we will mark it OK, and the agent moves further in [2,2].

Agent's fourth step:

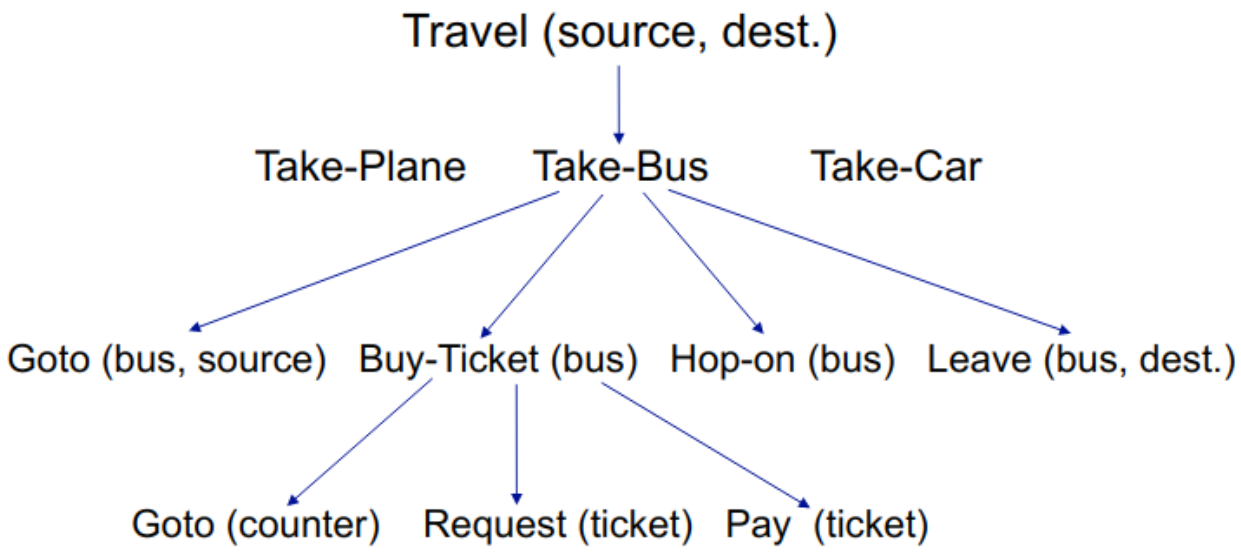
At room [2,2], here no stench and no breezes present so let's suppose agent decides to move to [2,3]. At room [2,3] agent perceives glitter, so it should grab the gold and climb out of the cave.

2. Explain Hierarchical Planning

Ans:

- Hierarchical planning is a planning method based on Hierarchical Task Network (HTN) or HTN planning.

- It combines ideas from Partial Order Planning & HTN Planning.
- HTN planning is often formulated with a single “top level” action called Act, where the aim is to find an implementation of Act that achieves the goal.
- In HTN planning, the initial plan is viewed as a very high level description of what is to be done.
- This plan is refined by applying decomposition actions.
- Each action decomposition reduces a higher level action to a partially ordered set of lower level actions.
- This decomposition continues until only the primitive actions remain in the plan.
- Consider the example of a hierarchical plan to travel from a certain source to destination.



- In the above hierarchical planner diagram, suppose we are Travelling from source “Mumbai” to Destination “Goa”.
- Then, you can plan how to travel: whether by Plane, Bus, or a Car. Suppose, you choose to travel by “Bus”.
- Then, “Take-Bus” plan can be further broken down into set of actions like: Goto Mumbai – Bus stop, Buy-Ticket for Bus, Hop-on Bus, & Leave for Goa.
- Now, the four actions in previous point can be individually broken down. Take, “By-Ticket for Bus”.
- It can be decomposed into: Goto Bus stop counter, Request Ticket & Pay for Ticket.
- Thus, each of these actions can be decomposed further, until we reach the level of actions that can be executed without deliberation to generate the required motor control sequences.

Advantages of Hierarchical Planning:

1. The key benefit of hierarchical structure is that, at each level of the hierarchy, plan is reduced to a small number of activities at the next lower level, so the **computational cost of finding the correct way** to arrange those activities for the current problem **is small**.
2. HTN methods **can create the very large plans** required by many real-world applications.
3. Hierarchical structure makes it **easy to fix problems** in case things go wrong.
4. For complex problems hierarchical planning is **much more efficient** than single level planning.

Disadvantages of Hierarchical Planning:

1. Many of the HTN planners **require a Deterministic environment**.
2. Many of the HTN planners are **unable to handle uncertain outcomes** of actions.

3. What actions would you take to prove "Some who are intelligent can't read"

   Whoever can read is literate.

   Dolphins are not literate.

   Some dolphins are intelligent.

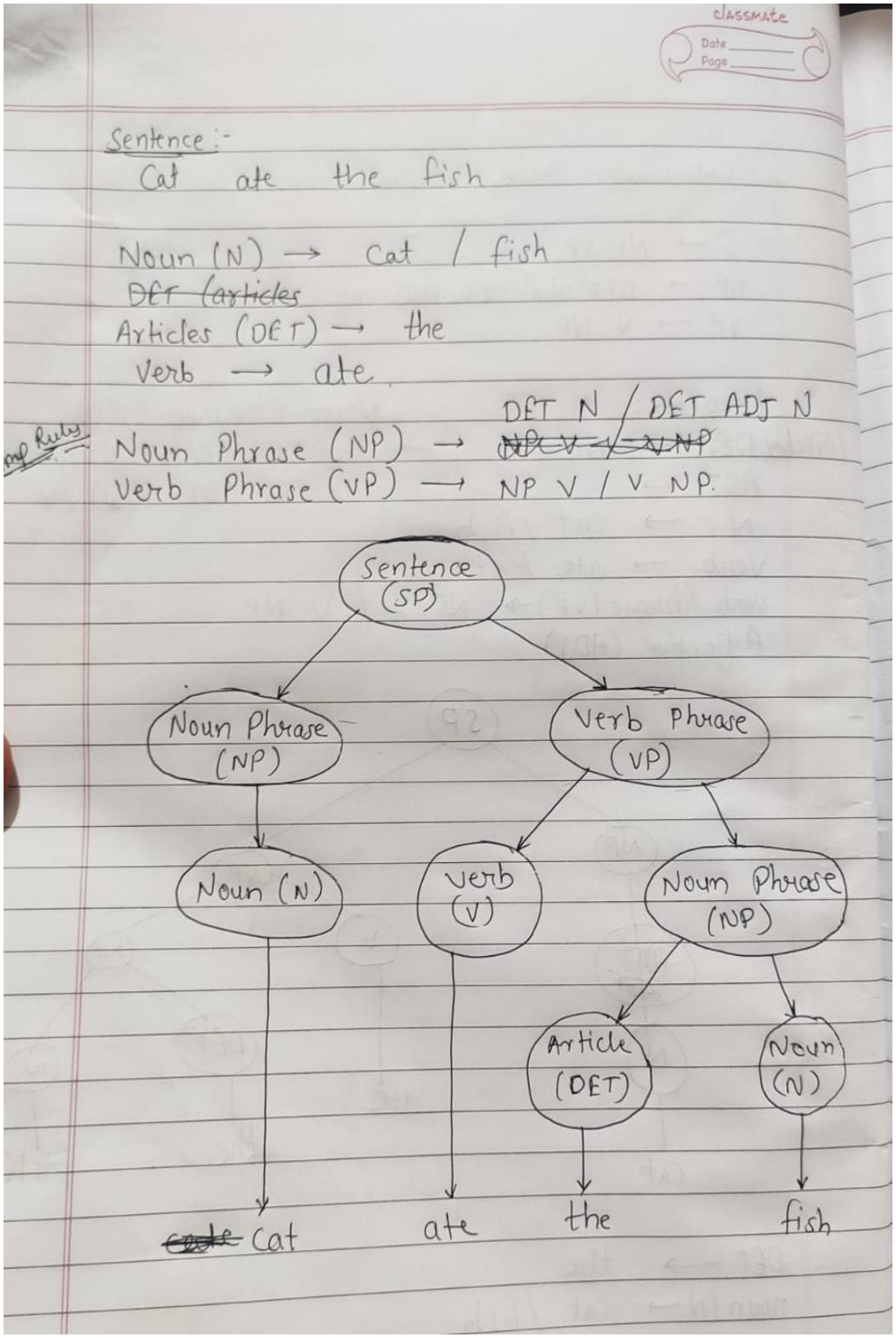
4. Compare the importance of Partial order planning over Total order planning.

Ans: Partial-order planning is the opposite of total-order planning, in which actions are sequenced all at once and for the entirety of the task at hand. The question arises when one has two competing processes, which one is better? The partial-order planning is superior to total-order planning, as it is faster and thus more efficient. The that partial-order planning performs better because it produces more trivial serializability than total-order planning. Trivial serializability facilitates a planner's ability to perform quickly when dealing with goals that contain subgoals. Planners perform more slowly when dealing with laboriously serializable or nonserializable subgoals. The determining factor that makes a subgoal trivially or laboriously serializable is the search space of different plans. They found that partial-order planning is more adept at finding the quickest path, and is therefore the more efficient of these two main types of planning.

5. What data is used to evaluate award and punishment of robot navigation?

6. Generate the parse tree for a sentence “Cat ate the fish”.

Ans:



7. How AI will help in the Robotics application.

Ans:

Robots can benefit from AI and machine learning in different ways, and these AI-enabled capabilities include:

- Computer vision. AI and computer vision technologies can help robots to identify and recognize objects they encounter, help pick out details in objects and help with navigation and avoidance.
- AI-enabled manipulation and grasping. Long considered a difficult task for robots, AI is being used to help robots with grasping items. With the help of AI, a robot can reach out and grasp an object without the need for a human controller.
- AI-enhanced navigation and motion control. Through enhanced machine learning capabilities, robots gain increased autonomy, reducing the need for humans to plan and manage navigation paths and process flows. Machine learning and AI help a robot analyze its surroundings and help guide its movement, which enables the robot to avoid obstacles, or in the case of software processes, automatically maneuver around process exceptions or flow bottlenecks.
- Real-world perception and natural language processing. For robots to have some level of autonomy, they often need to be able to understand the world around them. That understanding comes from AI-enabled recognition and natural language processing. Machine learning has shown significant ability to help machines understand data and identify patterns so that it can act as needed.

In the past, researchers have long thought about how to apply artificial intelligence to

robotics but ran into limitations of computational power, data constraints and funding. Many of those limitations are no longer in place, and as such, we now may be entering a golden age of robotics. With the help of machine learning, robots are becoming more responsive, more collaborative, and integrated into other systems.

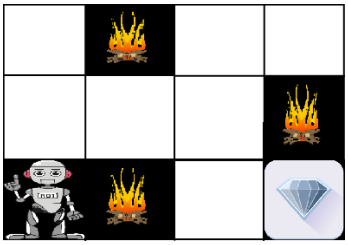
Likewise, many of the RPA vendors are adding intelligent process automation to their bots to help increase their usefulness. As such, they are looking at AI technologies such as NLP or computer vision to help make these bots more intelligent. Bots that leverage machine learning and adapt to new information and data can be considered intelligent tools that can significantly impact and increase the tasks performed rather than just bots.

8. Explain Reinforcement Learning

Ans:

Reinforcement learning is an area of Machine Learning. It is about taking suitable action to maximize reward in a particular situation. It is employed by various software and machines to find the best possible behavior or path it should take in a specific situation. Reinforcement learning differs from supervised learning in a way that in supervised learning the training data has the answer key with it so the model is trained with the correct answer itself whereas in reinforcement learning, there is no answer but the reinforcement agent decides what to do to perform the given task. In the absence of a training dataset, it is bound to learn from its experience.

Example: The problem is as follows: We have an agent and a reward, with many hurdles in between. The agent is supposed to find the best possible path to reach the reward. The following problem explains the problem more easily.



The above image shows the robot, diamond, and fire. The goal of the robot is to get the reward that is the diamond and avoid the hurdles that are fired. The robot learns by trying all the possible paths and then choosing the path which gives him the reward with the least hurdles. Each right step will give the robot a reward and each wrong step will subtract the reward of the robot. The total reward will be calculated when it reaches the final reward that is the diamond.

Main points in Reinforcement learning –

- Input: The input should be an initial state from which the model will start
- Output: There are many possible outputs as there are a variety of solutions to a particular problem
- Training: The training is based upon the input, The model will return a state and the user will decide to reward or punish the model based on its output.
- The model keeps continues to learn.
- The best solution is decided based on the maximum reward.

Types of Reinforcement: There are two types of Reinforcement:

1. Positive –
   Positive Reinforcement is defined as when an event, occurs due to a particular behavior, increases the strength and the frequency of the behavior. In other words, it has a positive effect on behavior.

2. Negative –
   Negative Reinforcement is defined as strengthening of behavior because a negative condition is stopped or avoided.

9. Explain Knowledge based agent

Ans:

- An intelligent agent needs **knowledge** about the real world for taking decisions and **reasoning** to act efficiently.
- Knowledge-based agents are those agents who have the capability of **maintaining an internal state of knowledge, reason over that knowledge,**

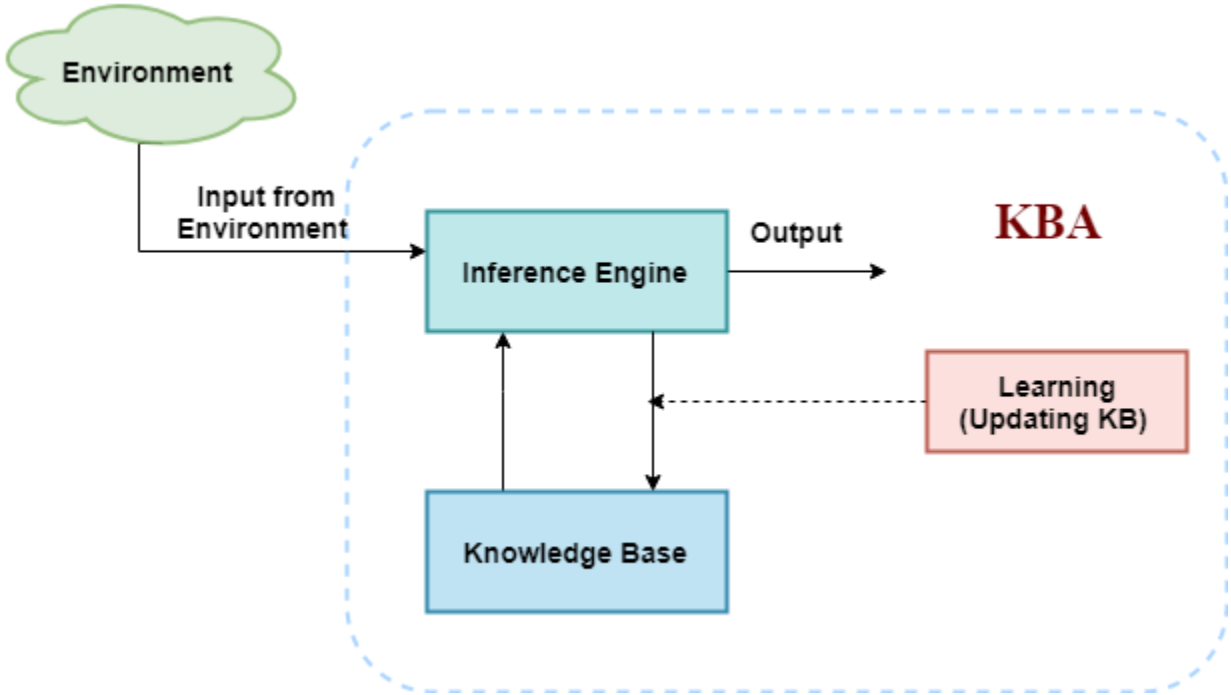
update their knowledge after observations and take actions. These agents can represent the world with some formal representation and act intelligently.

- Knowledge-based agents are composed of two main parts:
  - **Knowledge-base and**
  - **Inference system**.

A knowledge-based agent must able to do the following:

- An agent should be able to represent states, actions, etc.
- An agent Should be able to incorporate new percepts
- An agent can update the internal representation of the world
- An agent can deduce the internal representation of the world
- An agent can deduce appropriate actions.

The architecture of knowledge-based agent:



Various levels of knowledge-based agent:

A knowledge-based agent can be viewed at different levels which are given below:

1. Knowledge level

Knowledge level is the first level of knowledge-based agent, and in this level, we need to specify what the agent knows, and what the agent goals are. With these specifications, we can fix its behavior.

2. Logical level:

At this level, we understand that how the knowledge representation of knowledge is stored. At this level, sentences are encoded into different logics. At the logical level, an encoding of knowledge into logical sentences occurs.

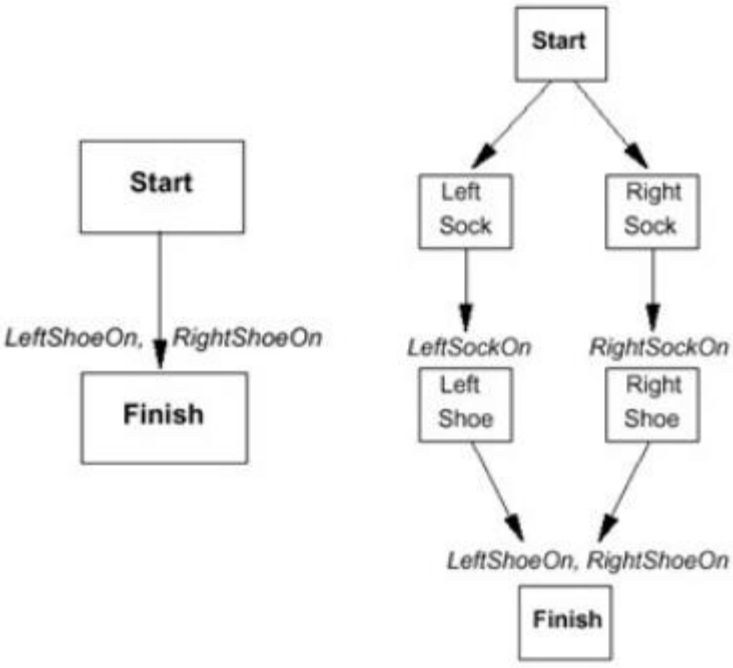
3. Implementation level:

This is the physical representation of logic and knowledge. At the implementation level agent perform actions as per logical and knowledge level. At this level, an automated taxi agent actually implement his knowledge and logic so that he can reach to the destination.

10. Explain Partial order planning

- Any algorithm that can place two actions into a plan without specifying which comes first is called as Partial Order Planning.
- Partial Order Planning works on several sub-goals independently & solves them with several sub-plans & thereafter combines them.
- With Partial Order Planning, problems can be decomposed so it can work well with non-cooperative environments. Thus, it is a plan which specifies all actions that need to be taken, but only specifies the order between actions when necessary.
- It uses a least commitment strategy.
- It consists of two states: “start” & “finish” where,

1. Start: No preconditions & Effects are initial states.
2. Finish: No effects & Preconditions are goals.



Partial Order Planning for Wearing a Shoe

Partial Order Planning as a search problem:

A partial-order plan consists of four components:

1. Set of actions:

- They form the steps of the plans.

2. Set of ordering preconditions:

- They are ordering constraints i.e. without performing action “x” we cannot perform “y”.

3. Set of casual links:

- It specifies which actions meet which preconditions of other actions.

4. Set of open preconditions:

- It specifies which preconditions are not fulfilled by any action in the partial order plan.

Advantages of Partial Order Planning:

1. Solves a huge state space problem in only a few steps.
2. Least commitment strategy means that search only occurs in places where sub-plans interact & delays the choice during the search.
3. Causal links used in this planner allows to recognize when to abandon a unworkable plan without wasting time exploring irrelevant parts.