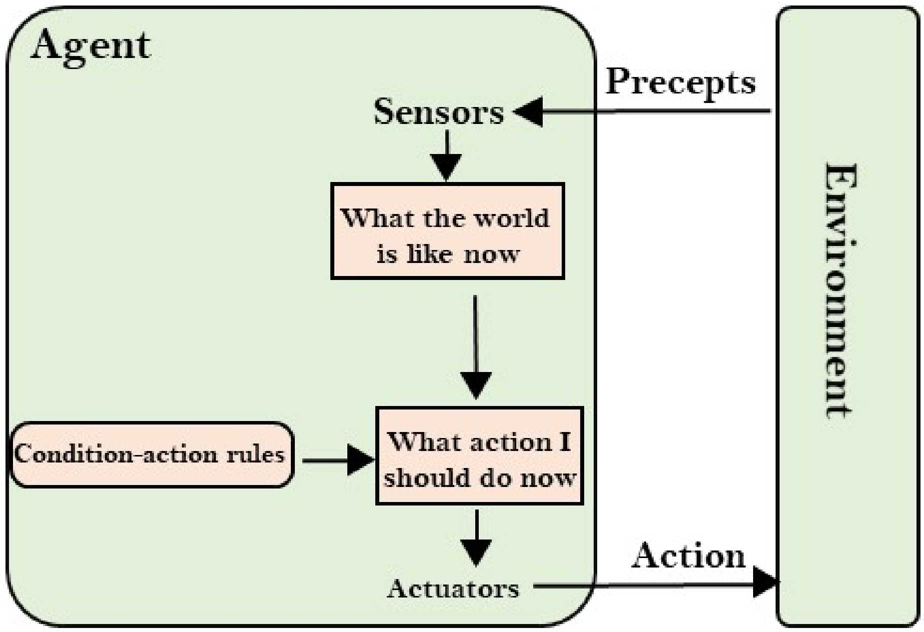
10. Explain about different types of Agents.

A. Agents can be grouped into five classes based on their degree of perceived intelligence and capability. All these agents can improve their performance and generate better action over the time. These are given below:

- Simple Reflex Agent
- Model-based reflex agent
- Goal-based agents
- Utility-based agent
- Learning agent
-

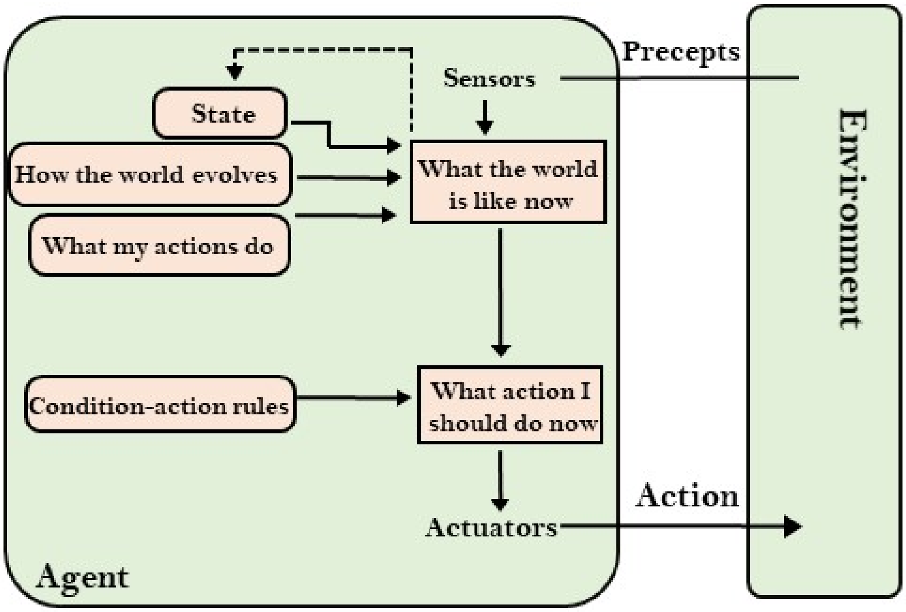
1. Simple Reflex agent:

- The Simple reflex agents are the simplest agents. These agents take decisions on the basis of the current percepts and ignore the rest of the percept history.
- The Simple reflex agent does not consider any part of percepts history during their decision and action process.
- Problems for the simple reflex agent design approach:
  - They have very limited intelligence
  - They do not have knowledge of non-perceptual parts of the
  - current state
  - Mostly too big to generate and to store.
  - Not adaptive to changes in the environment.



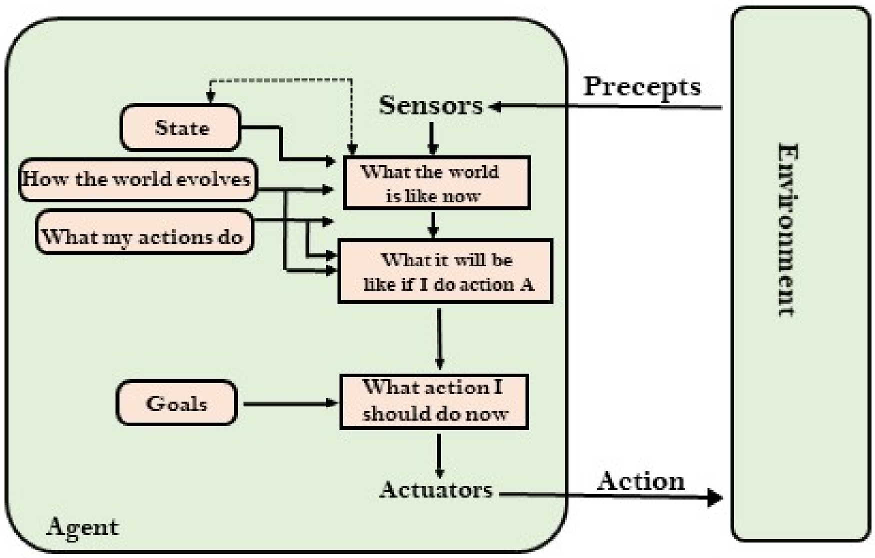
2. Model-based reflex agent

- The Model-based agent can work in a partially observable environment, and track the situation.
- A model-based agent has two important factors:
    - **Model:** It is knowledge about "how things happen in the world," so it is called a Model-based agent.
    - **Internal State:** It is a representation of the current state based on percept history.



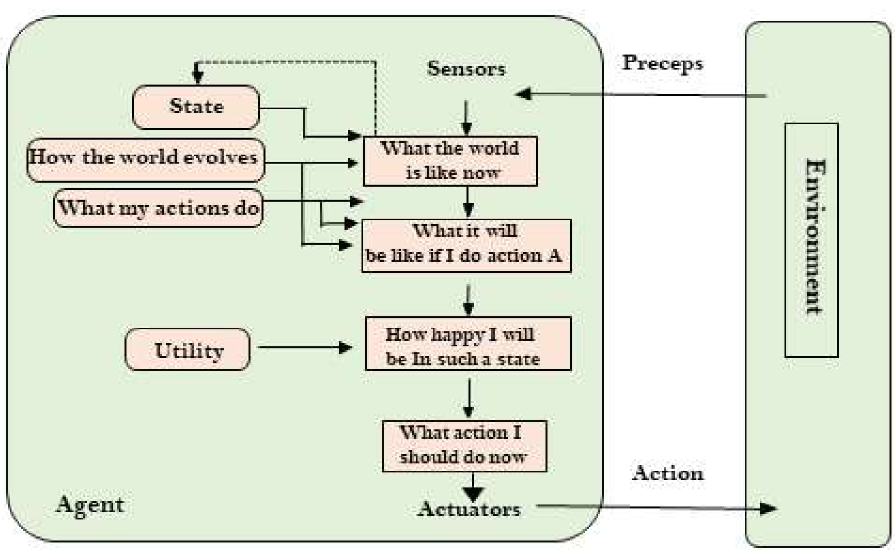
3. Goal-based agents

- The knowledge of the current state environment is not always sufficient to decide for an agent to what to do.
- The agent needs to know its goal which describes desirable situations.
- Goal-based agents expand the capabilities of the model-based agent by having the "goal" information.



4. Utility-based agents

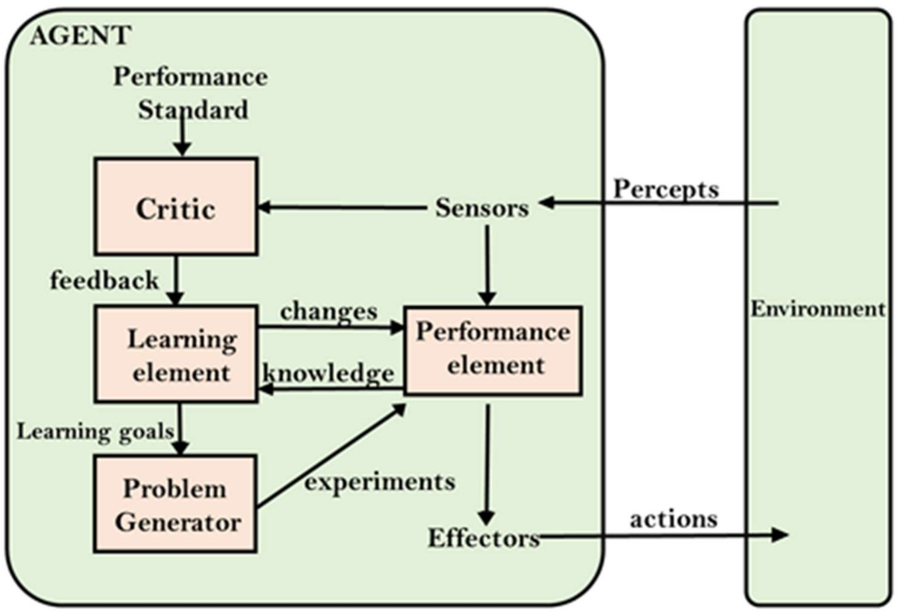
- These agents are similar to the goal-based agent but provide an extra component of utility measurement which makes them different by providing a measure of success at a given state.
- Utility-based agent act based not only goals but also the best way to achieve the goal.



-

5. Learning Agents

- A learning agent in AI is the type of agent which can learn from its past experiences, or it has learning capabilities.
- It starts to act with basic knowledge and then able to act and adapt automatically through learning.



11.Constrcut the DFS algorithm for any real time example.

A.Depth first Search or Depth first traversal is a recursive algorithm for searching all the vertices of a graph or tree data structure. Traversal means visiting all the nodes of a graph.

Depth First Search Algorithm

A standard DFS implementation puts each vertex of the graph into one of two categories:

1. Visited
2. Not Visited

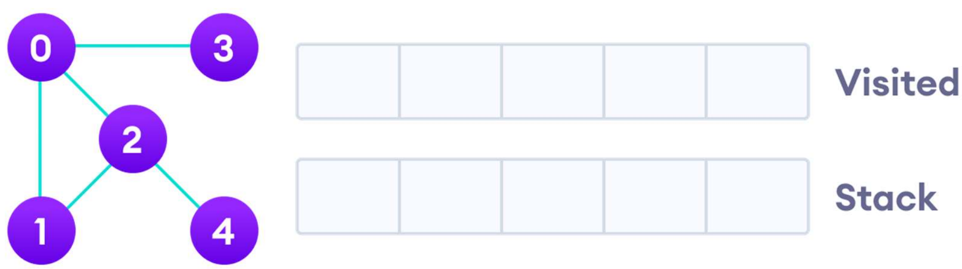
The purpose of the algorithm is to mark each vertex as visited while avoiding cycles.

The DFS algorithm works as follows:

1. Start by putting any one of the graph's vertices on top of a stack.
2. Take the top item of the stack and add it to the visited list.
3. Create a list of that vertex's adjacent nodes. Add the ones which aren't in the visited list to the top of the stack.
4. Keep repeating steps 2 and 3 until the stack is empty.

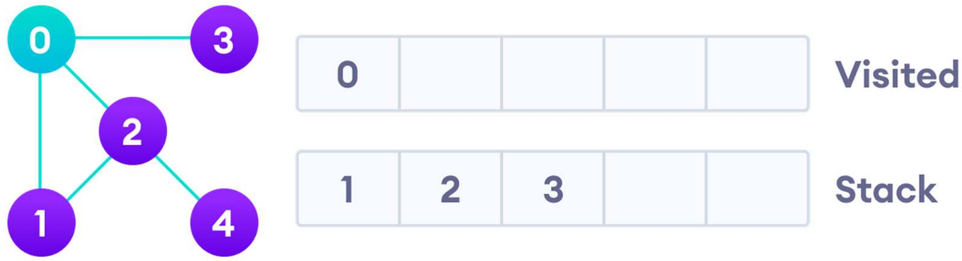
Depth First Search Example

Let's see how the Depth First Search algorithm works with an example. We use an undirected graph with 5 vertices.



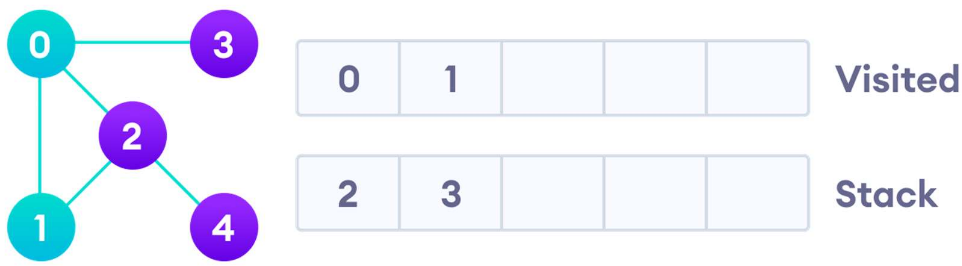
Undirected graph with 5 vertices

We start from vertex 0, the DFS algorithm starts by putting it in the Visited list and putting all its adjacent vertices in the stack.



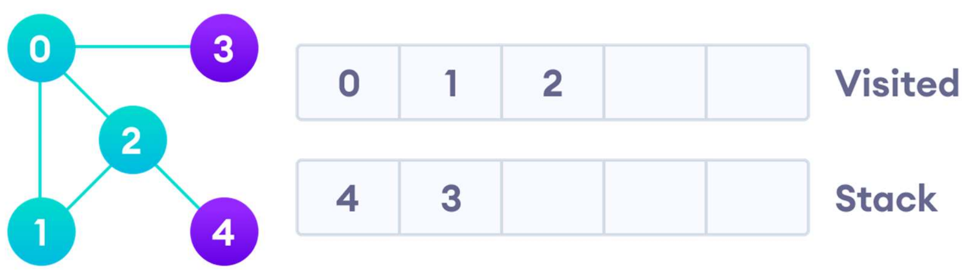
………………………..Visit the element and put it in the visited list

Next, we visit the element at the top of stack i.e. 1 and go to its adjacent nodes. Since 0 has already been visited, we visit 2 instead.

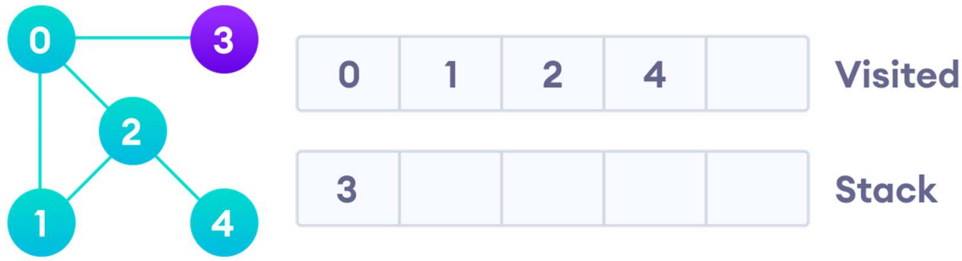


……………………...……..Visit the element at the top of stack

Vertex 2 has an unvisited adjacent vertex in 4, so we add that to the top of the stack and visit it.

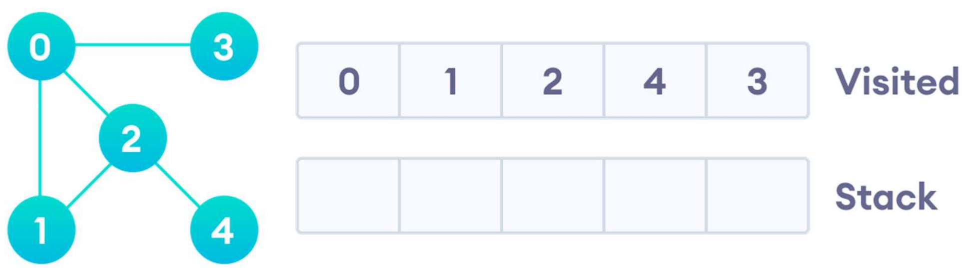


….Vertex 2 has an unvisited adjacent vertex in 4, so we add that to the top of ……………………………….the stack and visit it.



……Vertex 2 has an unvisited adjacent vertex in 4, so we add that to the top of …………………………..the stack and visit it.

After we visit the last element 3, it doesn't have any unvisited adjacent nodes, so we have completed the Depth First Traversal of the graph.



……After we visit the last element 3, it doesn't have any unvisited adjacent nodes, …………..so we have completed the Depth First Traversal of the graph.

Application of DFS Algorithm

1. For finding the path
2. To test if the graph is bipartite
3. For finding the strongly connected components of a graph
4. For detecting cycles in a graph.

12.Constrcut the Knowledge cycle and elaborate the different phases.

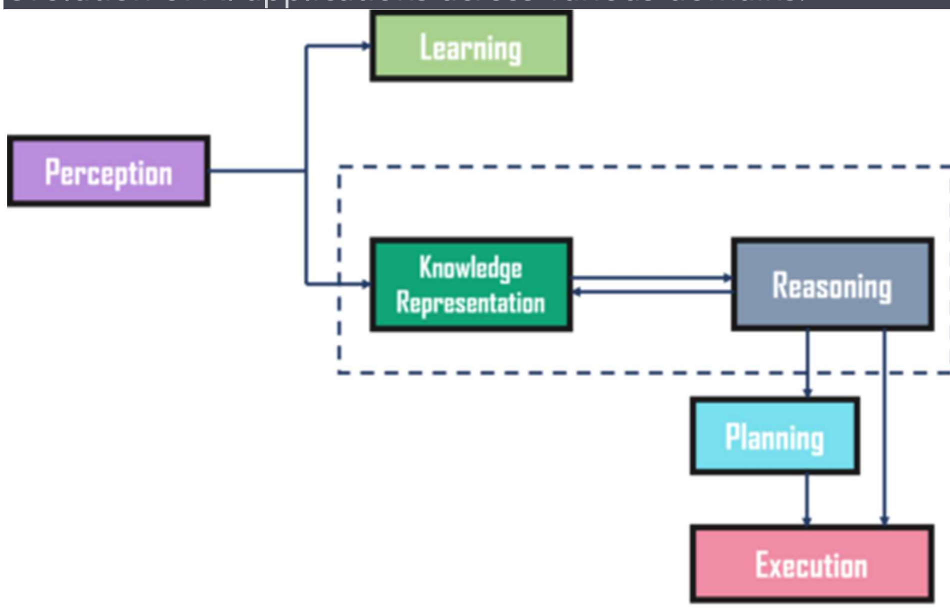
The knowledge cycle in the context of Artificial Intelligence (AI) involves the process of gathering, creating, refining, and applying knowledge to develop and improve AI systems. It encompasses various phases, each crucial for the advancement and effectiveness of AI applications. Here are the different phases of the knowledge cycle in AI:

1. **Data Collection and Acquisition**:
    - **Data Gathering**: This phase involves collecting data from various sources, such as sensors, databases, websites, or user interactions. Data can be structured or unstructured and may include text, images, audio, or video.
    - **Data Preprocessing**: Before feeding data into AI algorithms, it often requires cleaning, transformation, and normalization to ensure it is of high quality and consistency.
2. **Data Annotation and Labeling**:
    - **Data Labeling**: In supervised machine learning, data needs to be labeled to train AI models accurately. This phase may involve humans annotating data points with correct classifications or values.
3. **Feature Engineering**:
    - **Feature Selection and Engineering**: Engineers and data scientists work on selecting relevant features (attributes) from the data and creating new features to improve model performance. This step is crucial for extracting meaningful information from raw data.
4. **Model Development**:
    - **Algorithm Selection**: Choose the appropriate AI algorithm or model architecture based on the nature of the problem (e.g., deep learning for image recognition, decision trees for classification).
    - **Training**: Train the AI model using labeled data to learn patterns and make predictions or decisions. This phase may involve hyperparameter tuning to optimize model performance.
    - **Validation and Testing**: Validate the model's performance using a separate dataset and testing scenarios to ensure it generalizes well to new, unseen data.

5. **Knowledge Representation**:
   - **Model Interpretability**: In some cases, it's essential to understand how AI models make decisions. Techniques for model interpretability, such as feature importance analysis or attention mechanisms, help represent the model's knowledge in a human-understandable form.
6. **Deployment and Integration**:
   - **Integration with Applications**: Integrate AI models into real-world applications and systems, such as mobile apps, websites, or industrial automation.
   - **Scalability and Optimization**: Ensure that the AI system can handle varying workloads and is optimized for efficient operation.
7. **Monitoring and Maintenance**:
   - **Continuous Learning**: AI systems should adapt and learn from new data over time. Regularly retrain models and update data to keep the AI system up-to-date.
   - **Performance Monitoring**: Monitor the AI system's performance and address issues such as concept drift (when data distribution changes) or model degradation.
8. **Feedback Loop**:
   - **User Feedback**: Gather feedback from users and stakeholders to identify areas for improvement or refinement in the AI system.
   - **Knowledge Integration**: Incorporate user feedback and domain expertise back into the knowledge cycle to enhance the AI system's performance.
9. **Knowledge Sharing and Documentation**:
   - **Documentation**: Maintain comprehensive documentation about the AI system, including data sources, models, algorithms, and decision processes.
   - **Knowledge Sharing**: Share insights, best practices, and knowledge gained from AI development across teams and organizations to foster collaboration and learning.
10. **Ethical Considerations**:

- Throughout the knowledge cycle, ethical considerations related to bias, fairness, transparency, and privacy should be addressed to ensure responsible AI development and deployment.

The knowledge cycle in AI is iterative and ongoing, as AI systems continuously learn and adapt to new information and changing environments. It plays a vital role in the development, deployment, and evolution of AI applications across various domains.



9. Discuss the state of art and apply this to a real time example .

A. The term "state of the art" (SOTA) refers to the current highest level of development or advancement in a particular field or technology. In the context of technology and innovation, the state of the art represents the most advanced and cutting-edge achievements and capabilities in a given domain. To illustrate this concept, let's discuss the state of the art in the field of autonomous vehicles and provide a real-time example

Real-Time Example: Tesla's Full Self-Driving (FSD) Beta:

Scenario: Tesla, a prominent electric vehicle manufacturer, has been at the forefront of developing autonomous driving technology. They have been actively working on their Full Self-Driving (FSD) system.

Application: Tesla's FSD system is an example of the state of the art in autonomous vehicles.

Features of Tesla's FSD Beta (as of my last knowledge update in September 2021):

1. **Autosteer and Navigate on Autopilot**: Tesla's FSD system includes advanced driver-assistance features like Autosteer and Navigate on Autopilot, which enable the vehicle to control steering, acceleration, and braking on highways.
2. **Advanced Driver Monitoring**: FSD Beta includes driver monitoring features like cabin-facing cameras to ensure that the driver remains attentive and ready to take control if needed.
3. **Traffic-Aware Cruise Control**: Tesla's adaptive cruise control system can adjust the vehicle's speed based on traffic conditions and maintain a safe following distance.
4. **Autonomous Lane Changes**: The system can automatically change lanes when the driver initiates a lane change request by activating the turn signal.
5. **Full Self-Driving (FSD) Capability**: While not fully autonomous, Tesla's FSD Beta represents a significant advancement in autonomous driving technology, offering features like automated city driving, traffic signal recognition, and stop sign recognition.