2.5 Define in your own words the following terms: agent, agent function, agent program, rationality, autonomy, reflex agent, model-based agent, goal-based agent, utility-based agent, learning agent.

3.10 Define in your own words the following terms: state, state space, search tree, search node, goal, action, transition model, and branching factor.

3.11 What's the difference between a world state, a state description, and a search node? Why is this distinction useful?

2.5 AI Agent Definitions:

- **Agent:** An agent is anything that can sense its surroundings (environment) and take actions based on that information. Imagine a robot vacuum cleaner that uses sensors to detect dirt and then moves around to clean it.
- **Agent Function:** This defines how an agent chooses an action based on what it perceives. It's like a set of rules that tells the agent what to do in different situations.
- **Agent Program:** This is the actual computer code or instructions that make the agent function. It's the blueprint that tells the agent how to process information and act.
- **Rationality:** A rational agent makes decisions that are best for achieving its goals, considering all available information. It's like a chess player who chooses the move with the highest chance of winning.
- **Autonomy:** An autonomous agent acts on its own without needing constant human control. It's like a self-driving car that navigates the road without a driver behind the wheel.
- **Reflex Agent:** This is a simple agent that reacts directly to its current perception. Think of a thermostat that turns on the heater when it senses a cold temperature.
- **Model-Based Agent:** This agent has an internal model of its environment and uses it to decide on actions. It's like a self-driving car that uses a map and sensors to plan its route.
- **Goal-Based Agent:** This agent has specific goals it wants to achieve and takes actions that move it closer to those goals. Imagine a game AI that tries to win the game by making strategic moves.
- **Utility-Based Agent:** This agent assigns values (utilities) to different outcomes and chooses actions that maximize its overall benefit. Consider a shopping AI that picks items based on their price and your preferences.
- **Learning Agent:** This agent can improve its performance over time by learning from experience. It's like an AI playing a game that gets better at winning with each match.

3.10 Search Problem Definitions:

- **State:** A state describes the situation of the agent and its environment at a specific point in time. It's like a snapshot of the game board in chess or the current location of a robot on a map.
- **State Space:** This is the collection of all possible states the agent can be in. Imagine all the possible chessboard configurations or all the locations a robot can occupy in its environment.
- **Search Tree:** This is a tree-like structure that represents the possible sequences of actions the agent can take, with each node representing a state. It's like a decision tree showing all the possible paths the agent can follow.
- **Search Node:** A node in the search tree represents a specific state the agent can be in. It might also contain information about how the agent got to that state.
- **Goal:** This is the desired state the agent wants to reach. It's like the winning position in a game or the final destination for a robot.
- **Action:** An action is something the agent can do to change its state. It's like moving a piece in chess or instructing the robot to move forward.
- **Transition Model:** This describes how the agent's actions affect the environment and change its state. It tells the agent what happens when it takes a particular action.
- **Branching Factor:** This is the average number of possible actions the agent can take from any given state. Imagine how many squares a chess piece can move to on the board or how many directions a robot can go in.

3.11 World State vs. State Description vs. Search Node:

- **World State:** This is the actual physical state of the environment. It's the real chessboard with all the pieces in their positions, or the actual location of the robot in the world. We can't directly access this in AI problems.
- **State Description:** This is a representation of the world state that the agent can understand and use. It's like a digital version of the chessboard or a map showing the robot's location.
- **Search Node:** This is a node in the search tree that holds the agent's representation of the world state (state description) along with additional information relevant to search, like how it got there.