AI Assignment 2

Student IDs: anuhel2s, ljoy2s

Task 1: PEAS and Agent Program Types

For each subtask:

- Describe one iteration of the sense-plan-act cycle for the agent (3 points).
- State what types of agent programs could be running inside the agent and explain your reasoning (9 points).
- In the PEAS table, specify the task environment you chose for your agent (4 points).
- Describe the properties of your task environment and fill out the environment type table. Please justify your choices in the rightmost column of the environment table. (3+6 points)

Please solve 4 out of the 5 subtasks! (Total points: 100)

Task 1.1: Tesla Model 3 with Autopilot

Sense-plan-act cycle: Sense objects, lane markings, nearby vehicles, or obstacles → Plan the next driving action (e.g., lane change, speed adjustment, braking) or safety response → Act by steering, accelerating, or braking.

Agent program type(s): Model-based reflex agent, goal-based agent, and utility-based agent.

Model-based Reflex Agent: Tesla Autopilot creates and updates an internal model of the driving environment. It uses this model to make decisions based on current sensor data and the anticipated state of the road.

Goal-based Agent: The system operates with specific goals, such as staying in the correct lane, avoiding collisions, and following traffic rules to reach a destination safely.

Utility-based Agent: Tesla Autopilot may also optimize decisions based on utility functions, like maximizing safety and comfort by choosing the best speed or smoothest path.

PEAS Table	
Performance measure	Safely navigate roads, maintain lane position, follow traffic laws, avoid collisions, and respond to road conditions or hazards.

Environment	Highways, city streets, traffic, pedestrians, obstacles (e.g., other cars, road signs, barriers), and various weather conditions.
Actuators	Steering, braking, and acceleration mechanisms, along with indicators for signaling turns or lane changes.
Sensors	Cameras for object recognition, radar for detecting distances, ultrasonic sensors for close-range detection, GPS for location tracking.

Environment Table		
Fully observable?	No	The vehicle's sensors only provide partial information about the environment, such as visual data from cameras or distance data from radar.
Deterministic?	No	The environment is stochastic, as the behavior of other drivers, pedestrians, or changing road conditions can affect the outcome unpredictably.
Episodic?	No	The task is not episodic because actions taken (like turning or slowing down) are connected, and previous actions influence future decisions.
Static?	No	The environment is dynamic, with constantly

		changing factors such as traffic flow, road signs, and obstacles appearing suddenly.
Discrete?	No	The environment is continuous, with the car moving fluidly rather than in fixed steps. It adjusts speed and direction based on real-time data.
Single-agent?	No	The environment involves multiple agents, including other vehicles and pedestrians, which influence the car's behavior.

Task 1.2: AlphaGo

Sense-plan-act cycle: Sense the current state of the Go board, including stone positions and possible future moves → Plan the next move by evaluating potential actions using a neural network and search trees → Act by placing a stone in the optimal position on the board.

Agent program type(s): Model-based reflex agent, goal-based agent, and utility-based agent.

Model-based reflex agent: AlphaGo uses a model of the Go board, predicting the outcomes of potential moves.

Goal-based agent: AlphaGo operates with the goal of maximizing its chances of winning by making strategic moves.

Utility-based agent: It optimizes actions based on a value function that estimates the probability of winning from any given game state.

PEAS Table

Performance measure	Successfully win games of Go by choosing optimal moves and learning from each game played, maximizing the probability of winning.
Environment	Go board (19x19 grid), opponents (human or AI), and all possible game states that arise from moves made by both players.
Actuators	The system's ability to place a stone at a specific location on the Go board (virtual action through software).
Sensors	Neural network analysis of the Go board state, probability estimations of the next moves, and feedback from the game outcome (win or loss).

Environment Table		
Fully observable?	Yes	The entire Go board is fully visible at all times, so the agent has complete information about the current state of the game.
Deterministic?	Yes	The game of Go is deterministic; given the same sequence of moves, the outcome will always be the same.
Episodic?	No	Go is non-episodic because each move influences future moves, and decisions must take the entire game sequence into account.

Static?	Yes	The game state only changes when a player makes a move; otherwise, it remains static.
Discrete?	Yes	The game of Go has a finite number of discrete moves (placing a stone on one of 361 positions on the board).
Single-agent?	No	The environment involves two agents: AlphaGo and its opponent (human or AI), each influencing the game state.

Task 1.3: Google maps route planner

Sense-plan-act cycle:________________________

Agent program type(s):_______________________

PEAS Table	
Performance measure	
Environment	
Actuators	
Sensors	

Environment Table		
Fully observable?		
Deterministic?		
Episodic?		
Static?		
Discrete?		
Single-agent?		

Task 1.4: Samsung Bot Chef
https://www.youtube.com/watch?v=4mZztDca4Qo

Sense-plan-act cycle: Sense objects, voice commands, or obstacles → Plan the next cooking step or safety response → Act by manipulating kitchen tools or ingredients.

Agent program type(s): Model-based reflex agent and goal-based agent.

Model-based Reflex Agent: The Bot Chef builds a model of its environment. It has to remember where objects are (e.g., utensils, food items) and continuously updates its knowledge of the kitchen as it interacts. This makes it a model-based agent—it doesn't just react to immediate stimuli but uses its model of the environment to guide its actions.

Goal-based Agent: The Bot Chef works towards a specific goal: successfully completing a recipe. It doesn't just perform actions independently; it chooses actions that bring it closer to the goal of completing the meal preparation. This makes it a goal-based agent because it has to consider the steps needed to accomplish a long-term objective.

PEAS Table

Performance measure	Successfully assist users in cooking by executing voice commands, preparing ingredients, following recipes, and ensuring user safety.
Environment	Domestic kitchen with appliances, tools, and food ingredients. It must adapt to different kitchen layouts, objects, and user interactions.
Actuators	Robotic arms for handling objects like kitchen tools and ingredients, grippers to hold and manipulate items, motors for movement and control.
Sensors	Cameras for object recognition and computer vision, microphones for voice command detection, proximity sensors for safe interaction.

Environment Table		
Fully observable?	Partially	The Bot Chef's sensors (cameras, proximity sensors) provide information about its surroundings, but it may not perceive everything (e.g., items hidden from view).
Deterministic?	No	The environment is stochastic as there are variables the Bot Chef cannot control, like ingredient placement or unexpected user interactions.

Episodic?	Yes	The tasks are performed in episodes (e.g., individual recipe steps), and each step is relatively independent of previous or future actions.
Static?	No	The environment can change dynamically as ingredients may move, users may interact, or objects may be placed differently during a task.
Discrete?	No	The environment is continuous; Bot Chef operates with fluid actions such as grasping objects and moving around the kitchen.
Single-agent?	Yes	The Bot Chef operates as a single agent, handling tasks independently, though it interacts with humans who are not other autonomous agents.

Task 1.5: Grey Walter's Tortoise

https://www.youtube.com/watch?v=lLULRlmXkKo&t=64s

Sense-plan-act cycle: Sense light or obstacles → Decide whether to move toward the light or avoid obstacles → Act by moving accordingly.

Agent program type(s): Simple reflex agents.

The tortoises respond directly to their current inputs without any memory of past actions or reasoning about future steps. The sensors trigger immediate actions without the need for internal modeling or goal planning.

PEAS Table	
Performance measure	Navigate towards light, avoid obstacles, and (for Elsie) return to the charging station when low on power.
Environment	Rooms or open spaces with obstacles and light sources.
Actuators	Motors for movement and steering, along with mechanisms to avoid obstacles.
Sensors	Photoelectric cells for light detection, touch sensors to detect obstacles. (Elsie also has battery level sensors to detect when to recharge.)

Environment Table		
Fully observable?	Partially	The tortoises rely on light and touch sensors, but their vision is limited to their immediate surroundings.
Deterministic?	Yes	Both tortoises respond predictably to stimuli (light or obstacles). However, only Elsie can respond to battery levels and recharge automatically.
Episodic?	Yes	Both tortoises respond predictably to stimuli (light or obstacles). However, only

		Elsie can respond to battery levels and recharge automatically.
Static?	No	The environment may change (light sources turning on/off or obstacles being moved).
Discrete?	No	The tortoises' movement is continuous, adjusting in real-time to light and touch stimuli.
Single-agent?	Yes	Each tortoise (Elsie and Elmer) operates as a single agent, responding independently to its environment.