Part 3 - Agent for a Known Environment and Multiple Delivery or Collection Locations

Scenario Overview

- **Environment:** Same as Part 2
- **Scenario:** Pickup *or* Drop (must remain consistent)
- **Agents:** 3 autonomous agents
- **Parcels:** 10 total parcels (each agent must carry at least one)
  Agent1: 4 parcels
  Agent2: 3 parcels
  Agent3: 3 parcels

Agent Behavior

- Each agent must move from Start → Goal, hold for 3 seconds (pickup or deliver your parcel) then return to Start and stop.
- Each agent must have a different starting point.
- Every agent must have **multiple and different goal locations**.
- The speed of the agent increases or decreases by 10% per parcel, with a maximum limit of 90% increase or decrease.
  Pickup Scenario: Currentspeed * 0.9 ^ Parcelcount
  Drop Scenario: Currentspeed * 1.1 ^ Parcelcount
- The parcel that an agent will pick up or drop must be assigned via the Inspector.
- **When an agent stops, its speed must be exactly 0.**

Script Behavior Requirements

Waypoint Setup

- All waypoints must be interconnected
- Every waypoint must be marked as a **Goal type**

Pathfinding Logic

- Agents start from **different locations**
- Use **Ant Colony Optimization (ACO)** to:
    a. Navigate the environment
    b. Reach assigned goal locations
    c. Collect or deliver parcels
- The following **ACO parameters must be configurable via the Unity Inspector**:
  Alpha (α) – importance of pheromone trail
  Beta (β) – importance of heuristic distance
  Q value – pheromone deposit constant
  Default Pheromone – initial pheromone level on all connections
  Evaporation Factor (p) – pheromone evaporation rate
- After an agent has completed **all assigned parcel pickups or deliveries**:
  The agent must automatically switch from ACO to A* (Pathfinding Tester).
  During this switch:
  - The agent's speed must be set to exactly 0 in the ACO Tester.
  - The ACO Tester component must be disabled (unticked).
  - The A* Pathfinding Tester component must be enabled (ticked).

  The agent must then use A* to:
  - Navigate back to its original starting position.
  - Stop completely once the start position is reached.

Collision Avoidance

- If a slower agent detects another agent, it must:
  Slow down,
  Move slightly aside,
  Allow the other agent to pass before resuming movement.

Information Display (Canvas)

- Time (optional)
- Total distance traveled (optional)
- Number of parcels
- Current speed
- Status (e.g., Moving, Waiting, Stopped)
- Collision Avoidance Information
- Alpha Value
- Beta Value
- Q Value

You can refer to below image.

What to submit ?

● Unity project zip. (Format: UNIID_Portofolio1_Part2_Project.zip)

● Video (max 180s) (Format: UNID_Portofolio1_Part2Video.mp4)

(Simply narrate the simulation and the upgraded task.)