OOP Project: HiveMind - Autonomous Logistics

1. CONTEXT AND OBJECTIVES

It is the year 2030. Urban logistics is managed by autonomous fleets.
You play the role of Software Architect for the 'Hive Mind' system, a simulation and decision engine that coordinates a heterogeneous fleet of robots (Drones, Ground Robots, Scooters).

Main Objective:
Create a modular C++ application that:
1. Procedurally generates city maps (environment simulations).
2. Simulates physics and the lifecycle of agents.
3. Optimizes profit through a custom task allocation algorithm (Hive Mind).

2. WORLD DESCRIPTION

2.1. The Map (The Grid)
The city is represented as a matrix of size MxN. Each cell has a specific role:
- [.] Road (Space): Accessible for movement.
- [#] Wall: Obstacle. Cannot be traversed by ground units.
- [B] Base (Hub): Central storage, spawn location, and charging station.
- [S] Charging Station: Extra charging locations.
- [D] Client (Destination): Delivery locations.

2.2. 'Genesis' Module (Map Generation)
Requirements:
- Use Strategy Design Pattern via IMapGenerator interface.
- Implement: FileMapLoader (from file) and Procedural MapGenerator (random).
- Validation: Use Flood Fill/BFS to ensure paths exist from Base to all Clients.

3. FLEET AND PHYSICS

3.1. Agent Specifications
You must implement a polymorphic hierarchy derived from Agent.

Type	Sym	Medium	Speed	Battery	Cost/Tick	Capacity
Drone	A	Air	3	100	15	1
Robot	R	Ground	1	300	1	4
Scooter	S	Ground	2	200	4	2

3.2. Lifecycle (State Machine)
States: IDLE, MOVING, CHARGING (+25%/tick), DEAD (Battery=0 outside station).

4. BUSINESS LOGIC

Clients (D) are static. Packages are dynamic.
Packages are generated at Base (B) periodically. Each has a random Reward (200-800) and Deadline (10-20 ticks).

5. HIVEMIND ALGORITHM

Objective: Maximize Net Profit.
Profit = (Rewards) - (Operating Costs) - (Penalties)

Penalties:
- Dead Agent: -500
- Undelivered Package: -200
- Late Package: -50

Strategy: Dispatching (assigning tasks) and Routing (pathfinding decisions).

6. CONFIGURATION (simulation_setup.txt)

```
MAP_SIZE: 20 20
MAX_TICKS: 1000
MAX_STATIONS: 3
CLIENTS_COUNT: 10
DRONES: 3
ROBOTS: 2
SCOOTERS: 1
TOTAL_PACKAGES: 50
SPAWN_FREQUENCY: 10
```

7. DELIVERABLES & EVALUATION

1. C++ Source Code
2. Simulation Report (txt)
3. Documentation (PDF with UML)
4. Demo Video (3 min)

Grading:
- 25% Architecture (OOP, Patterns)
- 25% Functionality (Map gen, Physics)
- 10% Performance (Positive profit)
- 10% Style
- 30% Colloquium (Theory)

Deadline: January 8, 2025.