

Cellular Automata Based Ecosystem Simulation: Object-Oriented Programming Project Report

By: Huzaifa Iqbal(31563) & Ali Hamza(30618)

1. Overview

This project is an ecosystem simulator built using C++ and object-oriented programming (OOP) principles. It models a dynamic world where plants, herbivores, and carnivores interact, reproduce, and compete for survival over time.

Key Features

- Dynamic Grid System: A 20x20 grid where entities live and interact.
- Seasonal Effects: Weather impacts entity behavior (e.g., plants grow faster in summer, die in winter).
- Predator-Prey Mechanics: Carnivores hunt herbivores, herbivores eat plants.
- Reproduction & Lifecycle: Animals mate, reproduce, and die of old age or starvation.
- Migration: Animals move in/out of the ecosystem based on seasons.
- Statistics Tracking: Monthly reports on births, deaths, and population changes.

2. Setup Instructions

Prerequisites

- Compiler: Supports C++17 (e.g., g++, clang++, MSVC).
- Operating System: Windows, Linux, or macOS.

Compilation & Execution

1. Clone the repository
2. Compile the code:
`g++ -std=c++17 ecosystem.cpp -o ecosystem`
3. Run the simulation:
`./ecosystem`
(On Windows, use ecosystem.exe.)

3. Usage Guidelines

Initial Configuration

When the program starts, you will be prompted to:

1. Choose a hemisphere (N for Northern, S for Southern), affects seasonal cycles.
2. Set simulation duration (number of years).
3. Define initial populations:
 - Number of plants. (ideally between 230-280)
 - Number of herbivores. (ideally between 100-130)
 - Number of carnivores. (ideally between 5-10)

Simulation Controls

- The simulation runs month-by-month.
- After each month, the grid and statistics are displayed.
- Press Enter to proceed to the next month.
- Press Q to quit early.

Output Explanation

- Grid Symbols:
 - `P` = Plant
 - `H` = Male Herbivore
 - `h` = Female Herbivore
 - `C` = Male Carnivore
 - `c` = Female Carnivore
 - `*` = Empty cell
- Monthly Report: Shows births, deaths, migrations, and population counts.

4. Code Structure

FILE/CLASS	PURPOSE
Entity	Base class for all entities (plants, animals).
Plant	Handles plant growth, spreading, and seasonal death.
Animal	Base class for herbivores and carnivores (movement, eating, reproduction).
Herbivore	Eats plants, flees from carnivores, reproduces.
Carnivore	Hunts herbivores, reproduces less frequently.
Grid	Manages entity placement and grid operations.
MonthlyStats	Tracks monthly changes (deaths, births, etc.).

5. Example Simulation Workflow

1. Start: 10 plants, 5 herbivores, 2 carnivores.
2. Spring: Plants spread rapidly; some animals migrate in.
3. Summer: Carnivores hunt more.
4. Autumn: Some animals migrate out.
5. Winter: Plants die; weak animals starve.

6. End: Final population statistics displayed.

Happy Simulating! 🌿🦌🐺