CS 3005: Programming in C++

Predator Prey Simulation (Fight and Flight)

A variety of critters wander in the grasslands. Some are looking for grass to eat. We will call these the prey. Others are looking for prey to eat. We will call these the predators.

Assignment

In this step of the assignment, you will create an application that can generate 5 predators and 25 prey, and place them in a 20x20 grid. We call this collection the critters.

The critters must be created in random order (e.g. not all predators and then all prey). And this order must be preserved. No two critters may be created in the same location.

When a prey is asked to move, it must randomly choose from the 4 cardinal directions. If the direction of movement would not have it walk off the grid, and the new location isn’t already occupied, then it will move.

When a predator is asked to move, it must look for neighboring locations with prey. If one or more exist, the predator will choose one and move there, consuming the prey. If there are no neighboring prey, the predator must randomly choose from the 4 cardinal directions. If the direction of movement would not have it walk off the grid, and the new location isn’t already occupied, then it will move.

Every three time steps that a prey is asked to move, it will generate a new prey next to it, if there is an open space. New critters are to be added to the end of the critters order.

Every eight time steps that a predator is asked to move, it will generate a new predator next to it, if there is an open space. New critters are to be added to the end of the critters order.

If a predator is unable to eat a prey for three consecutive time steps, it will starve to death, and must be removed.

This will be a visual simulation. Your application must display all of the critters on a grid. The size of the display is your choice. But remember the grid is 20x20.

The simulation will allow one critter to move, reproduce and starve at a time. The critters must be allowed their turn in a cyclical order. Each movement must be initiated by the user pressing the \( s \) key on the keyboard.

Requirements

- Program must use OpenGL/Glut to display the grid and critters.
- Program must use OpenGL/Glut to receive the \( s \) key presses to advance the simulation.
- Program must have a 20x20 grid.
- Program must start with 5 predators.
- Program must start with 25 prey.
- Program must create and store the critters in an initially random order.
- Critters must move one at a time.
- Critters must move as described above.
- Critters must not leave the grid.
- Predators must try to consume prey.
- Predators and prey must reproduce at the rates specified above.
- New predators and prey must be added at the end of the container that stores all critters.
- Predators must starve at the rate specified above.
- Predators and prey must be drawn differently.
- The display must be kept up to date.
- The executable program must be called PP.

Hints

- The random order of predator/prey in the list of objects is strictly required. With polymorphism, your code will look clean. Without polymorphism, your code will look disorganized, and be difficult to get working correctly.
Show Off Your Work

To receive credit for this assignment, you must upload the source code (.h and .cpp files) and the Makefile to the Canvas submission system.

Additionally, the program must build and run. Any incorrect performance or memory errors will be counted against the assignment score.