CoyoteRoadrunnerSimulation

AUTHOR
Version
Mon May 6 2019

Table of Contents

Table of contents

Hierarchical Index

Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Class Index

Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Class Documentation

Agent Class Reference

`#include <Agent.h>`
Inherited by **Coyote**, and **Roadrunner**.

Public Member Functions

Agent (**SimulationBoard *board**, **Coordinates location**, unsigned int **breedCountdown**, BoardOccupantTypes type)
virtual **Agent * act** ()=0
bool **isAlive** ()

Protected Member Functions

Coordinates findRandomViableLocation (**Coordinates** currentPosition, BoardOccupantTypes criteria)
virtual **Agent * breed** ()=0
virtual void **die** ()
virtual void **move** ()=0

Protected Attributes

bool **alive**
flag to know whether the agent is alive

unsigned int **breedCountdown**
*The interval of time after which the objects of the classes that inherit from **Agent** will breed.*

Coordinates location
The current location of the object in the board.

SimulationBoard * board
The board in which the object resides.

Detailed Description

An abstract class representing an entity at play in the simulation. It is the base class of the **Coyote** and **Roadrunner** classes.

Constructor & Destructor Documentation

Agent::Agent (SimulationBoard * *board*, Coordinates *location*, unsigned int *breedCountdown*, BoardOccupantTypes *type*)

The constructor for the **Agent** class

Parameters:

board	The board on which the objects of the classes that inherit from **Agent** will reside
location	The location in which the objects of the classes that inherit from **Agent** will reside
breedCountdown	The interval of time after which the objects of the classes that inherit from **Agent** will breed
type	The type of the child class being instantiated (could be coyote or roadrunner)

Member Function Documentation

virtual Agent* Agent::act () `[pure virtual]`

Pure virtual function to be overriden by child class. It executes the actions that the object will make in its given turn. This action includes move, breed, and die

Returns:

If the object breeds and creates a child object, then it returns a pointer to the child object. Else, it returns NULL

Implemented in **Roadrunner** (*p.pagenum*), and **Coyote** (*p.pagenum*).

virtual Agent* Agent::breed () `[protected], [pure virtual]`

Pure virtual function to be overriden by child class. It executes the breed mechanism of the object.

Returns:

Returns a pointer to a brand new object spawned by this object. If breeding does not take place, a NULL pointer is returned

void Agent::die () `[protected], [virtual]`

The default implementation of the **die()** function for any class that inherits from **Agent**. Simply sets alive flag to false

Coordinates Agent::findRandomViableLocation (Coordinates *currentPosition*, BoardOccupantTypes *criteria*) `[protected]`

Implementation function that returns a random adjacent location out of all four adjacent locations (up, down, left, right) that satisfy the given criteria. If no such location is found, then an uninitialized location is returned

Parameters:

currentPosition	The position in relation to which the random adjacent location is to be calculated \criteria The criteria that must be met by the randomly selected location-to-be-returned

Returns:

The randomly selected adjacent location that also fulfills the criteria. If no location found, an uninitialized location is returned

bool Agent::isAlive ()

Checks if the object is alive.

Returns:

True if alive, false if dead

virtual void Agent::move () `[protected], [pure virtual]`

Pure virtual function to be overriden by child class. It executes the move mechanism of the object.

The documentation for this class was generated from the following files:

Agent.h
Agent.cpp

Coordinates Class Reference

`#include <SimulationBoard.h>`

Public Member Functions

Coordinates ()
Coordinates (unsigned int row, unsigned int column)
void **setCoordinates** (unsigned int row, unsigned int column)
void **setToNULL** ()
bool **initialized** ()
unsigned int **getRow** ()
unsigned int **getColumn** ()

Detailed Description

A class that is used to store coordinates to describe a location in a board.

Constructor & Destructor Documentation

Coordinates::Coordinates () `[inline]`

Default contructor. The **Coordinates** object is set to uninitialized

Coordinates::Coordinates (unsigned int *row*, unsigned int *column*) `[inline]`

Overloaded contructor. The **Coordinates** object has its row and column value set to the arguments provided

Parameters:

row	The row attribute to describe the location in the board
column	The column attribute to describe the location in the board

---

Member Function Documentation

unsigned int Coordinates::getColumn () `[inline]`

Used to get the column attribute

Returns:
the column value

unsigned int Coordinates::getRow () `[inline]`

Used to get the row attribute

Returns:
the row value

bool Coordinates::initialized () `[inline]`

Checks to see if the **Coordinates** object is "initialized".

Returns:

False if not initialized. Else, true

void Coordinates::setCoordinates (unsigned int *row*, unsigned int *column*) [inline]

Sets the **Coordinates** object's row and column attribute to new values

Parameters:

row	The row attribute to describe the location in the board
column	The column attribute to describe the location in the board

void Coordinates::setToNULL () [inline]

Uninitializes the **Coordinates** object by setting locationInitialized flag to false

---

The documentation for this class was generated from the following file:

SimulationBoard.h

Coyote Class Reference

`#include <Coyote.h>`
Inherits **Agent**.

Public Member Functions

Coyote (**SimulationBoard *board**, **Coordinates location**)
Agent * act ()

Additional Inherited Members

Detailed Description

The class that defines the attributes and functionalities of the coyote in the Coyote-Roadrunner simulation

---

Constructor & Destructor Documentation

Coyote::Coyote (SimulationBoard * *board*, Coordinates *location*)

The constructor for the **Coyote** class

Parameters:

board	The board on which this coyote object will reside
location	The location in which this coyote object will reside

---

Member Function Documentation

Agent * Coyote::act () `[virtual]`

Executes the actions that the coyote object will make in its given turn. This action includes move, breed, and die.

Returns:

If the coyote object breeds and creates a child object, then it returns a pointer to the child object. Else, it returns NULL

Implements **Agent** (*p.pagenum*).

The documentation for this class was generated from the following files:

Coyote.h
Coyote.cpp

Roadrunner Class Reference

`#include <Roadrunner.h>`
Inherits **Agent**.

Public Member Functions

Roadrunner (**SimulationBoard *board**, **Coordinates location**)
Agent * act ()

Additional Inherited Members

Detailed Description

The class that defines the attributes and functionalities of the roadrunner in the Coyote-Roadrunner simulation

---

Constructor & Destructor Documentation

Roadrunner::Roadrunner (SimulationBoard * *board*, Coordinates *location*)

The constructor for the **Roadrunner** class

Parameters:

board	The board on which this roadrunner object will reside
location	The location in which this roadrunner object will reside

---

Member Function Documentation

Agent * Roadrunner::act () `[virtual]`

Executes the actions that the roadrunner object will make in its given turn. This action includes move, breed, and die.

Returns:

If the roadrunner object breeds and creates a child object, then it returns a pointer to the child object. Else, it returns NULL

Implements **Agent** (*p.pagenum*).

The documentation for this class was generated from the following files:

Roadrunner.h
Roadrunner.cpp

SimulationBoard Class Reference

`#include <SimulationBoard.h>`

Public Member Functions

void **addAgent** (BoardOccupantTypes agentType, **Coordinates** location)
void **removeAgent** (**Coordinates** location)
void **moveAgent** (**Coordinates** sourceLocation, **Coordinates** destinationLocation)
bool **outOfBounds** (**Coordinates** location)
bool **existsHere** (BoardOccupantTypes type, **Coordinates** location)
unsigned int **numOfRoadRunners** ()
unsigned int **numOfCoyotes** ()
bool **boardIsEmpty** ()
void **printBoard** ()

Static Public Member Functions

static **SimulationBoard** * **get_instance** (unsigned int numOfRows, unsigned int numOfColumns)

Detailed Description

The singleton class that defines the virtual board on which the Coyote-Roadrunner simulation is run

---

Member Function Documentation

void SimulationBoard::addAgent (BoardOccupantTypes *agentType*, Coordinates *location*)

Adds an agent of the specified type to the **SimulationBoard** at the specified location

Parameters:

agentType	The type of the agent to be added to the board
location	The location to which the agent is to be added

bool SimulationBoard::boardIsEmpty ()

Checks to see if the board is board is empty

Returns:

True if board is empty, false otherwise

bool SimulationBoard::existsHere (BoardOccupantTypes *type*, Coordinates *location*)

First, makes sure given location is not out of bounds and then checks to see if the given type resides in that location

Parameters:

type	The type that is to be checked for in the location
location	The location in which the type to be checked

Returns:

true If the location exists and the location is occupied by the given type, false otherwise

SimulationBoard * SimulationBoard::get_instance (unsigned int *numOfRows*, unsigned int *numOfColumns*) `[static]`

Used to get the static instance of the **SimulationBoard** class

Parameters:

numOfRows	The number of rows in the board
numOfColumns	The number of columns in the board

Returns:

The static instance of the **SimulationBoard**

void SimulationBoard::moveAgent (Coordinates *sourceLocation*, Coordinates *destinationLocation*)

Moves an agent in the **SimulationBoard** from one location to another. If the destination is same as the source, then does nothing

Parameters:

sourceLocation	The location at which the agent currently resides
destinationLocation	The location to which the agent is to be moved to

unsigned int SimulationBoard::numOfCoyotes ()

Returns the total number of coyotes currently residing in the board

Returns:

The total number of coyotes in the board

unsigned int SimulationBoard::numOfRoadRunners ()

Returns the total number of roadrunners currently residing in the board

Returns:

The total number of roadrunners in the board

bool SimulationBoard::outOfBounds (Coordinates *location*)

Checks if the given location is out of bounds, i.e. checks if the given does not location exist in the board

Parameters:

location	The locatoin whose validity is to be checked

Returns:

true if the location does not exist, false if the location does exist

void SimulationBoard::printBoard ()

Prints the contents of the board

void SimulationBoard::removeAgent (Coordinates *location*)

Removes an agent from the **SimulationBoard** at the specified location, rendering the location unoccupied

Parameters:

location	The location from which the agent is to be removed

The documentation for this class was generated from the following files:

SimulationBoard.h
SimulationBoard.cpp

Index

INDEX