Disease Simulation

1.0

Generated by Doxygen 1.8.17

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

Chapter 3

Class Index

3.1 Class List

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

Chapter 4

Namespace Documentation

4.1 IDisease Namespace Reference

Classes

- class IDisease

  Interface IDisease allows setting the strength and growth condition of a disease.

Variables

- **ABC** = object

4.1.1 Detailed Description

Disease Interface.

Chapter 5

Class Documentation

5.1 Actor.Actor Class Reference

Actor class, which is the base class for Disease objects.

Inheritance diagram for Actor.Actor:
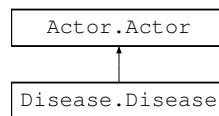
Public Member Functions

- def __init__ (self)

    Construct a new Actor object.
- def getID (self)

    Used for testing.
- def Iteration (self)

    Used for testing.
- def act (self)

    Prints on screen in the format "Iteration $<ID>$: Actor $<Actor\ ID>$".
- def setLocation (self, x, y)

    Sets the cell coordinates of this object.
- def addedToWorld (self, world)

    Sets the world this actor is into.
- def getWorld (self)

    Gets the world this object in into.
- def getX (self)

    Gets the X coordinate of the cell this actor object is into.
- def getY (self)

    Gets the Y coordinate of the cell this actor object is into.
- def __str__ (self)

    Return a string with this actor ID and position.

5.1.1   Detailed Description

Actor class, which is the base class for Disease objects.

Resolução da AD1 2020.2 - Prog. de Interfaces Gráficas - TSC/UFF

Author

ARIADNE GONÇALVES PINHEIRO Matrícula: 19113050151 Pólo: Duque de Caxias

Date

26/08/2020

5.1.2   Constructor & Destructor Documentation

5.1.2.1   __init__()

```
def Actor.Actor.__init__ (
            self )
```

Construct a new Actor object.

- Sets the initial values of its member variables.

- Sets the unique ID for the object and initializes the reference to the World object to which this Actor object belongs to null.

- The ID of the first Actor object is 0.

- The ID gets incremented by one each time a new Actor object is created.

- Sets the iteration counter to zero and initialize the location of the object to cell (0,0).

Reimplemented in Disease.Disease.

5.1.3   Member Function Documentation

5.1.3.1 act()

```
def Actor.Actor.act (
            self )
```

Prints on screen in the format "Iteration <ID>: Actor <Actor ID>".

The $< ID >$ is replaced by the current iteration number. $< ActorID >$ is replaced by the unique ID of the Actor object that performs the act(self) method.

For instance, the actor with ID 1 shows the following result on the output screen after its act(self) method has been called twice.

```
 Iteration 0: Actor 1
 Iteration 1: Actor 1
```

Reimplemented in Disease.Disease.

5.1.3.2 addedToWorld()

```
def Actor.Actor.addedToWorld (
            self,
            world )
```

Sets the world this actor is into.

Parameters

world	Reference to the World object this Actor object is added.

Exceptions

RuntimeError	when world is null.

5.1.3.3 getID()

```
def Actor.Actor.getID (
            self )
```

Used for testing.

Returns

ActorID

5.1.3.4 getWorld()

```
def Actor.Actor.getWorld (
            self )
```

Gets the world this object in into.

Returns

the world this object belongs to

5.1.3.5 getX()

```
def Actor.Actor.getX (
            self )
```

Gets the X coordinate of the cell this actor object is into.

Returns

the x coordinate of this Actor object.

5.1.3.6 getY()

```
def Actor.Actor.getY (
            self )
```

Gets the Y coordinate of the cell this actor object is into.

Returns

the y coordinate of this Actor object.

5.1.3.7 Iteration()

```
def Actor.Actor.Iteration (
            self )
```

Used for testing.

Returns

number of iterations

5.1.3.8 setLocation()

```
def Actor.Actor.setLocation (
            self,
            x,
            y )
```

Sets the cell coordinates of this object.

Parameters

x	the column.
y	the row.

Exceptions

ValueError	when x < 0 or x >= world width,
ValueError	when y < 0 or y >= world height,
RuntimeError	when the world is null.

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

- Actor.py

5.2 ActorTest.ActorTest Class Reference

ActorTest class.

Inheritance diagram for ActorTest.ActorTest:
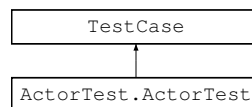
Public Member Functions

- def testConstructorAndGetMethods (self)

    Test the initialization of actor.
- def testSetLocationRuntimeError (self)

    Test the exception that happens when the world is null upon setting a location.

5.2.1 Detailed Description

ActorTest class.

Resolução da AD1 2020.2 - Prog. de Interfaces Gráficas - TSC/UFF

Author

ARIADNE GONÇALVES PINHEIRO Matrícula: 19113050151 Pólo: Duque de Caxias

Date

26/08/2020

5.2.2 Member Function Documentation

5.2.2.1 testSetLocationRuntimeError()

```
def ActorTest.ActorTest.testSetLocationRuntimeError (
            self )
```

Test the exception that happens when the world is null upon setting a location.

Exceptions

RuntimeError	when the world is null expected = RuntimeError.class

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

- ActorTest.py

5.3 Disease.Disease Class Reference

This Disease class is a sub-class of the Actor class.

Inheritance diagram for Disease.Disease:
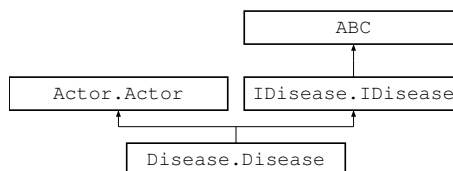
Public Member Functions

- def __init__ (self)

    Constructor.
- def setGrowthCondition (self, lTemp, hTemp, gRate)

    Sets the disease growth rate, lower temperature and higher temperature.
- def getGrowthCondition (self)

    Returns the disease growth rate, lower temperature and higher temperature.
- def getQuadrant (self)

    Returns the quadrant of this disease.
- def act (self)

    Print on screen in the format "Iteration $<ID>$: Actor $<Actor\ ID>$." The $<ID>$ is replaced by the current iteration number.
- def getStrength (self)

    Return the disease strength of this object.
- def __str__ (self)

    Return a string with the strength, growth and quadrant of this disease.

5.3.1 Detailed Description

This Disease class is a sub-class of the Actor class.

Resolução da AD1 2020.2 - Prog. de Interfaces Gráficas - TSC/UFF

Author

ARIADNE GONÇALVES PINHEIRO Matrícula: 19113050151 Pólo: Duque de Caxias

Date

26/08/2020

5.3.2 Constructor & Destructor Documentation

5.3.2.1 __init__()

```
def Disease.Disease.__init__ (
            self )
```

Constructor.

- Call its superclass's default constructor.

- Initialize the lower bound and the upper bound temperatures for the growth rate to 0.

- Set the growth rate to 0.

- Set the disease strength to 1.

Reimplemented from Actor.Actor.

5.3.3 Member Function Documentation

5.3.3.1 act()

```
def Disease.Disease.act (
            self )
```

Print on screen in the format "Iteration <ID>: Actor <Actor ID>." The $< ID >$ is replaced by the current iteration number.

$< ActorID >$ is replaced by the unique ID of the Actor object that performs the act() method.

Reimplemented from Actor.Actor.

5.3.3.2 getGrowthCondition()

```
def Disease.Disease.getGrowthCondition (
            self )
```

Returns the disease growth rate, lower temperature and higher temperature.

Returns

growth rate, lower temp and higher temp

5.3.3.3 getQuadrant()

```
def Disease.Disease.getQuadrant (
            self )
```

Returns the quadrant of this disease.

Returns

0, 1, 2 or 3.

5.3.3.4 getStrength()

```
def Disease.Disease.getStrength (
            self )
```

Return the disease strength of this object.

Returns

disease strength of the object.

Reimplemented from IDisease.IDisease.

5.3.3.5 setGrowthCondition()

```
def Disease.Disease.setGrowthCondition (
            self,
            lTemp,
            hTemp,
            gRate )
```

Sets the disease growth rate, lower temperature and higher temperature.

Parameters

lTemp	Lower bound temperature for the disease to grow at this gRate.
hTemp	Upper bound temperature for the disease to grow at this gRate.
gRate	The growth rate.

Reimplemented from IDisease.IDisease.

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

- Disease.py

5.4 DiseaseTest.DiseaseTest Class Reference

DiseaseTest class.

Inheritance diagram for DiseaseTest.DiseaseTest:

Public Member Functions

- def testStrength (self)

    Test the initial value of strength in different quadrants.

5.4.1 Detailed Description

DiseaseTest class.

Resolução da AD1 2020.2 - Prog. de Interfaces Gráficas - TSC/UFF

Author

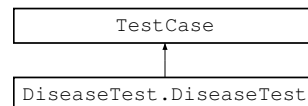ARIADNE GONÇALVES PINHEIRO Matrícula: 19113050151 Pólo: Duque de Caxias
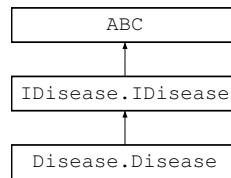
Date

26/08/2020

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

- DiseaseTest.py

5.5 IDisease.IDisease Class Reference

Interface IDisease allows setting the strength and growth condition of a disease.

Inheritance diagram for IDisease.IDisease:

```
┌─────────────────────┐
│         ABC         │
└─────────────────────┘
           ▲
           │
┌─────────────────────┐
│   IDisease.IDisease  │
└─────────────────────┘
           ▲
           │
┌─────────────────────┐
│   Disease.Disease    │
└─────────────────────┘
```

Public Member Functions

- def setGrowthCondition (self, lTemp, hTemp, gRate)

    Set the growth condition of a Disease object to gRate.
- def getStrength (self)

    Return the disease strength of the object implements this interface.

5.5.1 Detailed Description

Interface IDisease allows setting the strength and growth condition of a disease.

Resolução da AD1 2020.2 - Prog. de Interfaces Gráficas - TSC/UFF

Author

ARIADNE GONÇALVES PINHEIRO Matrícula: 19113050151 Pólo: Duque de Caxias

Date

26/08/2020

5.5.2 Member Function Documentation

5.5.2.1 setGrowthCondition()

```
def IDisease.IDisease.setGrowthCondition (
            self,
            lTemp,
            hTemp,
            gRate )
```

Set the growth condition of a Disease object to gRate.

The value of gRate gets multiplied to the current disease strength only when the disease is located in the world region with the average temperature in between the values of lTemp and hTemp.
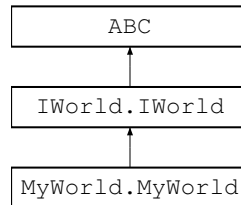
Reimplemented in Disease.Disease.

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

- IDisease.py

5.6 IWorld.IWorld Class Reference

Interface IWorld allows initializing and setting diseases for a world.

Inheritance diagram for IWorld.IWorld:

```
┌─────────────────────┐
│         ABC         │
└─────────────────────┘
          ▲
          │
┌─────────────────────┐
│    IWorld.IWorld    │
└─────────────────────┘
          ▲
          │
┌─────────────────────┐
│  MyWorld.MyWorld    │
└─────────────────────┘
```

Public Member Functions

- def **prepare** (self)
- def **setTemp** (self, quad, temp)
- def **getTemp** (self, quad)
- def **getObjects** (self)
- def **getSumStrength** (self)
- def **initDiseases** (self, numDisStr)
- def **initLocations** (self, locationsStr, diseaseArr)
- def **initGrowthConditions** (self, growthStr, diseaseArr)
- def **initTemps** (self, tempStr)

5.6.1 Detailed Description

Interface IWorld allows initializing and setting diseases for a world.

Resolução da AD1 2020.2 - Prog. de Interfaces Gráficas - TSC/UFF

Author

ARIADNE GONÇALVES PINHEIRO Matrícula: 19113050151 Pólo: Duque de Caxias
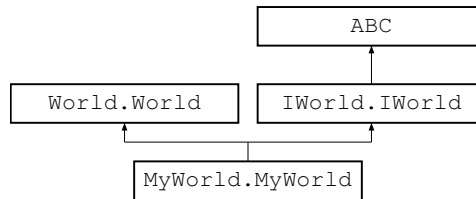
Date

26/08/2020

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

- IWorld.py

5.7 MyWorld.MyWorld Class Reference

This MyWorld class is a sub-class of the World class.

Inheritance diagram for MyWorld.MyWorld:

Public Member Functions

- def __init__ (self)

  Call the constructor of the World class with the width and height of 720 and 640 cells , respectively .
- def act (self)

  This method overrides the act () method in the World class .
- def prepare (self)

  Prepare the world .
- def setTemp (self, quad, temp)

  Set the temperature of the region of the world to the value of temp .
- def getTemp (self, quad)

  Return the temperature of the world region with the ID of quadID .
- def initDiseases (self, numDisStr)

  Create Disease objects ; the number of the objects equals to the value passed in numDisStr .
- def initLocations (self, locationsStr, diseaseArr)

  Add each Disease object into the MyWorld object implementing this method according to the information in locationStr .
- def initGrowthConditions (self, growthStr, diseaseArr)

  Set the lower bound and upper bound temperature and the growth rate for each disease according to the input growthStr .
- def initTemps (self, tempStr)

  Set the temperature for each quadrant of the MyWorld according to the value of the tempStr .
- def getSumStrength (self)

  Return the total disease strength of all the diseases in the class implementing this interface .

Public Attributes

- **avgTemp**

5.7.1 Detailed Description

This MyWorld class is a sub-class of the World class.

Resolução da AD1 2020.2 - Prog. de Interfaces Gráficas - TSC/UFF

Author

ARIADNE GONÇALVES PINHEIRO Matrícula: 19113050151 Pólo: Duque de Caxias

Date

26/08/2020

5.7.2 Constructor & Destructor Documentation

5.7.2.1 __init__()

```
def MyWorld.MyWorld.__init__ (
            self )
```

Call the constructor of the World class with the width and height of 720 and 640 cells , respectively .

Initialize a list to keep the average temperature of each world region (quadrant).

Call the prepare () method .

5.7.3 Member Function Documentation

5.7.3.1 act()

```
def MyWorld.MyWorld.act (
            self )
```

This method overrides the act () method in the World class .

This method prints :

" Iteration $<$ ITRID $>$: World disease strength is $<$ WorldDisease $>$" where $<$ ITRID $>$ is replaced by the current iteration number and $<$ WorldDisease $>$ is replaced by the returned value of getSumStrength () in 2 decimal places . An example is below .

Iteration 0: World disease strength is 2.00 Iteration 1: World disease strength is 3.00

Reimplemented from World.World.

5.7.3.2 getTemp()

```
def MyWorld.MyWorld.getTemp (
            self,
            quad )
```

Return the temperature of the world region with the ID of quadID .

The valid value is between zero and three inclusive .

Exceptions

ValueError	if the quad id is not within 0 and 3

Reimplemented from IWorld.IWorld.

5.7.3.3 initDiseases()

```
def MyWorld.MyWorld.initDiseases (
            self,
            numDisStr )
```

Create Disease objects ; the number of the objects equals to the value passed in numDisStr .

Return a list of object references to the created Disease objects .

An example of a valid numDisStr is below .

Ex : "2"

If numDisStr is None or it cannot be converted to a positive integer , print a message on screen " Check the NumDiseases line in simulation . config ." and return None .

No exceptions are thrown .

Reimplemented from IWorld.IWorld.

5.7.3.4 initGrowthConditions()

```
def MyWorld.MyWorld.initGrowthConditions (
            self,
            growthStr,
            diseaseArr )
```

Set the lower bound and upper bound temperature and the growth rate for each disease according to the input growthStr .

An example of a valid string for two Disease objects is :

Ex : "10.0 ,15.0 ,2.0;10.0 ,13.0 ,3.0"

If growthStr is empty or not in the correct format or does not have all the growth for all the Disease objects in the Disease array , print on screen " Check the DiseasesGrowth line in simulation . config ." and return -1.

Return 0 for a successful initialization of the Disease growth conditions . No exceptions are thrown .

Reimplemented from IWorld.IWorld.

5.7.3.5 initLocations()

```
def MyWorld.MyWorld.initLocations (
            self,
            locationsStr,
            diseaseArr )
```

Add each Disease object into the MyWorld object implementing this method according to the information in locationStr .

An example of a locationStr is "200 ,200;400 ,480". This means that the first Disease is planted at cell (200 ,200) and the second Disease is at cell (400 , 480).

If the locationStr is empty or not in the correct format or does not have all the cell coordinates of all the Disease objects , print on screen " Check the Locations line in simulation . config " and return -1.

Return 0 for a successful initialization of the Disease locations . No exceptions are thrown .

Reimplemented from IWorld.IWorld.

5.7.3.6 initTemps()

```
def MyWorld.MyWorld.initTemps (
            self,
            tempStr )
```

Set the temperature for each quadrant of the MyWorld according to the value of the tempStr .

An example of tempStr is below . The region temperatures for regions 0 , 1 , 2 , and 3 are 12 , 20 , 50 , and 100 , respectively .

Return 0 for a successful initialization of the quadrant temperatures . No exceptions are thrown .

Ex : "12;20;50;100"

If tempStr is empty or not in the correct format or does not have all the temperatures of all the regions , print on screen " Check the Temperature line in simulation . config ." and return -1

Reimplemented from IWorld.IWorld.

5.7.3.7 prepare()

```
def MyWorld.MyWorld.prepare (
            self )
```

Prepare the world .

Open a text file named " simulation . config " in the current path (directly under the project directory). Parse the configuration file for the number of Disease objects , the cell locations of these objects , the growth rates , and the temperature ranges associated with individual growth rates . Read Section 4 on the content of the configuration file before reading the rest .

Exceptions

IOError	when there are problems with opening a file, reading, or writing to a file.

Reimplemented from [IWorld.IWorld](#).

5.7.3.8 setTemp()

```
def MyWorld.MyWorld.setTemp (
            self,
            quad,
            temp )
```

Set the temperature of the region of the world to the value of temp .

The quadID indicates the region . The valid value is between [0 , 3]. Any value of float is accepted for temp .

Exceptions

ValueError	if the quad id is not within 0 and 3

Reimplemented from [IWorld.IWorld](#).

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

- MyWorld.py

5.8 World.World Class Reference

Inheritance diagram for World.World:
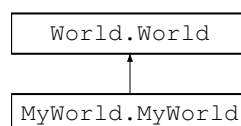
Public Member Functions

- def __**init**__ (self, worldWidth, worldHeight)
- def **createGrid** (self, h, w, d)
- def __**str**__ (self)
- def __**repr**__ (self)
- def **act** (self)
- def **addObject** (self, object, x, y)

- def **getHeight** (self)
- def **getWidth** (self)
- def **getDepth** (self)
- def **numberOfObjects** (self)
- def **getObjects** (self)
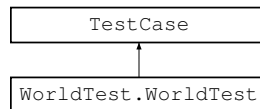- def **setGrid** (self, aGrid, numObjs)

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

- World.py

5.9 WorldTest.WorldTest Class Reference

WorldTest class.

Inheritance diagram for WorldTest.WorldTest:


```
          TestCase
             ▲
             │
    WorldTest.WorldTest
```

Public Member Functions

- def TestConstructorAndGetMethods (self)

    Test the initialization of the constructor that it is placed appropriately.
- def testGetObjects (self)

    Test that the objects that are placed are still the objects when retrieved.
- def testNumberOfObject (self)

    Test that the number of object function returns the correct result.
- def testSetGrid (self)

    Test if the copying of grid.
- def testAddObjectSyntaxError (self)

    Test that the add object function captures this error.
- def testAddObjectValueError (self)

    Test that the add object function captures invalid x and y coordinates.
- def testAddObjectRuntimeError (self)

    Test that the add object function captures null values.

5.9.1 Detailed Description

WorldTest class.

Resolução da AD1 2020.2 - Prog. de Interfaces Gráficas - TSC/UFF

Author

ARIADNE GONÇALVES PINHEIRO Matrícula: 19113050151 Pólo: Duque de Caxias

Date

26/08/2020

5.9.2 Member Function Documentation

5.9.2.1 testAddObjectRuntimeError()

```
def WorldTest.WorldTest.testAddObjectRuntimeError (
            self )
```

Test that the add object function captures null values.

Exceptions

RuntimeError	when the value being added is null expected = RuntimeError.class

5.9.2.2 testAddObjectSyntaxError()

```
def WorldTest.WorldTest.testAddObjectSyntaxError (
            self )
```

Test that the add object function captures this error.

Exceptions

SyntaxError	when already max number of objects are in that cell expected = SyntaxError.class

5.9.2.3 testAddObjectValueError()

```
def WorldTest.WorldTest.testAddObjectValueError (
            self )
```

Test that the add object function captures invalid x and y coordinates.

Exceptions

ValueError	when x and y are not within the boundaries of grid expected = ValueError.class

5.9.2.4 testSetGrid()

```
def WorldTest.WorldTest.testSetGrid (
            self )
```

Test if the copying of grid.

Exceptions

Exception	if the aGrid consists of invalid properties

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

- WorldTest.py

Index