СУ “Св. Климент Охридски”,
ФМИ – Софтуерно инженерство
Курсов проект по Обектно-ориентирано програмиране

God

Йоан Евгениев Стоянов, Факултетен № 61830

Съдържание

1. Въведение

Целта на проекта е да симулираме Бог, който може да контролира планети и същества, които ги населяват. Същества са 4 вида: Entity, Animal, Human, God. Бог има различни способности, сред които да унищожи дадена планета или да насели планета със същества.

2. Описание на приложените алгоритми

А) **Клас Point2D**: double getDistance(Point2D p1, Point2D p2)

- Изчисляване на разстоянието между 2 точки чрез питагоровата теорема.

Б) Клас Entity: `void Move()`

```
RandomNumberGenerator rng = new RandomNumberGenerator();
int moveRand = rng.generateNumberRange(2);
int deltaPositionX = 0;
int deltaPositionY = 0;
if (moveRand % 2 == 0) {
    deltaPositionX = position.getX() + rng.generateCoordinate();
    deltaPositionY = position.getY() + rng.generateCoordinate();
    if(deltaPositionX<=PLANET_LENGHT && deltaPositionY<=PLANET_LENGHT){
        position.setX(deltaPositionX);
        position.setY(deltaPositionY);
    }
}
else if (moveRand % 2 == 1){
    deltaPositionX = position.getX() - rng.generateCoordinate();
    deltaPositionY = position.getY() - rng.generateCoordinate();
    if(deltaPositionX>=0 && deltaPositionY>=0){
        position.setX(deltaPositionX);
        position.setY(deltaPositionY);
    }
}
this.setState(State.Moving);
```

- На базата на произволен принцип на единицата се променя местоположението в завимост от границите,които са зададени.

В)Клас Simulator: void doStuff(Entity entity)

```
private void doStuff(Entity entity) throws IllegalAccessException, IllegalArgumentException,
        InvocationTargetException, NoSuchMethodException, SecurityException, InstantiationException,
        InterruptedException {
    RandomNumberGenerator rng = new RandomNumberGenerator();
    String[] methodNames = { "Analyze", "Sleep", "Eat", "SearchingForFood" };
    Thread.sleep(3000);
    int num = rng.generateNumberRange(methodNames.length);
    Method[] methods = entity.getClass().getMethods();
    for (Method method : methods) {
        if (method.getName().equals(methodNames[num])) {
            method.invoke(entity.getClass().newInstance());
            break;
        }
    }
}
```

- Ползваме reflection,за да можем да вземем на дадения entity името на класа и съответно неговите методи. След това се итерира през тях докато се намери някой от изброените методи за действие(масива methodNames) и след това го извиква този метод.

3. Описание на програмния код

А)Class Point2D:

```
public class Point2D {
    private int x;
    private int y;

    public Point2D(int x, int y) {
    //constructor
    public void setX(int x) {
    //set X coordinate
    public void setY(int y) {
    //set Y coordinate
    public int getX() {
    //get X coordinate
    public int getY() {
    //get Y coordinate
    public double getDistance(Point2D p1, Point2D p2) {
    //get distance between 2 points
}
```

Б)EntityType и State са enums съдържащи съответно *entity*, *animal*, *human*, *god*, *unknown за EntityType и Moving*, *Attacking*, *Eating*, *SearchingForFood*, *Sleeping*, *Analyzing*, *Unknown за State*.

В)Entity

```
public class Entity {
    protected String name;
    protected double energy;
    protected double size;
    protected double weight;
    protected Point2D position;
    protected double strength;
    protected State state;
    protected boolean isAlive = true;
    protected EntityType entity;
    final static private int PLANET_LENGHT = 400;
    //upper bound for entities to move
    public Entity() {
    //default constructor with random values
    public Entity(String name, double energy, double size, double weight, Point2D position, double strength, State
    //constructor with parameter values
    public String getName() {
    //returns name of entity
    public void setName(String name) {
    //sets name of entity
    public EntityType getEntity() {
    //returns the entity's type
    public double getEnergy() {
    //returns energy of entity
    public void setEnergy(double energy) {
    //sets energy of entity
```

```
    public void setEnergy(double energy) {
    //sets energy of entity
    public double getSize() {
    //returns size of entity
    public void setSize(double size) {
    //sets size of entity
    public double getWeight() {
    //returns weight of entity
    public void setWeight(double weight) {
    //sets weight of entity
    public Point2D getPosition() {
    //returns position of entity
    public void setPosition(Point2D position) {
    //sets position(x and y coordinate) of entity
    public double getStrength() {
    //returns strength of entity
    public void setStrength(double strength) {
    //sets strength of entity
    public State getState() {
    //gets state
    public void setState(State state) {
    //sets state from enum State
    public void Attack(Entity ent) {
    //void method which lowers the energy of the given entity
    public void Move() {
    //void method which moves the entity to X position(X is randomly generated)
```

Г)Animal наследява Entity

```
public class Animal extends Entity {
    public Animal(){
    //default constructor
    public Animal(String name, double energy, double size, double weight, Point2D position, double strength, State s
    //constructor with parameters
    public void Eat() {
    //action changing the state to Eating
    public void Sleep() {
    //action changing the state to Sleeping
    public void SearchingForFood() {
    //action changing the state to SearchingForFood

}
```

Д)Human наследява Animal

```
public class Human extends Animal{
    public Human(){
    //default constructor with random values set
    public Human(String name, double energy, double size, double weight, Point2D position, double strength, State st
    //constructor with parameter values
    public void Analyze(){
    //setting the state to Analyzing
}
```

Е)God наследява Human

```
public class God extends Human{
    public God(String name, double energy, double size, double weight, Point2D position, double strength, State stat
    //constructor with parameter values
    public God() {
    //default constructor with random values
```

Ж)Planet

```
public class Planet {
    private String name;
    private List<Entity> population;
    //Concurrent ArrayList for population on planet
    private boolean isDestroyed;
    private RandomNumberGenerator rng = new RandomNumberGenerator();
    public Planet(){
    //default constructor
    public List<Entity> getPopulation(){
    //returns a list of the population
    public String getName(){
    //returns the name of the planet
    public int getPopulationCount(){
    //returns size of population

public void setPopulation(List<Entity> l){
//sets this planet's population

public boolean isDestroyed(){
//returns true if planet is destroyed

public void setDestroyed(){
//changes planet to destroyed

public void addPopulation(Entity e){
//adds an entity to population

public void destroyPopulation(){
//removes population from planet
```

З)Scene

```
public class Scene {
    private List<Planet> planets = new CopyOnWriteArrayList<Planet>();
    //concurrent ArrayList
    public List<Planet> getPlanets(){
    //returns an ArrayList of all planets
    public void createCreature(Planet pl,EntityType et){
    //creates a new entity(or subclass unit) with random variables and adds it to the planet's population
    public void createPlanet(){
    //adds a new planet(created with random variables) to the ArrayList
    public void destroyPlanet(Planet pl){
    //removes the given planet from the ArrayList
}
```

И)Simulator

```
public class Simulator {
    private God player;
    private Scene scene;
    final static protected int PLANET_SIZE = 300;
    //maximum elements that can be added at a time to a planet
    final static protected int MAXIMUM_PLANETS = 9;
    //number of maximum planets
    public Simulator() {
    //default Constructor
    public void Run() {
    //starts the Menu
    private void createPlanet() {
    //creates a new Planet with random variables and adds it to ArrayList
    private void destroyPopulation(String input) {
    //removes population from given planet
    private void destroyPlanet(String input) {
    //removes given planet from ArrayList
    private void showStatistics() {
    //prints each planet's population size
    private void addCreatures(String input) {
    //adds n entities(or subclass units) to given planet
    private void showHelp() {
    //shows the main menu
    private void removeDeadEntities(List<Entity> entities) {
    //updates the ArrayList of entities with dead ones removed
    private void moveEntities(List<Entity> entities) throws InterruptedException {
    //iterates through entities ArrayList and performs Move method on them

    private void executeAnAction(Planet pl, List<Entity> entities) throws InterruptedException {
    //chooses randomly an action
    private void doStuff(Entity entity) throws IllegalAccessException, IllegalArgumentException,
    //invokes a method that does a action based on random choosing by using reflection
    public void Update() throws InterruptedException {
    //updates each entity's position and performs an action chosen by given methods
    private boolean checkValidInput(String input) {
    //boolean function which checks for valid input
    public void Menu() {
    //main menu in which a command is given from the console
}
```

Й)RandomNumberGenerator

```
public class RandomNumberGenerator {
    private Random rand;
    private List<String> visited;
    //ArrayList for which planet names are currently taken
    public RandomNumberGenerator(){
    //default constructor
    public String generatePlanetName(){
    //generates a planet name from an array of names
    public int generateNumberRange(int range){
    //generates a number from 0 to range
    public int generateCoordinate(){
    //generates a coordinate from 0 to 50
    public int generateNumber(){
    //generates a number from 0 to 100
    public String generateName(EntityType et){
    //generates an entity name by joining the entity type and a random number
    public double generateStrength(){
    //generates a strength number from 0 to 200
}
```

К)MainGame

```
public class MainGame {

    public void StartGame(){
    //runs up the game with 2 threads:1 for menu and 1 for updates
}
```

4. Използвани технологии

IDE: Eclipse Version: Luna Service Release 2 (4.4.2)
Език: Java