Object-Oriented Programming

Lab #2

Box of particles

NW

W

SW

S

SE

E

NE

N

y

x

a particle

Imagine that we have a box to store particles. Initially, we place randomly 3 particles in that box. After each step, these particles will move freely inside the box. If two particles collide, a new particle will be placed randomly in the box.   
We want to simulate the movement particles for n steps and count the number of particles in the box.

Implement a box of particles (write a class of box, a class for particle) in such a way that

1. A box has a fixed size: fixed width and height (10pts)
2. Each particle has a position (x, y) where 0 ≤ x ≤ width of the box, and 0 ≤ y ≤ height of the box (10pts)
3. A particle can move in one of the directions below but cannot move out of the box (20pts).  
   + North (decreasing its y by 1),   
   + North East (decreasing its y by 1 and increasing its x by 1),   
   + East (increasing its x by 1),   
   + South East (increasing its y by 1 and increasing its x by 1)  
   + South (increasing its y by 1),   
   + South West (increasing its y by 1 and decreasing its x by 1),   
   + West (decreasing its x by 1),   
   + North West (decreasing its y by 1 and decreasing its x by 1)

Hint: declare an enum type for Direction

1. If two particles collide, a new particle will be placed randomly in the box(20pts)

and a class for simulation where for each step,

1. It makes all particles in the box move (5pts)
2. It shows the number of particles in the box (5pts)
3. It visualizes the box with particles inside (10pts)\*  
    -------------------------------

| \* \* \* |

| \* \*\* |

| \* |

-------------------------------

1. Search about singleton pattern and make the box as a singleton (20pts)

Reference: https://docs.oracle.com/javase/tutorial/java/javaOO/enum.html

**Particle.java**

public class Particle {

    public int x;

    public int y;

    public Particle(int y, int x){

        this.x = x;

        this.y = y;

    }

    public void setX(int x){

        this.x = x;

    }

    public void setY(int y){

        this.y = y;

    }

    public int getX(){

        return x;

    }

    public int getY(){

        return y;

    }

    public void okay(int x\_move,int y\_move){

        if (x\_move==1){

            x++;

        }

        if (x\_move==2){

            x--;

        }

        if (y\_move==1){

            y++;

        }

        if (y\_move==2){

            y--;

        }

    }

    public void move(int height, int width){

        int x\_move=0;

        int y\_move=0;

        while (x\_move==0 && y\_move==0){

            x\_move=(int)(Math.random()\*3);

            y\_move=(int)(Math.random()\*3);

        }

        if (x==0 && y==0){

            if (x\_move==2){

                x\_move=0;

            }

            if (y\_move==2){

                y\_move=0;

            }

        }

        if (x==0 && y==height-1){

            if (x\_move==2){

                x\_move=0;

            }

            if (y\_move==1){

                y\_move=0;

            }

        }

        if (x==width-1 && y==0){

            if (x\_move==1){

                x\_move=0;

            }

            if (y\_move==2){

                y\_move=0;

            }

        }

        if (x==width-1 && y==height-1){

            if (x\_move==1){

                x\_move=0;

            }

            if (y\_move==1){

                y\_move=0;

            }

        }

        if (x==0){

            if (x\_move==2){

                x\_move=0;

            }

        }

        if (x==width-1){

            if (x\_move==1){

                x\_move=0;

            }

        }

        if (y==0){

            if (y\_move==2){

                y\_move=0;

            }

        }

        if (y==height-1){

            if (y\_move==1){

                y\_move=0;

            }

        }

        okay(x\_move,y\_move);

    }

}

**Box.java**

public class Box {

    public int width=50;

    public int height=20;

    public int numPartical=3;

    public int location[][] = new int[height][width];

    private static Box instance;

    private Box(){

    }

    public static Box getInstance(){

        if(instance == null){

            instance = new Box();

        }

        return instance;

        }

    public void display() {

        for (int j=0; j<width; j++) {

            if (location[0][j] == 1) {

                System.out.print("\*");

            } else {

                System.out.print("-");

            }

        }

        System.out.println();

        for (int i=1; i<height-1; i++) {

            for (int j=0; j<width; j++) {

                if (location[i][j] == 1) {

                    System.out.print("\*");

                } else {

                    if (j==0 || j==width-1) {

                        System.out.print("|");

                    } else {

                        System.out.print(" ");

                    }

                }

            }

            System.out.println();

        }

        for (int j=0; j<width; j++) {

            if (location[height-1][j] == 1) {

                System.out.print("\*");

            } else {

                System.out.print("-");

            }

        }

    }

    public void setPartical(int y, int x) {

        location[y][x] = 1;

    }

    public void removePartical(int y, int x) {

        location[y][x] = 0;

    }

    public int getPartical(int y, int x) {

        return location[y][x];

    }

    public void resetLocation() {

        for (int i=0; i<height; i++) {

            for (int j=0; j<width; j++) {

                location[i][j] = 0;

            }

        }

    }

    public static void main(String[] args) {

        Box b = Box.getInstance();

        Particle[] p = new Particle[b.width\*b.height];

        b.resetLocation();

        System.out.println("Initial location:");

        for (int i=0; i<b.numPartical; i++) {

            p[i] = new Particle((int)(Math.random()\*b.height), (int)(Math.random()\*b.width));

            b.setPartical(p[i].getY(), p[i].getX());

        }

        b.display();

        System.out.println();

        System.out.println("Number of Partical: "+b.numPartical);

        System.out.println();

        int z=0;

        while (z!=-1){

            System.out.println("With n = "+(z+1)+" :");

            for (int i=0; i<b.numPartical; i++) {

                b.removePartical(p[i].getY(), p[i].getX());

                p[i].move(b.height, b.width);

            }

            for (int i=0; i<b.numPartical; i++) {

                b.setPartical(p[i].getY(), p[i].getX());

            }

            int old\_numPartical=b.numPartical;

            for (int i=0; i<old\_numPartical-1; i++) {

                for (int j=i+1; j<old\_numPartical; j++) {

                    if (p[i].getX() == p[j].getX() && p[i].getY() == p[j].getY()) {

                        if (b.numPartical!=b.width\*b.height-1){

                            b.numPartical++;

                            p[b.numPartical-1] = new Particle((int)(Math.random()\*b.height), (int)(Math.random()\*b.width));

                            b.setPartical(p[b.numPartical-1].getY(), p[b.numPartical-1].getX());

                        }

                        else {

                            System.out.println("The box is full");

                            return;

                        }

                    }

                }

            }

            b.display();

            System.out.println();

            /\*for (int i=0; i<b.numPartical; i++) {

                System.out.println(p[i].getY() + " "+ p[i].getX());

            }\*/

            System.out.println("Number of Partical: "+b.numPartical);

            System.out.println();

            try {

                Thread.sleep(200);

            } catch (InterruptedException e) {

                // TODO Auto-generated catch block

                e.printStackTrace();

            }

            z++;

        }

    }

}

**Output**

