Simulation Engineering Exercises 08

Name: Chenfeng Zhu
Matrikelnummer: 450485
University: TU-Clausthal
Program: ITIS
Language: JAVA
Lecturer: Dr. Umut Durak

1. Distributed Trapeze Simulation with Graph

Assumptions:
NULL.

Design:

a) 2D: draw a line as the rope and a circle as the mass point of the artist in a canvas. Change the coordinates of the end point of the line and the center point of the circle according the length of the rope and the angle:
```
x2 = x1 - length *sin(angle);
y2 = y1 + length *cos(angle);.
```
b) 3D Simulation Design: create a virtual universe and a locale. Then create two branch groups, one of which is for 3D Shapes and the other of which is for View Point. The 3D Shapes include the appearance (color, size) and the geometry (coordinates); the View Point includes the coordinates and the light (direction and bound range). Change the coordinates of the line and the sphere as the 2D.

Main code:

a) 2D:
```
// graph
Canvas canvas;
double x1 = 150, y1 = 50, len = l * 30;
double x2 = x1 - len * Math.sin(th);
double y2 = y1 + len * Math.cos(th);
private double radius = 15;

// fresh the line:
x2 = x1 - len * Math.sin(th);
y2 = y1 + len * Math.cos(th);
Platform.runLater(new Runnable() {
    public void run() {
        GraphicsContext gc = canvas.getGraphicsContext2D();
        pendulum(gc);
        refreshDashboard();
    }
});
```

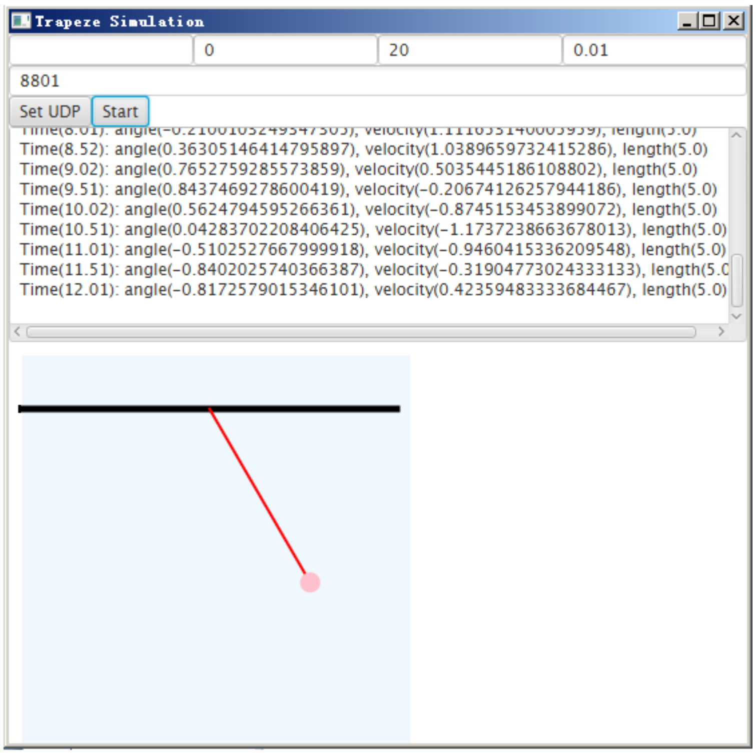
```
public void pendulum(GraphicsContext gc) {
    gc.clearRect(0, 0, 300, 300);
    gc.setFill(Color.ALICEBLUE);
    gc.fillRect(10, 10, 290, 290);
    // draw the ceiling.
    gc.setStroke(Color.BLACK);
    gc.setLineWidth(5);
    gc.strokeLine(10, y1, 290, y1);
    // draw the line
    gc.setLineWidth(2);
    gc.setStroke(Color.RED);
    gc.strokeLine(x1, y1, x2, y2);
    gc.setFill(Color.PINK);
    gc.fillOval(x2 - radius / 2, y2 - radius / 2, radius, radius);
}
```

b) 3D Simulation (not finished yet):

```
// Create the universe
SimpleUniverse universe = new SimpleUniverse();
// Create a branch group for 3D objects
BranchGroup group = new BranchGroup();

// Create a sphere as the artist.
Sphere sphere = new Sphere(0.5f);
group.addChild(sphere);

// Create a line as the rope.
float vert[] = { 0.5f, 0.5f, 0.0f, -0.5f, 0.5f, 0.0f, 1.5f, 2.5f, 3.5f };
float color[] = { 0.0f, 0.5f, 1.0f, 0.0f, 0.5f, 1.0f, 0.0f, 0.5f, 1.0f };
LineArray line = new LineArray(6, LineArray.COORDINATES | LineArray.COLOR_3);
line.setCoordinate(0, vert);
line.setColors(0, color);
LineAttributes linea = new LineAttributes();
linea.setLineWidth(2.0f);
linea.setLineAntialiasingEnable(true);
Appearance app = new Appearance();
app.setLineAttributes(linea);
Shape3D shape = new Shape3D();
shape.setGeometry(line);
shape.setAppearance(app);
group.addChild(shape);

// Create a red light that shines for 100 from the origin
// the color of the light.
Color3f light1Color = new Color3f(1.8f, 0.1f, 0.1f);
// the direction of the light.
Vector3f light1Direction = new Vector3f(4.0f, -7.0f, -12.0f);
DirectionalLight light1 = new DirectionalLight(light1Color, light1Direction);
// the bound range of the light.
BoundingSphere bounds = new BoundingSphere(new Point3d(0.0, 0.0, 0.0), 100.0);
light1.setInfluencingBounds(bounds);
group.addChild(light1);

// set the view point. look towards the trapeze.
universe.getViewingPlatform().setNominalViewingTransform();
// add the group of objects to the Universe
universe.addBranchGraph(group);
```

Images:

Run the simulation with 2D graph:



```
0    20    0.01
8801
Time(8.01): angle(-0.21001032493547303), velocity(1.1116331400003939), length(5.0)
Time(8.52): angle(0.36305146414795897), velocity(1.0389659732415286), length(5.0)
Time(9.02): angle(0.7652759285573859), velocity(0.5035445186108802), length(5.0)
Time(9.51): angle(0.8437469278600419), velocity(-0.20674126257944186), length(5.0)
Time(10.02): angle(0.562479459526636l), velocity(-0.8745153453899072), length(5.0)
Time(10.51): angle(0.04283702208406425), velocity(-1.1737238663678013), length(5.0)
Time(11.01): angle(-0.5102527667999918), velocity(-0.9460415336209548), length(5.0)
Time(11.51): angle(-0.8402025740366387), velocity(-0.31904773024333133), length(5.0
Time(12.01): angle(-0.8172579015346101), velocity(0.42359483333684467), length(5.0)
```

Run the simulation with 3D graph:

NOT YET.

Output results:

NULL.

Conclusion:

NULL.

Code:

https://github.com/sampig/SimulationEngineering