Contemporary Data Processing Technology (CCOD)

Lab 1 (September 08, 2016)

Mironchuk Vlad, AS-36

Controll two metro cars using Speed, Distance, and Brake

In this practice we made a model of 2 trains on plain loop with length of 1000 points. Every train at the beginning have the speed of 10 points and move in clockwise stream. On each iteration we change speed adding -1, 0, +1 at random. Speed should be >= 0.

When the distance between trains is 0 we stop the movement.

Here is the source code:

Index.html:

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <title>Lab1</title>
  <style>
    .status {
      display: inline-block;
      position:absolute;
    }
  </style>
</head>

<body>
    <canvas id="myCanvas" width="500" height="500">
    </canvas>
  <div class="status">
    <div style="color:#00FF00">Train 1(green):</div>
    <div id="train1"></div>
    <div style="color:#FF0000">Train 1(red):</div>
    <div id="train2"></div>
  </div>
  <script src="./script.js"></script>
</body>

</html>
```

Script.js:

```
function DrawTrainPosition(angle_1, angle_2) {
  var point_1 = ConvertAngleToCoordinate(250, 250, 200, angle_1);
  var point_2 = ConvertAngleToCoordinate(250, 250, 200, angle_2);

  console.log('x', point_1.x);
  console.log('y', point_1.y);

  var canvas = document.getElementById("myCanvas");
```

```
    var painter = canvas.getContext("2d");

    painter.clearRect(0, 0, 500, 500);

    painter.beginPath();
    painter.fillStyle = "black";
    painter.arc(250, 250, 200, 0, 2 * Math.PI);
    painter.stroke();

    painter.beginPath();
    painter.fillStyle = "#00FF00";
    painter.arc(point_1.x, point_1.y, 5, 0, 2 * Math.PI);
    painter.fill();

    painter.beginPath();
    painter.fillStyle = "#FF0000";
    painter.arc(point_2.x, point_2.y, 5, 0, 2 * Math.PI);
    painter.fill();

    console.log('Draw');
}
function ConvertAngleToCoordinate(x, y, r, angle) {
    return {
        x: Math.round(x + r * Math.cos(angle)),
        y: Math.round(y + r * Math.sin(angle))

    }
}
var angle_1 = 0;
var angle_2 = Math.PI;

var train_1 = {
    speed: 10,
    stop: 0,
    distance: 500
};
var train_2 = {
    speed: 10,
    stop: 0,
    distance: 500
};

var isStop = false;

function ChangeValue() {
    console.log('angle_1', angle_1);
    console.log('angle_2', angle_2);
    DrawTrainPosition(angle_1, angle_2);
    var divTrain1 = document.getElementById("train1");
    var divTrain2 = document.getElementById("train2");


    if (train_1.distance <= 0 || train_2.distance <= 0) {
        DrawTrainPosition(angle_1, angle_1);
```

```
        train_1.distance = train_1.distance <= 0 ? 0 : 1000;
        train_2.distance = train_2.distance <= 0 ? 0 : 1000;
        isStop = true;
    }

    divTrain1.innerText = 'speed: ' + train_1.speed + '\ndistance: ' + train_1.distance+'\n\n';
    divTrain2.innerText = 'speed: ' + train_2.speed + '\ndistance: ' + train_2.distance;

    train_1.speed += Math.floor(3 * Math.random()) - 1;
    if (train_1.speed < 0) {
        train_1.speed = 0;
    }
    train_1.distance -= train_1.speed;
    train_2.distance += train_1.speed;


    train_2.speed += Math.floor(3 * Math.random()) - 1;
    if (train_2.speed < 0) {
        train_2.speed = 0;
    }
    train_2.distance -= train_2.speed;
    train_1.distance += train_2.speed;


    console.log('train_1.speed', train_1.speed);
    console.log('train_1.distance', train_1.distance);
    console.log('train_2.speed', train_2.speed);
    console.log('train_2.distance', train_2.distance);
    angle_1 += train_1.speed * Math.PI / 500;
    angle_2 += train_2.speed * Math.PI / 500;
}

setInterval(() => {
    if (isStop) {
        return;
    }
    ChangeValue();
}, 50);
```