Contemporary Data Processing Technology (CCOD)

Lab 2 (September 15, 2016)

Fedorov Danil, AS-36

In this practice we continued use a model of two trains on plain loop with length of 1000pixels. On each iteration we change speed adding -2, 0 or +2 at random. Also we have two rules : {if x>=14 then z=1, x=x-1} so in this case speed can be >=0 & <=14 & {if y<=50 then z=9, x=x-9} – in this if some train have distance <=50 to front train => break = 9;

Source code:

Index.html

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

<head>
  <meta charset="UTF-8">
  <title></title>
  <link rel="stylesheet" href="css/style.css">
</head>

<body>

  <div class="subway_loop">
    <img class="train greentrain" src="img/train.png" width="20px"
    height="10px">
    <img class="train redtrain" src="img/train.png" width="20px"
    height="10px">
  </div>

  <div class="speed redtrainSpeed">red speed: <span
  class="redtrainSpeedValue"></span></div>
  <div class="speed greentrainSpeed">green speed: <span
  class="greentrainSpeedValue"></span></div>

  <div class="distance redtrainDistance">red to green distance: <span
  class="redtrainDistanceValue"></span></div>
  <div class="distance greentrainDistance">green to red distance: <span
  class="greentrainDistanceValue"></span></div>

  <div class="brake brakeredtrain">red train brake: <span
  class="brakeredtrainValue"></span></div>
  <div class="brake brakegreentrain">green train brake: <span
  class="brakegreentrainValue"></span></div>

  <script src="js/script.js"></script>
</body>

</html>
```

style.css

```css
.subway_loop {
  margin: 20% auto;
  width: 159px;
  height: 159px;
  border: 1px solid black;
  border-radius: 80px;
}

.train {
  position: relative;
  border: 2px solid;
}

.redtrain {
  left: 50%;
  top: 0;
  border-color: red;
}

.greentrain {
  left: 50%;
  top: 100%;
  border-color: green;
}

.speed {
  position: fixed;
  font-size: 14px;
  left: 5%;
}

.redtrainSpeed {
  bottom: 0px;
}

.greentrainSpeed {
  bottom: 20px;
}

.distance {
  position: fixed;
  font-size: 14px;
  left: 35%;
}

.redtrainDistance {
  bottom: 0px;
}

.greentrainDistance {
  bottom: 20px;
}

.brake {
  position: fixed;
  font-size: 14px;
  left: 75%;
}

.brakeredtrain {
  bottom: 0px;
}

.brakegreentrain {
  bottom: 20px;
}
```

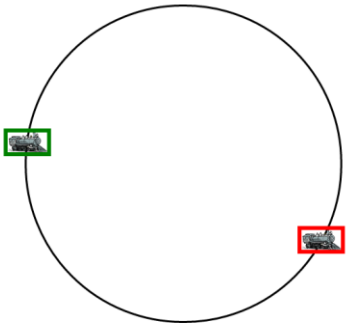
script.js

```
var greentrain = document.querySelector(".greentrain");
var redtrain = document.querySelector(".redtrain");
var redtrainSpeedValue = document.querySelector(".redtrainSpeedValue");
var redtrainDistanceValue = document.querySelector(".redtrainDistanceValue");
var greentrainSpeedValue = document.querySelector(".greentrainSpeedValue");
var greentrainDistanceValue = document.querySelector(".greentrainDistanceValue");
var brakegreentrainValue = document.querySelector(".brakeredtrainValue");
var brakeredtrainValue = document.querySelector(".brakegreentrainValue");
var startSpeed = 0.001;
var x0 = 77;
var y0 = 74;
var r = 79;
var i = -1.5;
var j = 1.5;
var start = Date.now();
var redSpeed = 0.001,
  greenSpeed = 0.001;
var rand;
var redway = 0,
  greenway = 0;


var rightRed = redtrain.style.left + 20;
var leftRed = redtrain.style.left;

var rightGreen = greentrain.style.left + 20;
var leftGreen = greentrain.style.left;


var topRed = redtrain.style.top;
var bottomRed = redtrain.style.top - 10;

var topGreen = greentrain.style.top - 10;
var bottomGreen = greentrain.style.top;

var timer = setInterval(function () {
  brakegreentrainValue.innerHTML = 0;
  brakeredtrainValue.innerHTML = 0;
  var timePassed = Date.now() - start;
  redtrain.style.left = x0 + r * Math.cos(i) - 35 + 'px';
  redtrain.style.top = y0 + r * Math.sin(i) - 5 + 'px';

  greentrain.style.left = x0 + r * Math.cos(j) - 10 + 'px';
  greentrain.style.top = y0 + r * Math.sin(j) - 5 + 'px';

  i = i + redSpeed;
  j = j + greenSpeed;
```

```
  rand = randomValue(-2, 2) / 10000;
  if (redSpeed > 0 || rand > 0) {
    redSpeed += rand;
  }
  if (redSpeed > 0.002) redSpeed = 0.002;
  redway += redSpeed;

  redtrainSpeedValue.innerHTML = Math.round(redSpeed * 10000);

  redtrainDistanceValue.innerHTML = 500 + Math.round(Math.abs(redway - greenway) / 6.0414 *
  1000);

  rand = randomValue(-2, 2) / 10000;
  if (greenSpeed > 0 || rand > 0) {
    greenSpeed += rand;
  }
  if (greenSpeed > 0.002) greenSpeed = 0.002;
  greenway += greenSpeed;

  greentrainSpeedValue.innerHTML = Math.round(greenSpeed * 10000);

  greentrainDistanceValue.innerHTML = 1000 - (500 + Math.round(Math.abs(redway - greenway) /
  6.0414 * 1000));

  //start second task code
  //{if x>=14 then z=1, x=x-1}
  if (Math.round(greenSpeed * 10000) >= 14) {
    greenSpeed -= 1;
    brakegreentrainValue.innerHTML = "1";
    console.log("green 1")
  }
  if (Math.round(redSpeed * 10000) >= 14) {
    redSpeed -= 1;
    brakeredtrainValue.innerHTML = "1";
    console.log("red 1")
  }

  //{if y<=50 then z=9, x=x-9}
  if (1000 - (500 + Math.round(Math.abs(redway - greenway) / 6.0414 * 1000)) <= 50) {
    greenSpeed -= 9;
    brakegreentrainValue.innerHTML = "9";
  }
  if (500 + Math.round(Math.abs(redway - greenway) / 6.0414 * 1000) <= 50) {
    redSpeed -= 9;
    brakeredtrainValue.innerHTML = "9";
  }
```

```
  if (redSpeed < 0) redSpeed = 0;
  if (greenSpeed < 0) greenSpeed = 0;
  //end second task code
  if (Math.abs(redway - greenway) > 3.02) {
    clearInterval(timer);
  }

}, 0.5);


function randomValue(min, max) {
  return Math.round(min - 0.5 + Math.random() * (max - min + 1));
}
```

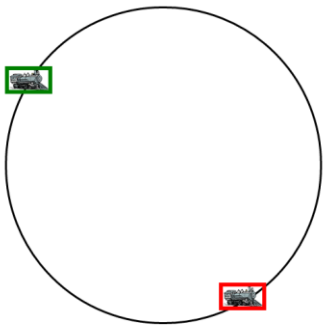
Result:



green speed: 0 | green to red distance: 452 | green train brake: 0
red speed: 14 | red to green distance: 548 | red train brake: 1

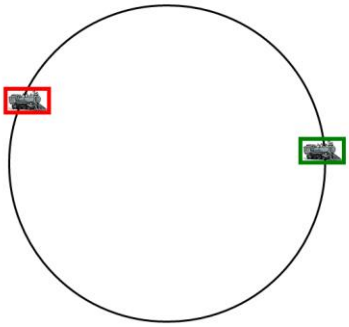
Mu_Speed = 14=>mu_s=0;
Mu_Dist = 548=>mu_d=0.6;
Mu_Brake = 1=>mu_b=0;
M=min(mu_s, mu_d)*mu_b => M=0*0=0

Z will always be equals 0 so Mu will always be 0.



green speed: 6 | green to red distance: 430 | green train brake: 0
red speed: 14 | red to green distance: 570 | red train brake: 1

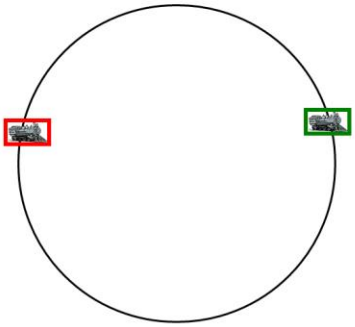
M=0



green speed: 15 green to red distance: 450 green train brake: 1

red speed: 6 red to green distance: 551 red train brake: 0

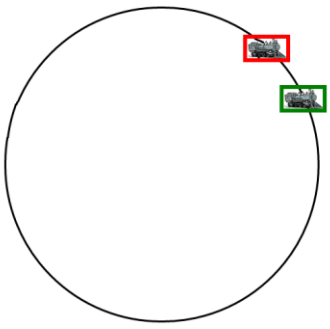
M=0



green speed: 0 green to red distance: 452 green train brake: 0

red speed: 14 red to green distance: 548 red train brake: 1

M=0



green speed: 7 green to red distance: 950 green train brake: 0

red speed: 12 red to green distance: 50 red train brake: 9

M=0