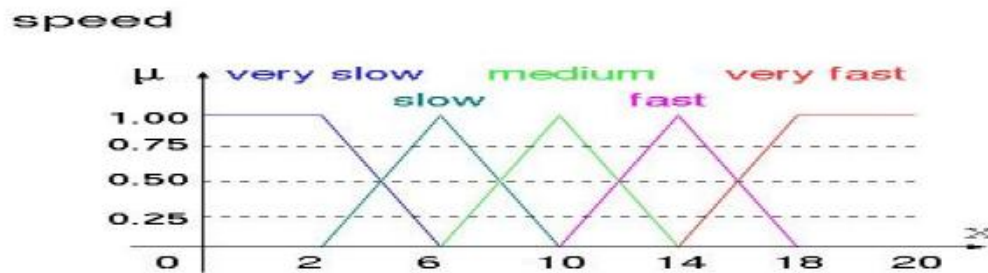


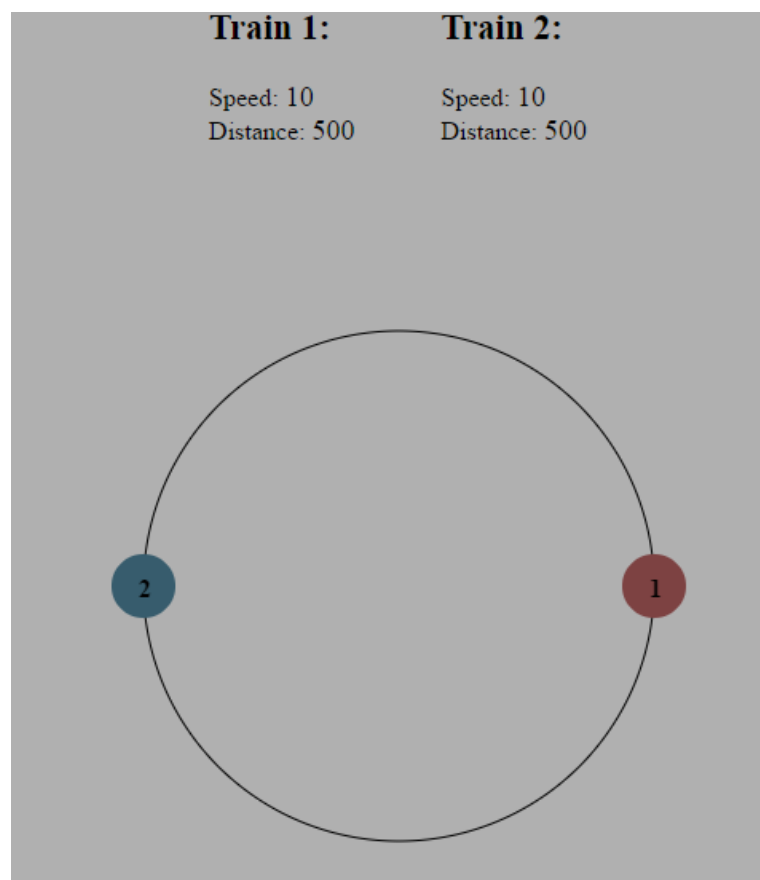
CCOD
Lab 1 09.09.2016
Akira Imada
Student – Aleksey Trotsiuk (AS - 36)

We have a metro which length is 1000px. We have 2 trains. They move in clockwise direction. Each train has start speed, which is 10. Max speed is 20, minimum speed is 0 (we do not allow to exceed or to be less than 0). On each iteration, speed of a train could be changed (from -1 to 1 including 0). The program stops when trains are collide



Below I attach 5 snapshots of trains:

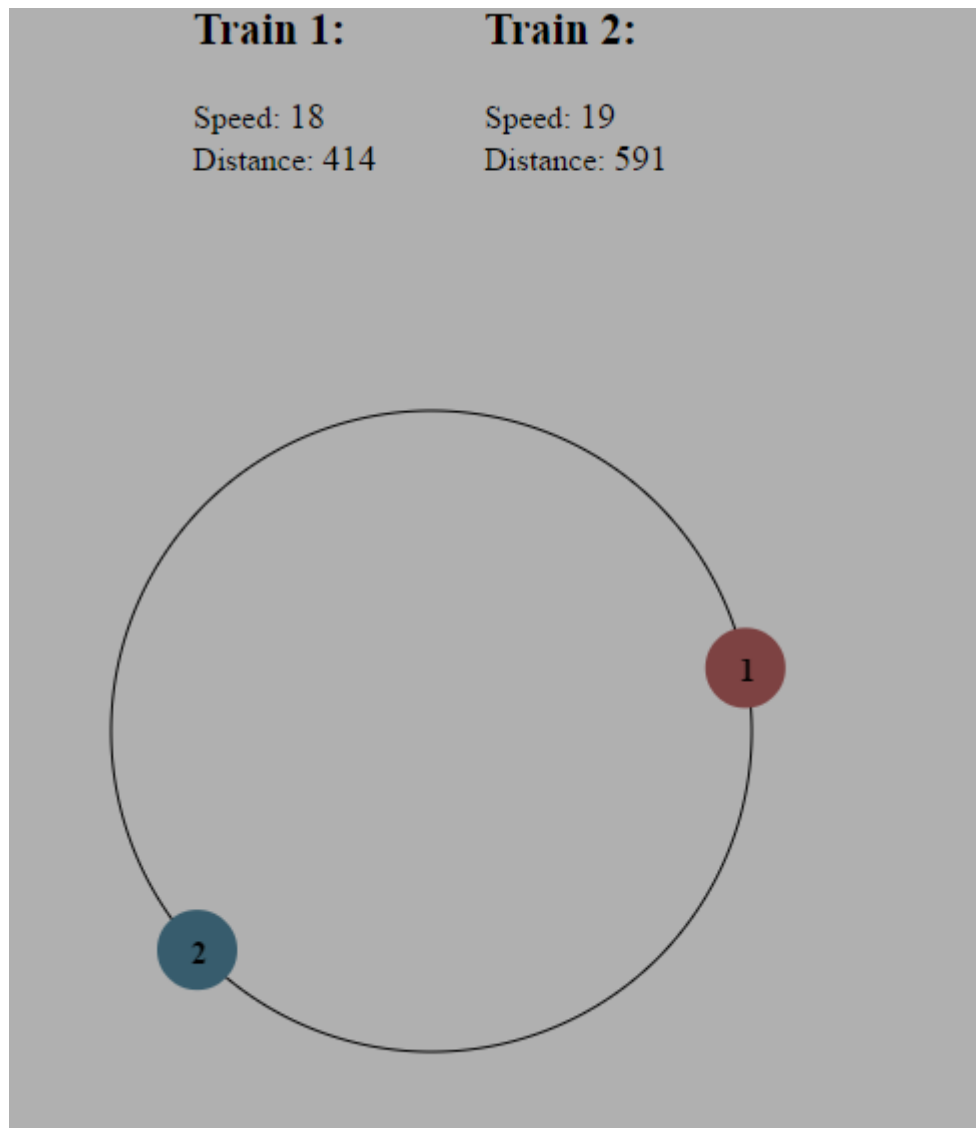
1.



$\mu_1 = \{0; 0; 1; 0; 0;\}$

$\mu_2 = \{0; 0; 1; 0; 0;\}$

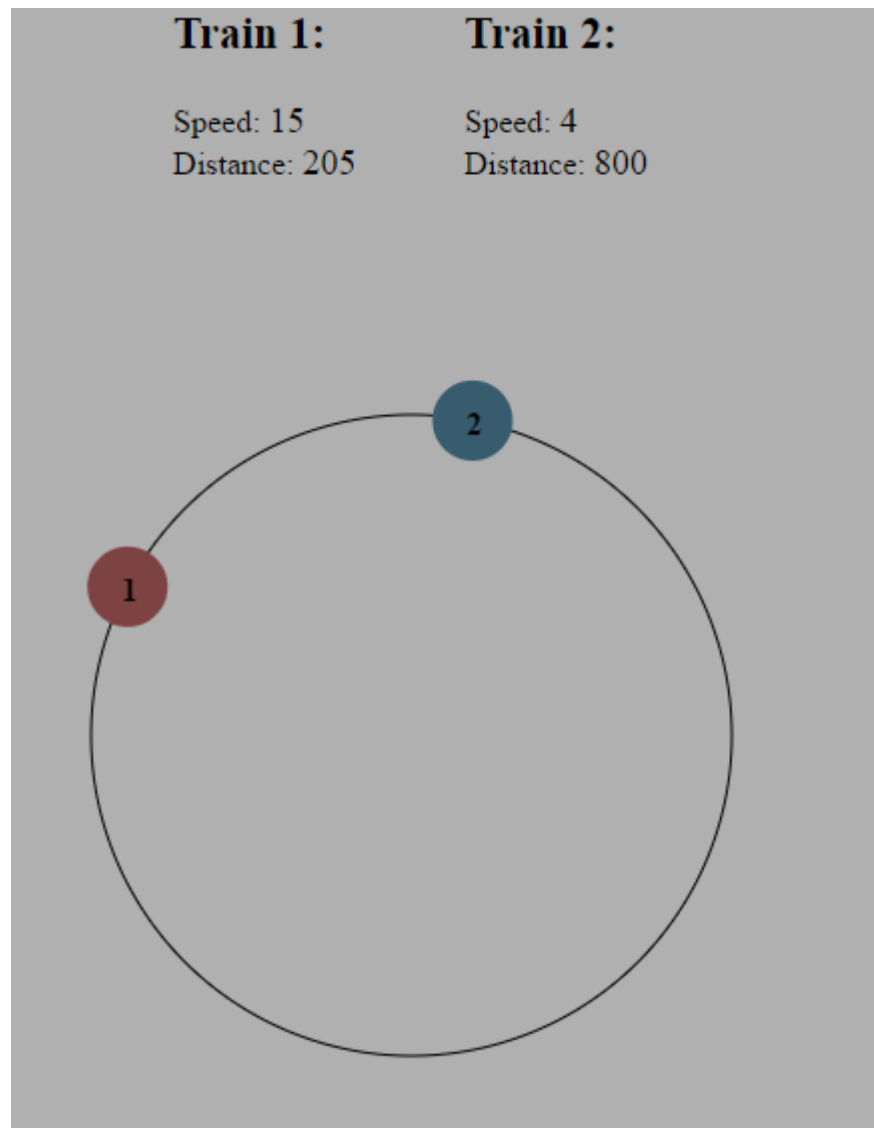
2.



$\mu_1=\{0; 0; 0; 0; 1;\}$

$\mu_2=\{0; 0; 0; 0; 1;\}$

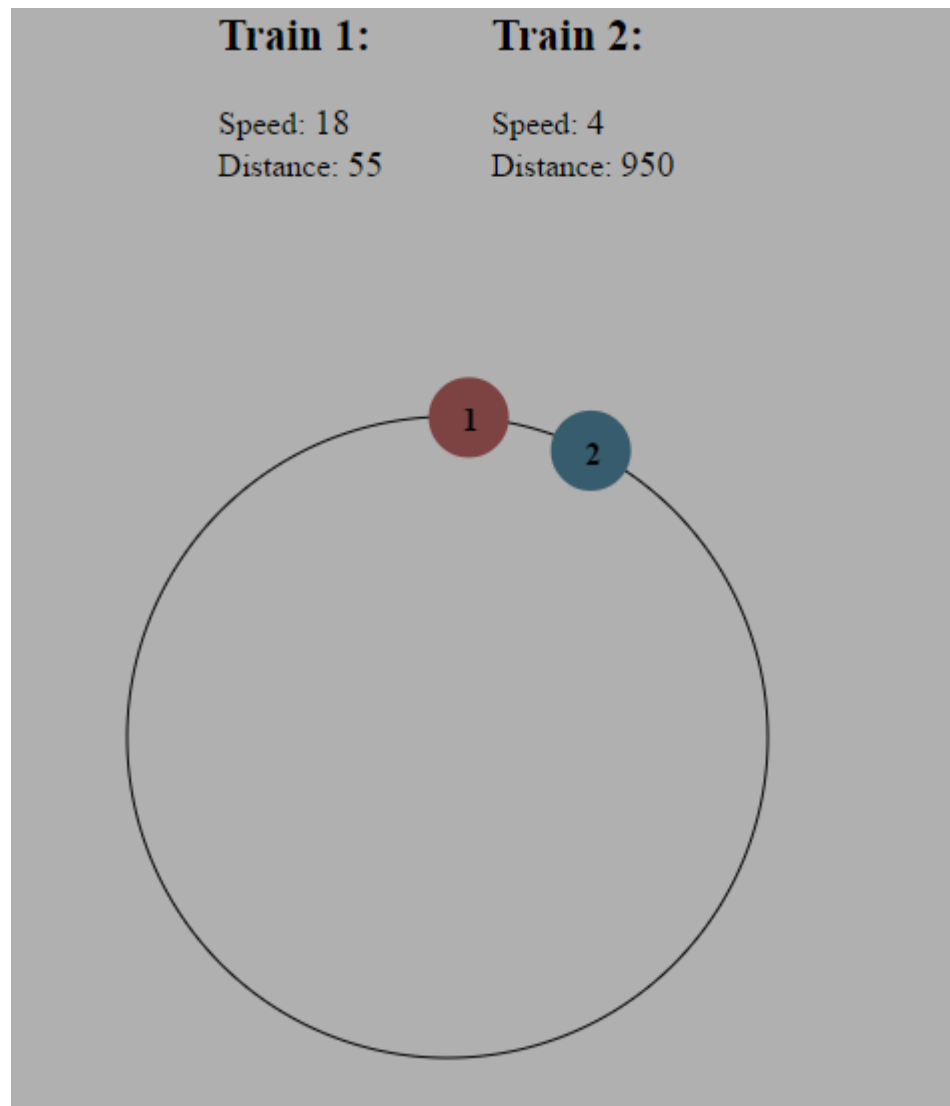
3.



$\mu_1 = \{0; 0; 0; 0.5; 0.5;\}$

$\mu_2 = \{0.5; 0.5; 0; 0; 0;\}$

4.



$\mu_1 = \{0; 0; 0; 0; 1;\}$

$\mu_2 = \{0.5; 0.5; 0; 0; 0;\}$

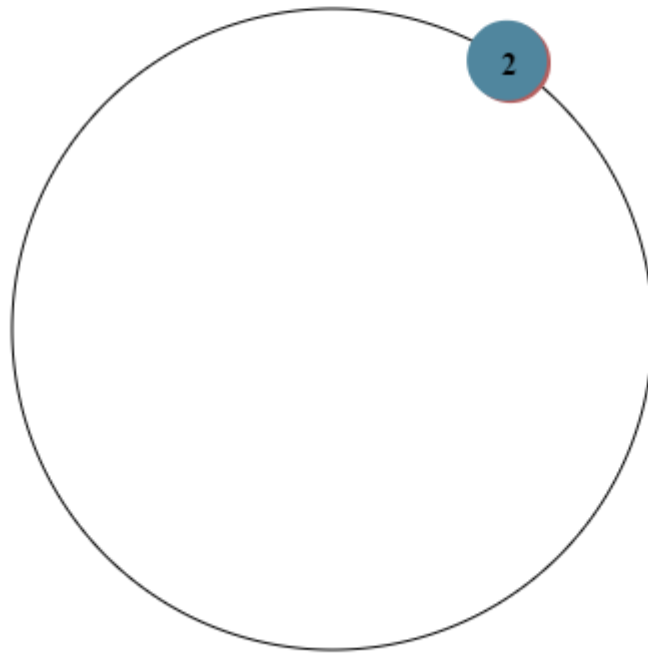
5.

Train 1:

Speed: 16
Distance: 6

Train 2:

Speed: 2
Distance: 999



$\mu_1 = \{0; 0; 0; 0.5; 0.5;\}$

$\mu_2 = \{1; 0; 0; 0; 0;\}$

]

SOURCE CODES:

Index.html

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Lab 1</title>
5          <link rel="stylesheet" type="text/css" href="styles.css">
6      </head>
7      <body>
8          <div class="main-content">
9              <div class="trains-info">
10                 <div class="train-content">
11                     <p class="title">Train 1:</p>
12                     <ul class="train-data">
13                         <li>Speed: <p id="train-speed1"></p></li>
14                         <li>Distance: <p id="train-distance1"></p></li>
15                     </ul>
16                 </div>
17                 <div class="train-content">
18                     <p class="title">Train 2:</p>
19                     <ul class="train-data">
20                         <li>Speed: <p id="train-speed2"></p></li>
21                         <li>Distance: <p id="train-distance2"></p></li>
22                     </ul>
23                 </div>
24             </div>
25             <canvas id="canvas" width="500" height="500"></canvas>
26         </div>
27         <script src = "script/index.js"></script>
28     </body>
29 </html>
30
```

Styles.css

```
1  .main-content {
2      position: relative;
3      text-align: center;
4  }
5
6  .main-content .trains-info .train-content {
7      display: inline-block;
8      text-align: left;
9      padding: 25px;
10 }
11
12 .main-content .trains-info .train-content .title {
13     margin: 0;
14     font-size: 22px;
15     font-weight: bold;
16 }
17
18 .main-content .trains-info .train-content .train-data {
19     list-style: none;
20     padding: 20px 0 0 0;
21     margin: 0;
22 }
23
24 .main-content .trains-info .train-content .train-data p {
25     display: inline;
26     margin: 0;
27     padding: 0;
28     font-size: 18px;
29 }
```

```
1  (function() {
2      "use strict";
3
4      const WIDTH = 1000;
5      const HEIGHT = 1000;
6      const X_CENTER = 250;
7      const Y_CENTER = 250;
8      const RADIUS = 160;
9      const TRAIN_RADIUS = 20;
10     const ARC_ANGLE = 2 * Math.PI;
11     const MOVE_UPDATE = 30;
12     const SPEED_UPDATE = 29;
13     const TRAIN_NUMBERS = 2;
14     const MAX_SPEED = 20;
15     const MIN_SPEED = 0;
16
17     let canvas = document.getElementById("canvas");
18     let context = canvas.getContext("2d");
19     let trains = [{
20         angle: 0,
21         speed: 10,
22         distance: 500,
23         color: "#b55f5f"
24     },
25     {
26         angle: 180,
27         distance: 500,
28         speed: 10,
29         color: "#50869e"
30     }
31     ];
32     let moveTrain;
33     let changeTrainSpeed;
34
35     init();
```



```

35
36     function init() {
37         updateWindow();
38
39         moveTrain = setInterval(move, MOVE_UPDATE);
40         changeTrainSpeed = setInterval(changeSpeed, SPEED_UPDATE);
41     }
42
43     function move() {
44         updateWindow();
45         drawTrains();
46         changeTrainsAngel();
47         if (isBreak()) {
48             stop();
49         }
50     }
51
52     function changeSpeed() {
53         for (let i = 0; i < trains.length; i++) {
54             trains[i].speed += getRandomInt(-1, 2);
55             if (trains[i].speed > MAX_SPEED) {
56                 trains[i].speed = MAX_SPEED;
57             }
58             if (trains[i].speed < MIN_SPEED) {
59                 trains[i].speed = MIN_SPEED
60             }
61             let trainSpeed = "train-speed" + (i + 1);
62             let trainDistance = "train-distance" + (i + 1);
63             let speed = document.getElementById(trainSpeed);
64             let distance = document.getElementById(trainDistance);
65             speed.innerHTML = trains[i].speed;
66             distance.innerHTML = trains[i].distance;
67         }
68     }

```

```

70     function stop() {
71         updateWindow();
72         drawTrains();
73         clearInterval(moveTrain);
74         clearInterval(changeTrainSpeed);
75     }
76
77     function drawTrains() {
78         for (let i = 0; i < trains.length; i++) {
79             let position = getTrainPosition(trains[i].angle);
80
81             context.beginPath();
82             context.arc(position.x, position.y, TRAIN_RADIUS, 0, ARC_ANGLE);
83             context.fillStyle = trains[i].color;
84             context.fill();
85             context.fillStyle = "black";
86             context.font = "bold 16px serif";
87             context.textBaseline = "top";
88             context.fillText(i + 1, position.x - 3, position.y - 7);
89         }
90     }
91
92     function getTrainPosition(angle) {
93         let y = RADIUS * Math.sin(toRadian(angle)) + Y_CENTER;
94         let x = RADIUS * Math.cos(toRadian(angle)) + X_CENTER;
95         return {
96             x: x,
97             y: y
98         };
99     }
100
101     function changeTrainsAngel() {
102         for (let i = 0; i < trains.length; i++) {
103             trains[i].angle += trains[i].speed / 5;
104         }

```

```

106
107     function toRadian(angle) {
108         return angle * Math.PI / 180;
109     }
110
111     function isBreak() {
112         trains[0].distance = Math.round(RADIUS * toRadian(trains[1].angle - trains[0].angle));
113         trains[1].distance = Math.round(1005 - trains[0].distance);
114         return trains[0].distance <= 0 || trains[1].distance <= 0;
115     }
116
117     function updateWindow() {
118         context.clearRect(0, 0, WIDTH, HEIGHT);
119         context.beginPath();
120         context.arc(X_CENTER, Y_CENTER, RADIUS, 0, ARC_ANGLE);
121         context.stroke();
122     }
123
124     function getRandomInt(min, max) {
125         return Math.floor(Math.random() * (max - min)) + min;
126     }
127
128 })();

```