PredatorのJavaScript,p5.jsプログラム

https://p5js.org/examples/examples/Simulate_Soft_Body.php
のプログラムを少しだけ変更しました。

index.thmlファイル

```html
<html>
<head>
  <meta charset="UTF-8">
<script src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.5.2/p5.js"></script>
 <script language="javascript" type="text/javascript" src="SoftBody.js"></script>
<style> body {padding: 0; margin: 0;} </style>
</head>

<body>
</body>
</html>
```

SoftBody.jsファイル

```javascript
// center point
var centerX = 0.0, centerY = 0.0;

var radius = 45, rotAngle = -90;

var accelX = 0.0, accelY = 0.0;
```

```
var deltaX = 0.0, deltaY = 0.0;

var springing = 0.0009, damping = 0.98;


//corner nodes

var nodes = 16;


//zero fill arrays

var nodeStartX = [];

var nodeStartY = [];

var nodeX = [];

var nodeY = [];

var angle = [];

var frequency = [];


// soft-body dynamics

var organicConstant = 1.0;


function setup() {

  createCanvas(1000, 500);


  //center shape in window

  centerX = width/2;

  centerY = height/2;


  //initialize arrays to 0

  for (var i=0; i<nodes; i++){

    nodeStartX[i] = 0;

    nodeStartY[i] = 0;

    nodeY[i] = 0;
```

```
    nodeY[i] = 0;

    angle[i] = 0;

  }


  // iniitalize frequencies for corner nodes

  for (var i=0; i<nodes; i++){

    frequency[i] = random(5, 12);

  }


  noStroke();

  frameRate(100);

}


function draw() {

  //fade background

  fill(256);

  rect(100,100,width-200, height-200,90);

    fill(255, 100, 10);

  ellipse(mouseX, mouseY, 10, 30);

  ellipse(mouseX, mouseY-20, 10, 10);

  ellipse(mouseX-2, mouseY+25, 5, 30);

  ellipse(mouseX+5, mouseY+25, 5, 30);

  ellipse(mouseX+7, mouseY-2, 3, 15);

  ellipse(mouseX-7, mouseY-10, 20, 3);

  drawShape();

  moveShape();

}


  function drawShape() {
```

```
//  calculate node  starting locations
for (var i=0; i<nodes; i++){
  nodeStartX[i] = centerX+cos(radians(rotAngle))*radius;
  nodeStartY[i] = centerY+sin(radians(rotAngle))*radius;
  rotAngle += 360.0/nodes;
}


// draw polygon
curveTightness(organicConstant);
fill(30,0,0);
beginShape();
for (var i=0; i<nodes; i++){
  curveVertex(nodeX[i], nodeY[i]);
}
for (var i=0; i<nodes-1; i++){
  curveVertex(nodeX[i], nodeY[i]);
}
endShape(CLOSE);
}

function moveShape() {
//move center point
deltaX = mouseX-centerX;
deltaY = mouseY-centerY;

// create springing effect
deltaX *= springing;
deltaY *= springing;
accelX += deltaX;
```

```
    accelY += deltaY;


    // move predator's center

    centerX += accelX;

    centerY += accelY;


    // slow down springing

    accelX *= damping;

    accelY *= damping;


    // change curve tightness

    organicConstant = 1-((abs(accelX)+abs(accelY))*0.1);


    //move nodes

    for (var i=0; i<nodes; i++){

      nodeX[i] = nodeStartX[i]+sin(radians(angle[i]))*(accelX*2);

      nodeY[i] = nodeStartY[i]+sin(radians(angle[i]))*(accelY*2);

      angle[i] += frequency[i];

    }

}
```