

TBAG Exercise 7

7.1 p5 dot product & cross product

Dotprodukt

$$a(0,0,1) \quad b(2,0,0)$$

$$\vec{a} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad b \begin{pmatrix} 2 \\ 0 \\ 0 \end{pmatrix}$$

$$\vec{a} \cdot \vec{b} = (2 \cdot 0) + (0 \cdot 0) + (1 \cdot 0)$$

$$\vec{a} \cdot \vec{b} = 0 + 0 + 0$$

$$\vec{a} \cdot \vec{b} = 0$$

Cross product

$$\vec{a} \times \vec{b} = (0 \cdot 0) - (1 \cdot 0) + (1 \cdot 2) - 0 - 2 \cdot 0$$

$$= 0 - 0 + 2 - 0 + 0$$

$$= 2$$

Task 7.2

The **dot product** in p5 is calculated with the function `dot()` and calculates the dot product of two Vectors.

Example from p5 reference:

```
let v1 = createVector(1, 2, 3);  
let v2 = createVector(2, 3, 4);  
print(v1.dot(v2)); // Prints "20"
```

The **cross product** in p5 is calculated with the function `cross()`. It calculates and returns a vector composed of the cross product between two vectors.

Example from p5 reference:

```
let v1 = createVector(1, 2, 3);  
let v2 = createVector(1, 2, 3);  
v1.cross(v2); // v's components are [0, 0, 0]
```

Application idea 1: With the help of the dot product one can get the angle between a light source and a surface to calculate how the light would be reflected from the surface.

Application idea 2: By calculating the dot product between two Vectors, one can calculate if the two are pointing in the same direction or if one of them is in the field of view of the other one. This is decided by the result of the dot product, which is negative if the two Vectors point in different directions (more than 90 degrees from each other), 0 if they are perpendicular to each other and positive if the angle between the two is less than 90 degrees, which would mean that they are pointing in a similar direction.

7.3 Programming Basic

<https://editor.p5js.org/Zani/sketches/jL3ABjd-3>

```
function Ball(x, y) {
  this.position = createVector(x,y);
  this.rad = 30;
  this.speed = createVector(random(2,6),random(2,6));
  this.col = color(255,255,255);
  this.display = function() {
    noStroke();
    fill(this.col);
    ellipse(this.position.x,this.position.y, this.rad, this.rad);
  };

  this.move = function() {
    // Update the position of the shape
    this.position.add(this.speed);
    // Test to see if the shape exceeds the boundaries of the screen
    // If it does, reverse its direction by multiplying by -1
    if(this.position.x<this.rad||this.position.x>width-this.rad){
      this.speed.x = -1*this.speed.x;
    }
    if(this.position.y<this.rad||this.position.y>height-this.rad){
      this.speed.y = -1*this.speed.y;
    }
  }
}
```

```

You, a few seconds ago | 1 author (You)
let position;
let speed;

let ball1;
let ball2;

function setup() {
  createCanvas(640, 360);
  frameRate(30);
  ellipseMode(RADIUS);

  ball1 = new Ball(width/2, height/2);
  ball2 = new Ball(width/4, height/4);
}

function draw() {
  background(102);

  ball1.display();
  ball1.move();
  ball2.display();
  ball2.move();
  ball1.collission(ball2);
  ball2.collission(ball1);

  this.collission = function(other) {
    // direction change if they bump
    let d = this.position.dist(other.position);
    if(d < this.rad + this.rad){
      this.speed.x = -1*this.speed.x;
      this.speed.y = -1*this.speed.y;
      //Color change if they bump
      this.col = color(random(0,255), random(0,255), random(0,255));
    }
  }

  this.changeSpeed = function(){
    this.speed.mult(-1);
  }
}

function mousePressed(){
  ball1.changeSpeed();
}

```

how to improve?

doing a lottery with more balls and a whole, where they can bounce out.

how to make it 3d?

<https://p5js.org/examples/form-3d-primitives.html>