

Gamedev Snake

Name	GameDev
Location	c007
Time	10.30-12.30
Date	14/11

P5 web editor

- new
 - <https://editor.p5js.org/WilantNackaerts/sketches/2sykWQBCu>
- old
 1. <https://editor.p5js.org/MathiasSpanhove/sketches/HyS0Uj8dQ>
 2. https://editor.p5js.org/MathiasSpanhove/sketches/rJv_Dilu7
 3. <https://editor.p5js.org/MathiasSpanhove/sketches/rkz5uoLOQ>
 4. <https://editor.p5js.org/MathiasSpanhove/sketches/ryYpdj8dQ>
 5. <https://editor.p5js.org/MathiasSpanhove/sketches/r14hGoUd7>
 6. <https://editor.p5js.org/MathiasSpanhove/sketches/H1O7Ak-Fm>
 7. <https://editor.p5js.org/WilantNackaerts/sketches/HkMAhKZY7>

Start

Studenten naar <https://editor.p5js.org/> hun een account laten aanmaken dit kan heel makkelijk door hun account te linken aan github wat alle studenten normaal zouden moeten hebben.

Uitleggen p5js:

P5 is een framework op de taal javascript(algo), het framework is speciaal ontwikkeld voor op een makkelijke manier een visuele voorstelling te creëren van een voorwerp, spel, etc.

Bij het openen van een nieuw p5 project is er direct een functie setup en een functie draw gedeclareerd. In setup staat createCanvas(400,400) deze functie gaat een canvas teken, dit is de oppervlakte van het spel.

In draw staat background(220) dit gaat er voor zorgen dat het canvas de witte achtergrond kleur heeft. Draw is ook de game loop.

auto-refresh bij iedereen uit zetten anders kan de browser soms crashen.

Vierkant teken op canvas:

samen met de studenten naar de wiki gaan vanop de homepage van p5 (<https://p5js.org/>) en dan doorklikken naar reference. In reference scrollen naar shape en dan naar de documentatie van rect() gaan en uitleggen dat ze met deze functie een vierkant kunnen teken

In draw een vierkant teken ongeveer op het midden van het scherm van ongeveer 20 pixels groot.

rect(100, 100, 20, 20); positie x-as, positie y-as, hoogte, breedte.

- setup()
 - noStroke() verwijdert rand rond vierkant
- draw()
 - fill(0) kleur vierkant aanpassen

Kijken of iedereen nog kan volgen en dit al heeft.

Snake maken en laten bewegen stap 1.

variabele aanmaken en uitleggen wa een variabele is.

- let snake

In setup object snake declareren

- setup()
 - snake = `{ headPosition: createVector(0, 0)}`

In draw basis beweging toevoegen:

- draw()
 - snake.headPosition.x += 0;
 - snake.headPosition.y += 0;

In rect de x en de y positie aanpassen.

- rect(snake.headPosition.x, snake.headPosition.y, 20, 20)

Het blokje op een makkelijke manier al eens laten bewegen en alles goed uitleggen wat net gebeurt is:

de loop van draw zou nu duidelijker moeten zijn
zorgen dat de x en y waarde duidelijk zijn in rect

kijken of iedereen mee is.

Snake laten bewegen stap 2 (pijltes)

Aan snake in setup een x en y direction toevoegen

- setup()
 - snake
 - xdir: 0
 - ydir: 0
 - setDirection(newXDirection, newYDirection)
 - this.xdir = newXDirection
 - this.ydir = newYDirection

Samen met studenten in de documentatie gaan kijken naar de functie keyPressed.

De functie keyPressed toevoegen en uitleggen wat we gaan doen eventueel aan de hand van whiteboard zodat de x en y positioning duidelijk is.

switch maken om te bewegen

- function keyPressed()
 - switch(keyCode)
 - case LEFT_ARROW:
 - snake.setDirection(-1, 0)
 - break;
 - case RIGHT_ARROW:
 - snake.setDirection(1, 0)
 - break;
 - case DOWN_ARROW:
 - snake.setDirection(0, 1);
 - break;
 - case UP_ARROW:
 - snake.setDirection(0, -1);
 - break;

uit draw snake.headPosition.x en y verwijderen en een updatePosition in setup naken

- setup()
 - snake
 - updatePosition()
 - this.headPosition.x += this.xdir;
 - this.headPosition.y += this.ydir

in rect de grote van het vierkant aanpassen naar 1, 1

alles scale naar met behulp van variable resolution 20 en uitleggen wat scale doet

- draw()
 - scale(resolution)

frameRate naar 5 zetten (variable) zodat het spel trager gaat.

- setup()
 - frameRate(speed)

Food maken en laten verschijnen

De variable hoogte en breedte toevoegen.

In setup:

- setup
 - w = floor(width / resolution);
 - h = floor(height / resolution);

Uitleggen wat floor doet.

De variable food toevoegen en initialiseren in setup.

aan food een positie toevoegen

```
▪ setup()
  ▪ food
    ▪ x = 0
    ▪ y = 0
    ▪ show()
      ▪ fill(255,0,0);
      ▪ rect(this.x, this.y, 1, 1);
    ▪ spawn()
      ▪ this.x = floor(random(w));
      ▪ this.y = floor(random(h));
  ▪ food.spawn();
▪ draw()
  ▪ food.show()
```

Uitleggen dat show functie properder is en dit ook voor de snake maken

```
▪ setup()
  ▪ snake
    ▪ show
      ▪ fill(0)
      ▪ rect(snake.headPosition.x, snake.headPosition.y, 1, 1);
▪ Draw()
  ▪ snake.show()
```

Snake eet food

Controleren of de snake op de zelfde positie is als het eten.

In draw if structuur de posities controleren en goed uitleggen wat je juist gaat controleren.

```
▪ draw()
  ▪ if(snake.eat(food))
    ▪ food.spawn();
▪ setup()
  ▪ snake
    ▪ eat(food)
      ▪ if(this.headPosition.x === food.x && this.headPosition.y === food.y)
      ▪ print("opgegeten")
      ▪ return true;
```

Om te beginnen gewoon een print("opgegeten") schrijven en dan een nieuwe positie maken.

Body aan de snake toevoegen

In setup aan snake een body toevoegen (lege array)

in update position een copie van het head in de body steken

```
▪ this.body.unshift(this.headPosition.copy());
  ▪ adds new item to the beginning of an array and returns the length of an array
▪ this.body.pop (einde) verwijderd het laatste item in de array
  ▪ laatste deel (tail) verwijderen
```

aan body food toevoegen in eat functie

```
▪ this.body.unshift(food);
```

Body van de snake op het canvas teken.

for loop na teken van snake.

```
▪ setup()
  ▪ snake
    ▪ show()
      ▪ for (let part of snake.body) {
        rect(part.x, part.y, 1, 1);
      }
```

Snake gaat dood

aan snake een dead object toevoegen

- setup()
 - snake
 - isDead()
 - if (this.headPosition.x > w - 1 || this.headPosition.x < 0 || this.headPosition.y > h - 1 || snake.headPosition.y < 0) {
 - return true;
 - }
 - for (let part of this.body) {
 - if (part.x === this.head.x && part.y === this.head.y) {
 - return true;
 - }
 - }

aan draw een check toevoegen om te kijken of de snake dood is

- draw()
 - if(snake.dead()){
 - background(255, 0, 0);
 - textSize(64 / resolution);
 - textAlign(CENTER, CENTER);
 - fill(0, 100, 255);
 - text('GAME OVER', w / 2, h / 2);
 - noLoop();