



Figure 1: Boek 7: sinus en cosinus

#	Omschrijving
25	Sinus en cosinus 1: grafiek
26	Sinus en cosinus 2: tekenen
27	Sinus en cosinus 3: planeten
28	Sinus en cosinus 4: schieten

Contents

Voorwoord	1
Sinus en cosinus 1: tekenen.	2
Sinus en cosinus 2: tekenen.	12
Sinus en cosinus 3: zonnestelsel	20
Sinus en cosinus 4: schieten	29

Voorwoord



Figure 1: Het logo van De Jonge Onderzoekers

Dit is het Processing boek van de Dojo. Processing is een programmeertaal. Dit boek leert je die programmeertaal.

Over dit boek

Dit boek heeft een CC-BY-NC-SA licentie.



Figure 2: De licentie van dit boek

(C) Dojo Groningen 2016-2018

Het is nog een beetje een slordig boek. Er zitten tiepvauten in en de opmaak is **niet altijd even mooi**.

Daarom staat dit boek op een GitHub. Om precies te zijn, op <https://github.com/richelbilderbeek/Dojo>. Hierdoor kan iedereen die dit boek te slordig vindt minder slordig maken.

Sinus en cosinus 1: tekenen.

In deze les gaan we een sinus en cosinus tekenen.

Sinus en cosinus 1: opdracht 1

Type deze code over:

```
float x = 0;

void setup()
{
  size(314, 200);
}

void draw()
{
  final float y = sin(x);
  point(x, y);
  x = x + 1;
}
```

Wat zie je?



314 is ongeveer honderd keer het getal pi

Sinus en cosinus 4: Eindopdracht

Nu maken we het spel af:

- Zorg dat de kogel in het begin niet zichtbaar is (tip: zet `y_kogel` op min honderd)
- Als de spatie ingedrukt wordt, zet dan `hoek_kogel` gelijk aan `hoek_kanon`

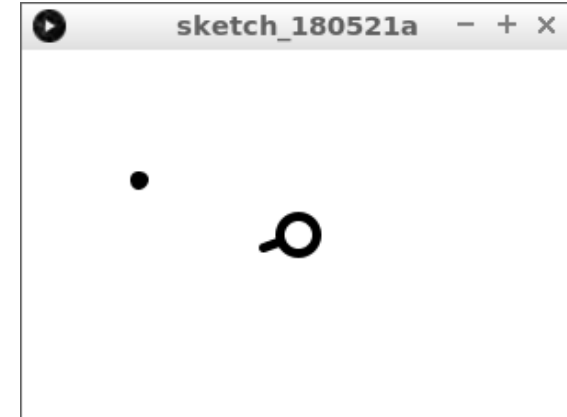


Figure 30: Sinus en cosinus 4: Eindopdracht

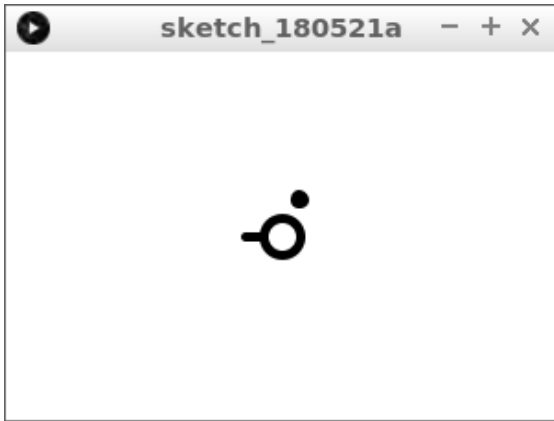


Figure 29: Sinus en cosinus 4: opdracht 6

Sinus en cosinus 4: oplossing 6

```
float hoek_kanon = 0;
float x_kogel = 0;
float y_kogel = 0;
float hoek_kogel = 0;

void setup()
{
  size(300, 200);
  strokeWeight(5);
}

void draw()
{
  background(255, 255, 255);
  final float x_midden = width / 2;
  final float y_midden = height / 2;
  final float x_kanon = x_midden + (cos(hoek_kanon) * 20);
  final float y_kanon = y_midden - (sin(hoek_kanon) * 20);
  x_kogel += cos(hoek_kogel);
  y_kogel -= sin(hoek_kogel);
  line(x_midden, y_midden, x_kanon, y_kanon);
  ellipse(x_midden, y_midden, 20, 20);
  ellipse(x_kogel, y_kogel, 5, 5);
  hoek_kanon += 0.1;
  if (keyPressed)
  {
    if (key == ' ')
    {
      x_kogel = x_kanon;
      y_kogel = y_kanon;
    }
  }
}
```

Sinus en cosinus 1: oplossing 1

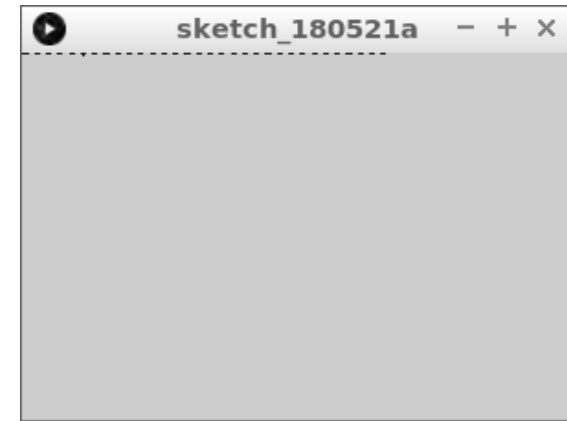


Figure 3: Sinus en cosinus 1: oplossing 1

Je ziet een soort stippellijn aan de bovenkant.

Sinus en cosinus 1: opdracht 2

- Verplaats de stippellijn omlaag. Doe dit door by y de halve hoogte op te tellen



Tip: het is slim om ronde haakjes om `height / 2` te zetten'

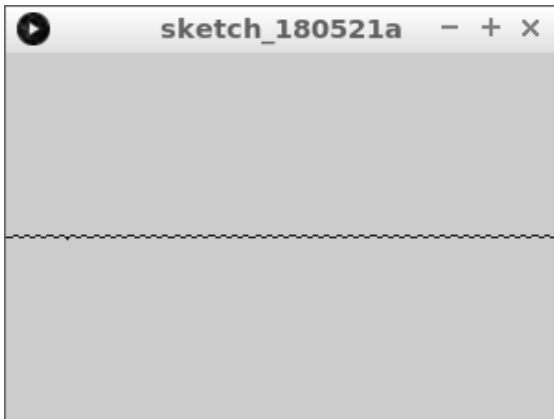


Figure 4: Sinus en cosinus 1: opdracht 2

Sinus en cosinus 4: oplossing 5

```
float hoek_kanon = 0;
float x_kogel = 0;
float y_kogel = 0;

void setup()
{
  size(300, 200);
  strokeWeight(5);
}

void draw()
{
  background(255, 255, 255);
  final float x_midden = width / 2;
  final float y_midden = height / 2;
  final float x_kanon = x_midden + (cos(hoek_kanon) * 20);
  final float y_kanon = y_midden - (sin(hoek_kanon) * 20);
  line(x_midden, y_midden, x_kanon, y_kanon);
  ellipse(x_midden, y_midden, 20, 20);
  ellipse(x_kogel, y_kogel, 5, 5);
  hoek_kanon += 0.1;
  if (keyPressed)
  {
    if (key == ' ')
    {
      x_kogel = x_kanon;
      y_kogel = y_kanon;
    }
  }
}
```

Sinus en cosinus 4: opdracht 6

- Maak een nieuwe variabele boven de `setup` functie met de naam `hoek_kogel`
- Geef `hoek_kogel` de beginwaarde nul
- Laat `x_kogel` steeds meer worden met de cosinus van `hoek_kogel`
- Laat `y_kogel` steeds *minder* worden met de sinus van `hoek_kogel`

- krijgt `y_kogel` de waarde van `y_kanon`
- blijft het kanon gewoon doordraaien

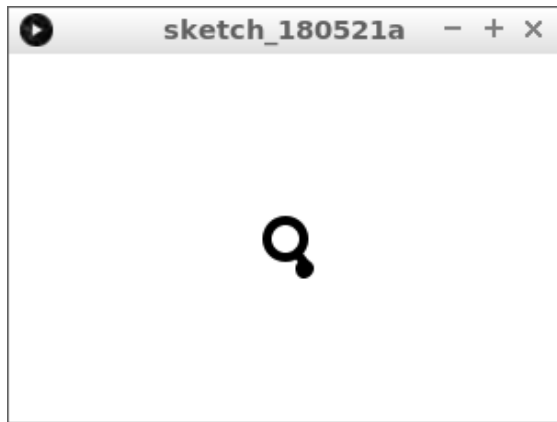


Figure 28: Sinus en cosinus 4: opdracht 5

Sinus en cosinus 1: oplossing 2

```
float x = 0;

void setup()
{
  size(314, 200);
}

void draw()
{
  final float y = sin(x) + (height / 2);
  point(x, y);
  x = x + 1;
}
```



Door de haakjes weet de computer de volgorde van een berekening



Een sinus heeft de vorm van een golf die slingert tussen -1 en 1

Sinus en cosinus 1: opdracht 3

- Vervang $\sin(x)$ door $(\sin(x) * 50)$

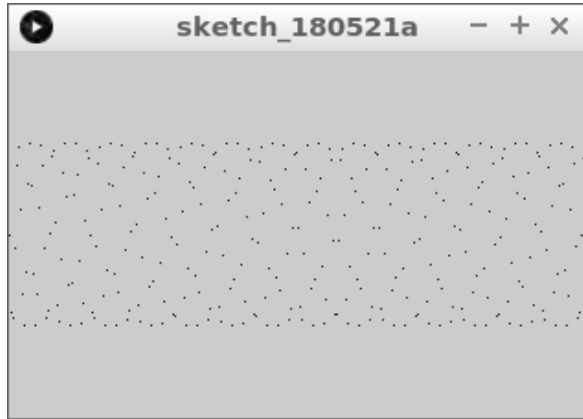


Figure 5: Sinus en cosinus 1: opdracht 3



Haakjes zijn belangrijk!

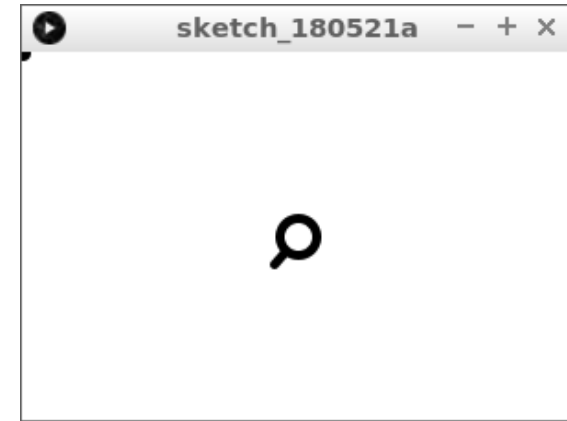


Figure 27: Sinus en cosinus 4: opdracht 4

Sinus en cosinus 4: oplossing 4

```
float hoek_kanon = 0;
float x_kogel = 0;
float y_kogel = 0;

void setup()
{
  size(300, 200);
  strokeWeight(5);
}

void draw()
{
  background(255, 255, 255);
  final float x_midden = width / 2;
  final float y_midden = height / 2;
  final float x_kanon = x_midden + (cos(hoek_kanon) * 20);
  final float y_kanon = y_midden - (sin(hoek_kanon) * 20);
  line(x_midden, y_midden, x_kanon, y_kanon);
  ellipse(x_midden, y_midden, 20, 20);
  ellipse(x_kogel, y_kogel, 5, 5);
  hoek_kanon += 0.1;
  if (keyPressed)
  {
    if (key == ' ')
    {
      hoek_kanon = 0;
    }
  }
}
```

Sinus en cosinus 4: opdracht 5

- Als je op de spatiebalk drukt, dan ...
 - krijgt x_{kogel} de waarde van x_{kanon}

Sinus en cosinus 4: oplossing 3

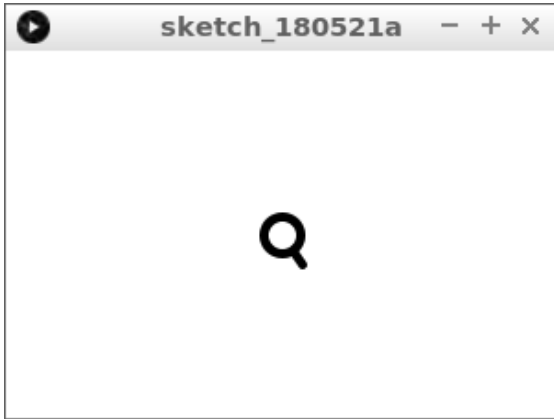


Figure 26: Sinus en cosinus 4: opdracht 3

```
float hoek_kanon = 0;

void setup()
{
  size(300, 200);
  strokeWeight(5);
}

void draw()
{
  background(255, 255, 255);
  final float x_midden = width / 2;
  final float y_midden = height / 2;
  final float x_kanon = x_midden + (cos(hoek_kanon) * 20);
  final float y_kanon = y_midden - (sin(hoek_kanon) * 20);
  line(x_midden, y_midden, x_kanon, y_kanon);
  ellipse(x_midden, y_midden, 20, 20);
  hoek_kanon += 0.1;
  if (keyPressed)
  {
    if (key == ' ')
    {
      hoek_kanon = 0;
    }
  }
}
```

Sinus en cosinus 4: opdracht 4

- Maak bovenaan de variabelen `x_kogel` en `y_kogel`
- Zet de beginwaarden van `x_kogel` en `y_kogel` op nul
- Teken een kogel met middelpunt (`x_kogel`, `y_kogel`) met een cirkel die 5 pixels breed en hoog is

Sinus en cosinus 1: oplossing 3

Je ziet nu een patroon.

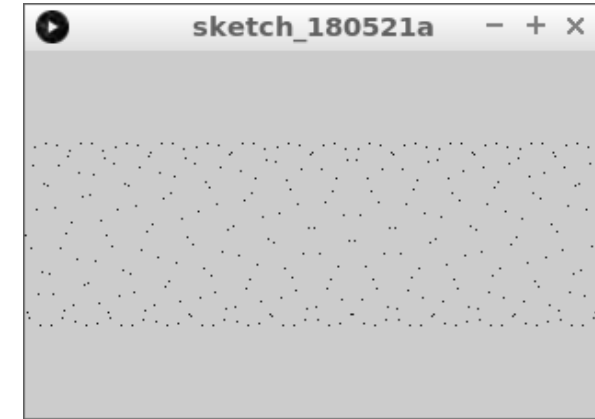


Figure 6: Sinus en cosinus 3: opdracht 2

```
float x = 0;

void setup()
{
  size(314, 200);
}

void draw()
{
  final float y = (sin(x) * 50) + (height / 2);
  point(x, y);
  x = x + 1;
}
```



Het is een rommel, omdat we ongeveer 100 sinussen door elkaar tekenen



* 50 zorgt ervoor dat de sinus vijftig keer vergroot wordt

Sinus en cosinus 1: opdracht 4

- Vervang `sin(x)` door `sin(x / 100)`

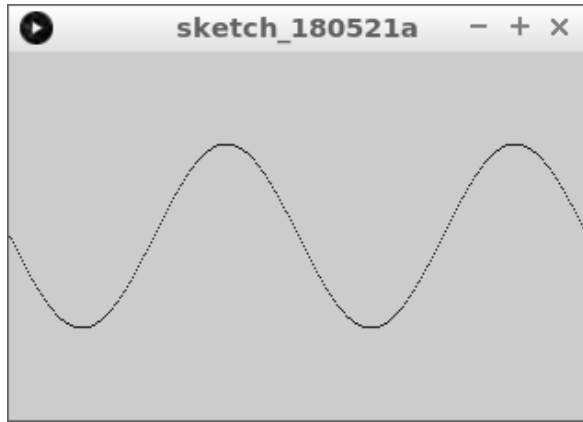


Figure 7: Sinus en cosinus 1: opdracht 4

Sinus en cosinus 1: oplossing 4

Je ziet nu een golf: een sinus!.

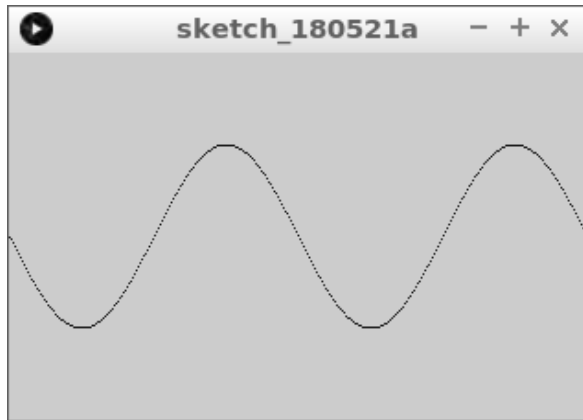


Figure 8: Sinus en cosinus 1: opdracht 4

```
float x = 0;

void setup()
{
  size(314, 200);
}

void draw()
{
  final float y = (sin(x / 25) * 50) + (height / 2);
  point(x, y);
}
```

Sinus en cosinus 4: opdracht 3

- Zorg ervoor dat `hoek_kanon` nul wordt, als je op de spatiebalk drukt

Tip:

```
if (keyPressed)
{
  if (key == ' ')
  {
    //Zet hoek_kanon op nul
  }
}
```

Sinus en cosinus 4: oplossing 2

```
float hoek_kanon = 0;

void setup()
{
  size(300, 200);
  strokeWeight(5);
}

void draw()
{
  background(255, 255, 255);
  final float x_midden = width / 2;
  final float y_midden = height / 2;
  final float x_kanon = x_midden + (cos(hoek_kanon) * 20);
  final float y_kanon = y_midden - (sin(hoek_kanon) * 20);
  line(x_midden, y_midden, x_kanon, y_kanon);
  ellipse(x_midden, y_midden, 20, 20);
  hoek_kanon += 0.1;
}
```

```
x = x + 1;
}
```

Sinus en cosinus 1: opdracht 5

- Vervang $\sin(x)$ door $\cos(x)$

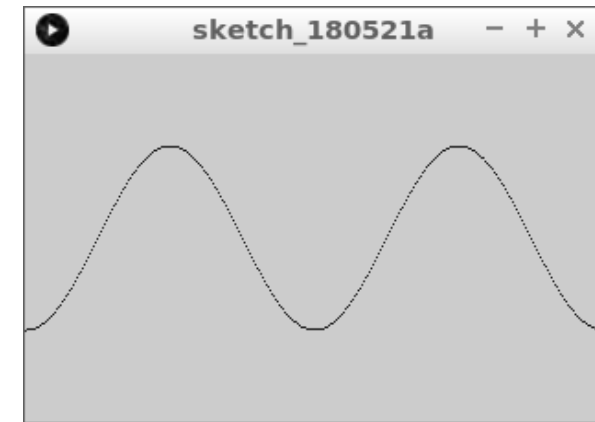


Figure 9: Sinus en cosinus 1: opdracht 5

Sinus en cosinus 1: oplossing 5

Je ziet nu een golf: een sinus!

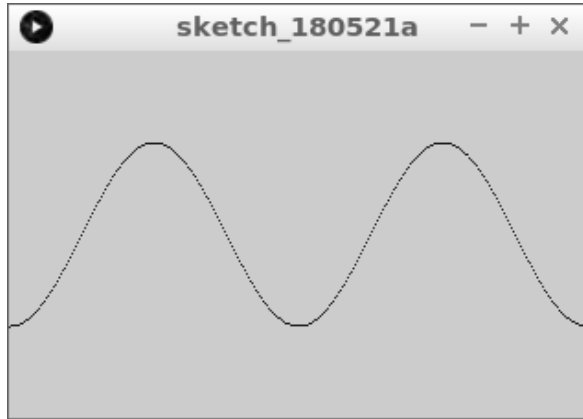


Figure 10: Sinus en cosinus 1: opdracht 5

```
float x = 0;

void setup()
{
  size(314, 200);
}

void draw()
{
  final float y = (sin(x / 25) * 50) + (height / 2);
  point(x, y);
  x = x + 1;
}
```



$x / 25$ zorgt ervoor dat we 25x rustiger door de sinus gaan

Sinus en cosinus 1: Eindopdracht

- Teken zowel de sinus als cosinus
- Zorg dat elke golf een keer op en neer gaat
- Teken de cosinus rood, de sinus blauw

Sinus en cosinus 4: opdracht 2

- Laat het kanon draaien door `hoek_kanon` steeds 0.1 meer te maken

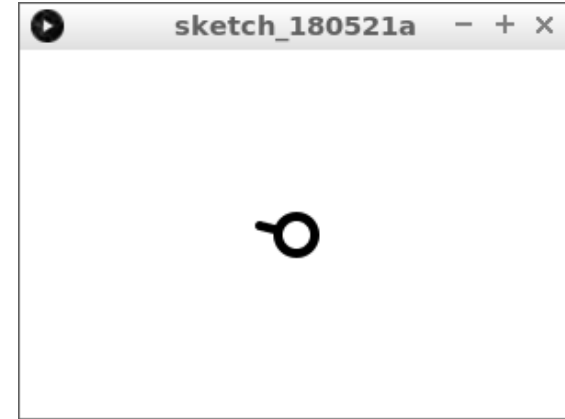


Figure 25: Sinus en cosinus 4: opdracht 2

Sinus en cosinus 4: oplossing 1

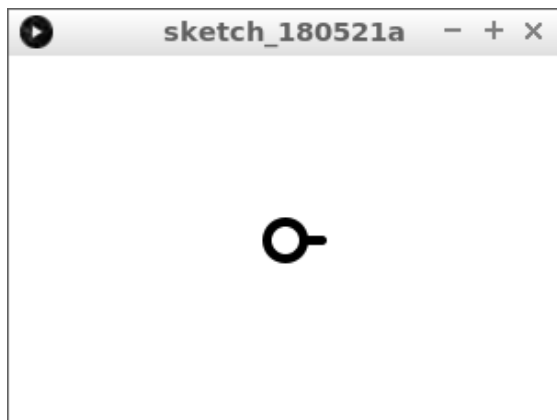


Figure 24: Sinus en cosinus 4: oplossing 1

Je ziet een cirkel in het midden, met een kanon

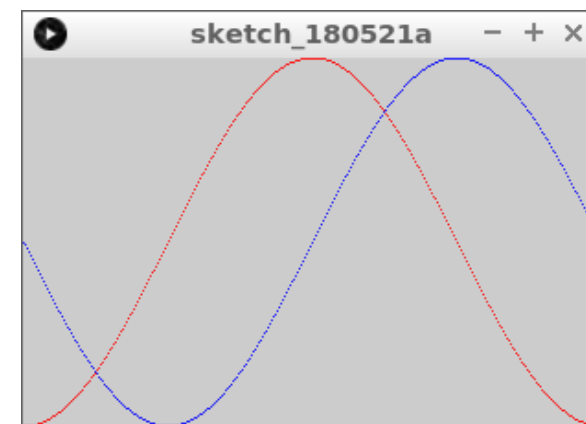


Figure 11: Sinus en cosinus 1: Eindopdracht

Sinus en cosinus 2: tekenen.

In deze les gaan we een sinus en cosinus gebruiken om een maan om een planeet te laten slingeren.

Sinus en cosinus 2: opdracht 1

Type deze code over:

```
float hoek = 0;

void setup()
{
  size(300, 200);
}

void draw()
{
  final float x = cos(hoek) * 100;
  final float y = sin(hoek) * 100;
  point(x, y);
  hoek = hoek + 0.01;
}
```

Wat zie je?

Sinus en cosinus 4: opdracht 1

Type deze code over:

```
float hoek_kanon = 0;

void setup()
{
  size(300, 200);
  strokeWeight(5);
}

void draw()
{
  background(255, 255, 255);
  final float x_midden = width / 2;
  final float y_midden = height / 2;
  final float x_kanon = x_midden + (cos(hoek_kanon) * 20);
  final float y_kanon = y_midden - (sin(hoek_kanon) * 20);
  line(x_midden, y_midden, x_kanon, y_kanon);
  ellipse(x_midden, y_midden, 20, 20);
}
```

Wat zie je?

Sinus en cosinus 4: schieten

In deze les gaan we een sinus en cosinus gebruiken om een kogel te schieten

Sinus en cosinus 2: oplossing 1

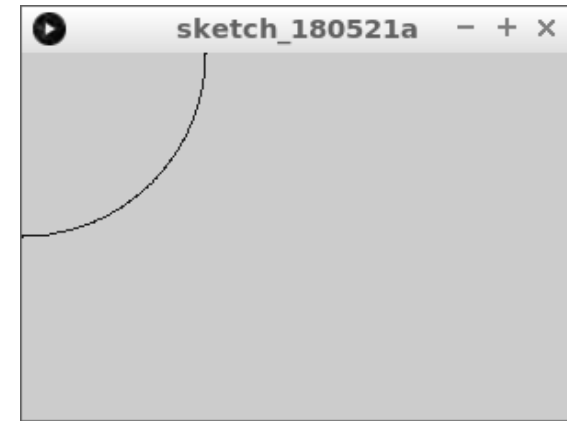


Figure 12: Sinus en cosinus 2: oplossing 1

Je ziet een kwart cirkel in de linkerbovenhoek.

Sinus en cosinus 2: opdracht 2

- Verplaats de cirkel naar het midden'
- Doe dit door by y de halve hoogte op te tellen
- Doe dit door by x de halve breedte op te tellen



Tip: het is slim om ronde haakjes om `height / 2` te zetten'

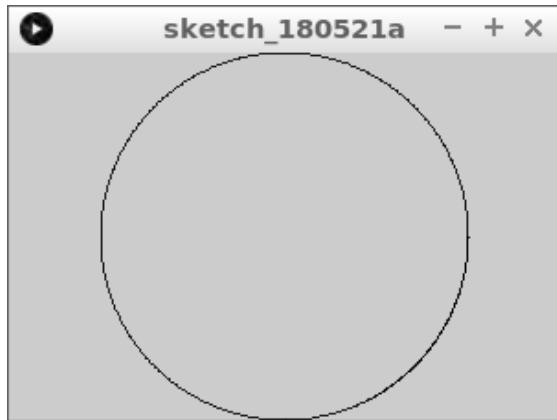


Figure 13: Sinus en cosinus 2: opdracht 2

Sinus en cosinus 3: Eindopdracht

Laat de maan om de aarde cirkelen!

- Maak een nieuwe variabele `hoek_maan`, zet deze helemaal bovenaan
- Maak `hoek_maan` meer aan het eind van de `draw` functie. Gebruik een waarde zoals 0.1
- Maak `x_maan` nu `x_aarde` plus tien keer de sinus van `hoek_aarde`
- Hetzelfde voor `y_maan`

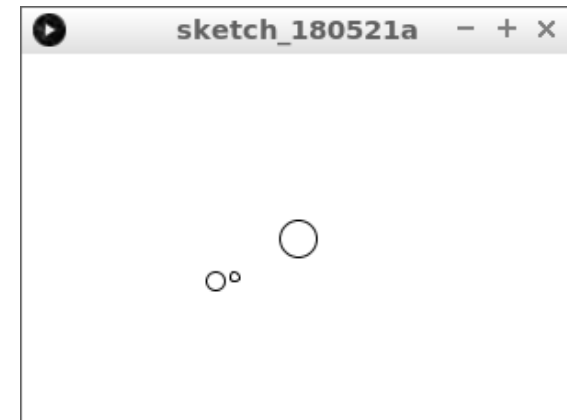


Figure 23: Sinus en cosinus 3: Eindopdracht

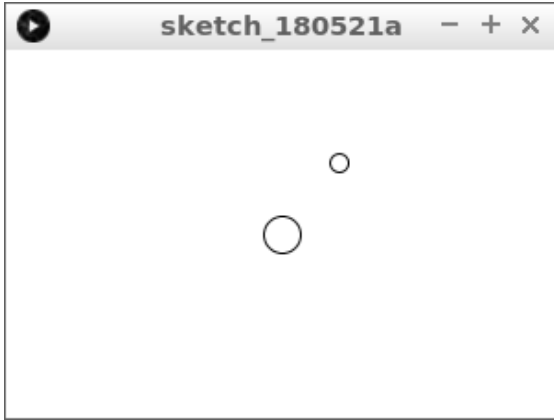


Figure 21: Sinus en cosinus 3: opdracht 4

Sinus en cosinus 3: opdracht 5

- Maak twee nieuwe variabelen: `x_maan` en `y_maan` (binnen de `draw` functie)
- De waarde van `x_maan` is de waarde van `x_aarde` plus 10
- De waarde van `y_maan` is de waarde van `y_aarde` plus 10
- Teken de maan als een cirkel met een breedte en hoogte van 5 pixels

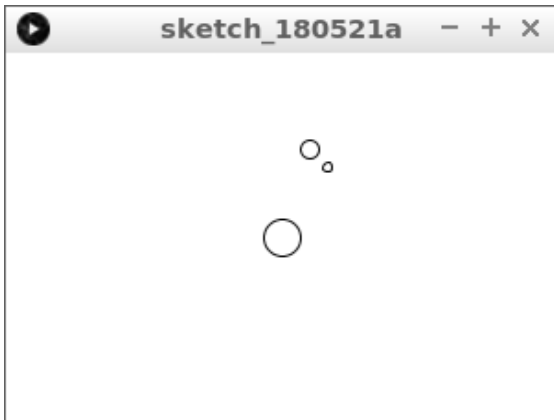


Figure 22: Sinus en cosinus 3: opdracht 5

Sinus en cosinus 2: oplossing 2

```
float hoek = 0;

void setup()
{
  size(300, 200);
}

void draw()
{
  final float x = cos(hoek) * 100 + (width / 2);
  final float y = sin(hoek) * 100 + (height / 2);
  point(x, y);
  hoek = hoek + 0.01;
}
```



Door de haakjes weet de computer de volgorde van een berekening



Een sinus en cosinus samen kunnen een cirkel tekenen

Sinus en cosinus 2: opdracht 3

- Teken een cirkel in plaats van een punt. De cirkel moet 20 pixels breed zijn, en 20 pixels hoog

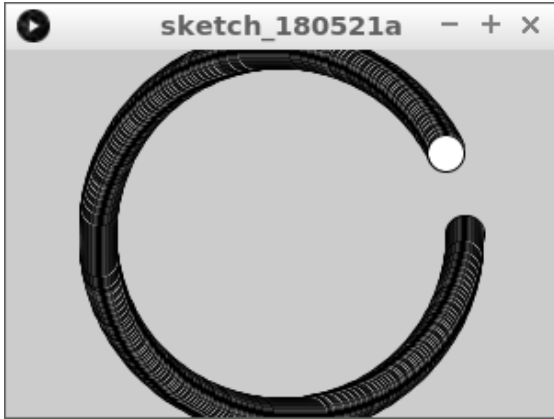


Figure 14: Sinus en cosinus 2: opdracht 3

Sinus en cosinus 3: oplossing 3

Je ziet nu de Aarde heen en weer gaan

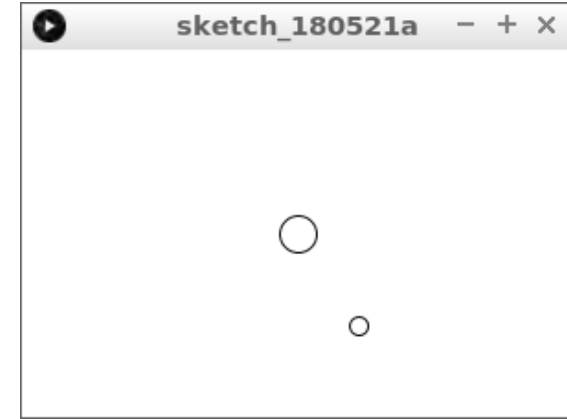


Figure 20: Sinus en cosinus 3: opdracht 2

```
float hoek_aarde = 0;

void setup()
{
  size(300, 200);
}

void draw()
{
  background(255, 255, 255);
  final float x_zon = width / 2;
  final float y_zon = height / 2;
  ellipse(x_zon, y_zon, 20, 20);
  final float x_aarde = x_zon + (50 * sin(hoek_aarde));
  final float y_aarde = y_zon + 50;
  ellipse(x_aarde, y_aarde, 10, 10);
  hoek_aarde += 0.01;
}
```

Sinus en cosinus 3: opdracht 4

- Zorg dat de Aarde nu om de zon heen gaat
- Maak `y_aarde` nu `y_zon` plus vijftig keer de cosinus van `hoek_aarde`

Sinus en cosinus 3: opdracht 3

- Maak een nieuwe variabele `hoek_aarde`, zet deze helemaal bovenaan
- Maak `hoek_aarde` meer aan het eind van de `draw` functie. Gebruik een waarde zoals 0.01
- Maak `x_aarde` nu `x_zon` plus vijftig keer de sinus van `hoek_aarde`

Wat zie je?

Sinus en cosinus 2: oplossing 3

Je ziet nu een patroon.

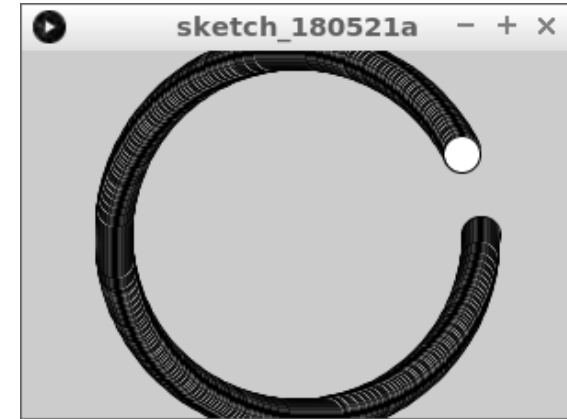


Figure 15: Sinus en cosinus 3: opdracht 2

```
float hoek = 0;

void setup()
{
  size(300, 200);
}

void draw()
{
  final float x = cos(hoek) * 100 + (width / 2);
  final float y = sin(hoek) * 100 + (height / 2);
  ellipse(x, y, 20, 20);
  hoek = hoek + 0.01;
}
```

Sinus en cosinus 2: opdracht 4

- Zorg dat de bal geen streep meer achterlaat
- Gebruik hiervoor een witte achtergrond



Tip: gebruik background



Figure 16: Sinus en cosinus 2: opdracht 4

Sinus en cosinus 2: Eindopdracht

- Teken ook een cirkel in het midden, met een breedte en hoogte van 50 pixels
- Laat de bewegende cirkel tegen de klok ingaan
- Zorg dat de bewegende cirkel niet meer half het scherm uit gaat. Doe dit, door de afstand met het midden kleiner te maken

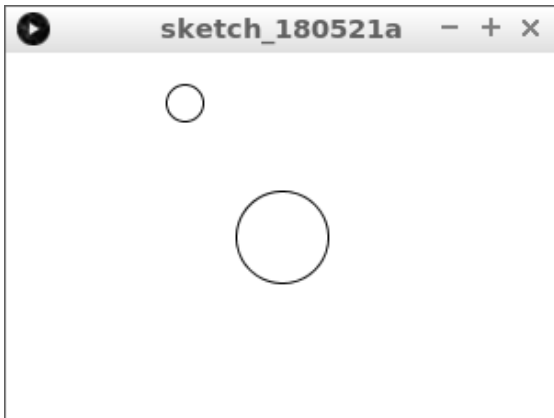


Figure 17: Sinus en cosinus 2: Eindopdracht

Sinus en cosinus 3: oplossing 2

```
void setup()
{
  size(300, 200);
}

void draw()
{
  background(255, 255, 255);
  final float x_zon = width / 2;
  final float y_zon = height / 2;
  ellipse(x_zon, y_zon, 20, 20);
  final float x_aarde = x_zon + 50;
  final float y_aarde = y_zon + 50;
  ellipse(x_aarde, y_aarde, 10, 10);
}
```

Sinus en cosinus 3: opdracht 2

- Maak twee nieuwe variabelen: `x_aarde` en `y_aarde` (binnen de `draw` functie)
- De waarde van `x_aarde` is de waarde van `x_zon` plus 50
- De waarde van `y_aarde` is de waarde van `y_zon` plus 50
- Teken de Aarde als een cirkel met een breedte en hoogte van 10 pixels

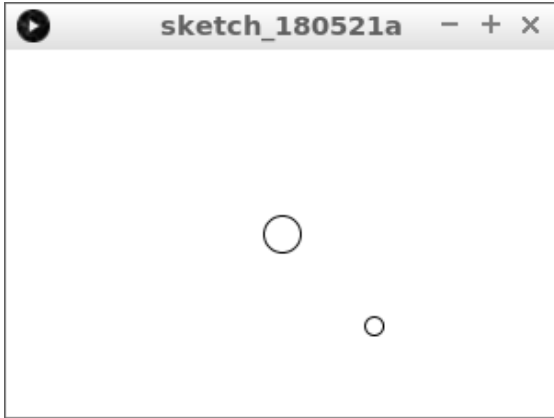


Figure 19: Sinus en cosinus 3: opdracht 2

Sinus en cosinus 3: zonnestelsel

In deze les gaan we een sinus en cosinus gebruiken om een zonnestelsel te maken

Sinus en cosinus 3: opdracht 1

Type deze code over:

```
void setup()
{
  size(300, 200);
}

void draw()
{
  background(255, 255, 255);
  final float x_zon = width / 2;
  final float y_zon = height / 2;
  ellipse(x_zon, y_zon, 20, 20);
}
```

Wat zie je?

Sinus en cosinus 3: oplossing 1



Figure 18: Sinus en cosinus 3: oplossing 1

Je ziet een cirkel in het midden: de zon!