



Basis Figuren

Inleiding

Met de hulp van de schildpad kunnen verschillende figuren getekend worden. Van zeer eenvoudig tot zeer complex. Vaak kunnen de figuren op verschillende manieren getekend worden. De ene manier geeft een eenvoudigere broncode dan de andere.

In deze les gaan we kijken naar basis figuren die gebruikt kunnen worden om andere meer complexe figuren toch redelijk eenvoudig te maken.

Als je je afvraagt waarom deze figuurtjes tekenen? Het antwoord is dat het in principe niet om de figuren gaat maar om het aanleren van bepaalde denkpatronen en het gestructureerd kunnen oplossen van een probleem. Er worden tekeningen gebruikt omdat dit visueel is en het beter duidelijk maakt hoe programma's en computers werken.

Eigenschappen van de basis figuren

De basis figuren zijn een aantal wiskundige figuren die je al in de wiskunde lessen hebt gekregen.

De eigenschappen voor deze figuren zijn:

- Alle figuren hebben alleen zijden met dezelfde lengten.
- Alle figuren (*behalve eentje*) hebben de zelfde hoeken.
- Alle figuren kunnen eenvoudig met de iteratie structuur gemaakt worden.
- De schildpad begint en eindigt altijd op dezelfde plaats.
- De begin positie is links onder in de figuur. Daar we er vanuit kunnen gaan dat bij elke opdracht de schildpad in het midden op het scherm staat en naar boven wijst. Tevens draait de schildpad normaal altijd rechtsom.

Voor elk van de basis figuren gaan we de volgende stappen uitvoeren:

- Maak de analyse.
- Maak de PSD.
- Schrijf de broncode.
- Test de broncode in Small Basic.

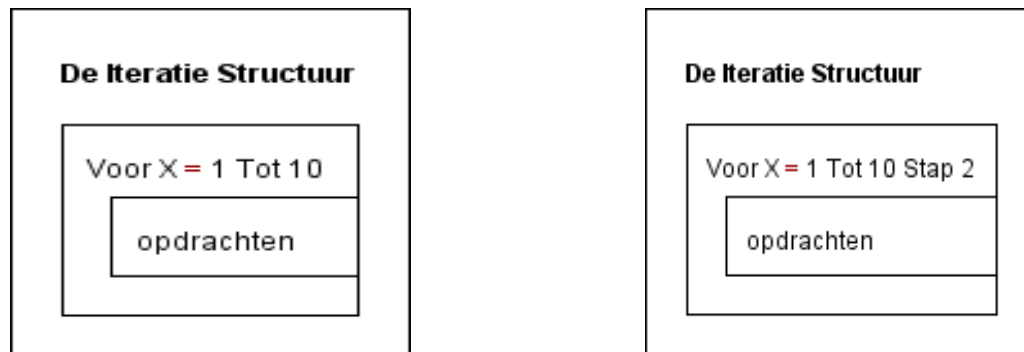


Basis Figuren

Iteratie voor basis figuren

Voor het maken van de basis figuren wordt de iteratie structuur gebruikt. De reden hiervoor is dat er zo veel efficiëntere broncode wordt geschreven.

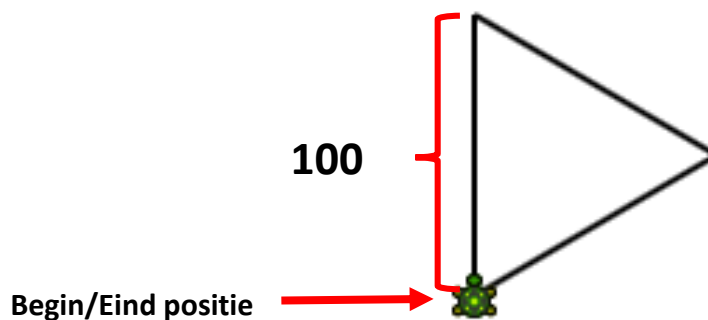
Er wordt gebruik gemaakt van de VOOR ... TOT ... STAP structuur daar de voorwaarden voor het herhalen altijd bekend zijn bij het maken van de basis figuren.



De stap opdracht wordt gebruikt om regelmatige (zelfde grootte) stappen te kunnen overslaan.

Denk eraan de STAP kan zowel een positief als een negatieve waarde hebben.

De gelijkzijdige driehoek



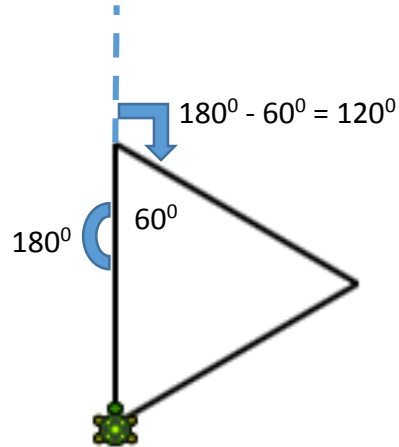
Bij een gelijkzijdige driehoek zijn de maten van de zijden even groot en ook de hoeken van de driehoek zijn even groot.

De som van de hoeken van een driehoek is 180° graden. Dus elke hoek is $180/3 = 60^{\circ}$ graden.



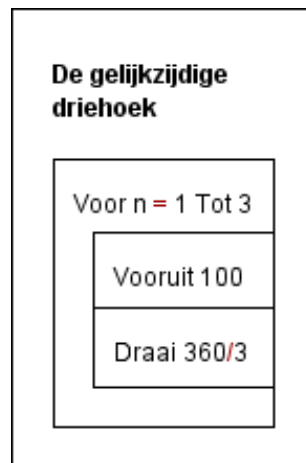
Basis Figuren

De draaihoek die de schildpad moet maken is $180 - 60 = 120^0$ graden.



De hoek van een lijn is namelijk 180^0 graden, de helft van een cirkel.

De PSD voor een gelijkzijdige driehoek wordt dus:



De bron code is als volgt:

```
1 For n = 1 To 3
2   Turtle.Move(100)
3   Turtle.Turn(360/3)
4 EndFor
```



Basis Figuren

Hierdoor draait de schildpad 90^0 graden rechtsom (*met de klok mee*)

De structuuropdrachten voor iteratie en selectie blijven hetzelfde.

Vraag: Wat is de opdracht zijn om de schildpad achteruit te laten bewegen?

Dit kan op twee manieren:

1. ACHTERUIT 100
2. VOORUIT -100

Vraag: Wat is de opdracht om de schildpad 90^0 links om te laten draai?

Antwoord op deze vraag kan op twee manieren:

1. DRAAI -90
2. DRAAI 270

PSD voor VIERKANT

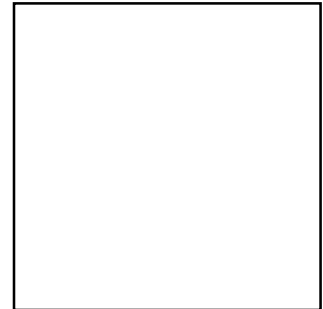
Voor het kunnen maken van een PSD moet er eerst geanalyseerd worden. Bij de analyse moet op een aantal zaken gelet worden. Naar mate we verder in de stof komen zullen het aantal punten waarop gelet moet worden toe nemen.

Stel we willen de schildpad een vierkant laten tekenen.

We moeten ons dan een aantal zaken afvragen.

Bijvoorbeeld:

- Waar beginnen we met tekenen?
- Hoe lang zijn de zijden van de figuur?
- Hoe groot zijn de hoeken van de figuur?
- Hoe groot zijn de **draaihoeken** die de schildpad moet maken.
 - Elke keer als de schildpad aan het einde van een lijn komt moet deze draaien om de volgende lijn te kunnen tekenen. De hoek die de schildpad moet maken noemt men de draaihoek. Deze zijn niet altijd hetzelfde als de hoeken van de figuur.

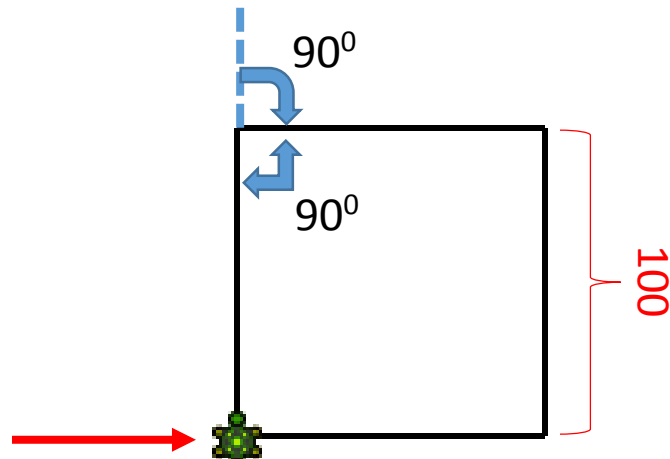




Basis Figuren

Het vierkant (gelijkzijdige rechthoek)

Bij een vierkant geldt dat alle zijden even lang zijn en alle hoeken even groot (90^0)



Begin positie

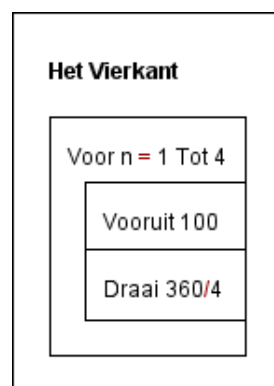
In de afbeelding hierboven zien we dat de beginpositie van de schildpad links onder in de figuur is.

De lengte van de zijden van de figuur zijn allemaal even groot en in dit geval wordt er aan gegeven dat deze 100 stappen is.

De figuur kent vier hoeken elk met een grootte van 90^0 graden.

Voor de draaihoeken die de schildpad moet maken geldt tevens dat deze 90^0 graden zijn.

De PSD voor het vierkant wordt dus:



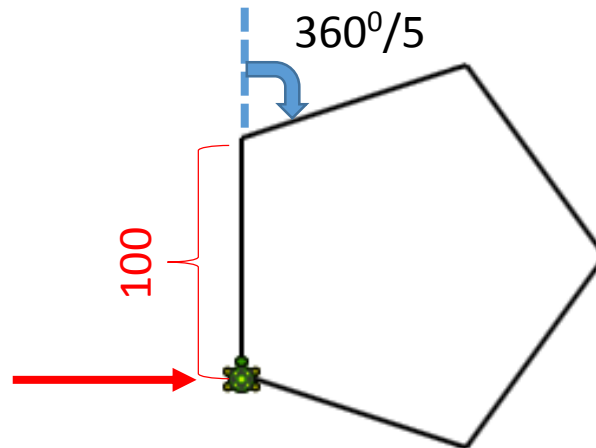


Basis Figuren

De broncode voor het vierkant is als volgt:

```
1 For n = 1 To 4
2   Turtle.Move(100)
3   Turtle.Turn(360/4)
4 EndFor
```

De n -hoek



In de afbeelding hierboven staat een vijfhoek, ook wel Pentagon. Deze figuur heeft vijf zijden van dezelfde lengte en vijf hoeken van dezelfde grootte.

Bij de gelijkzijdige driehoek geldt dat de draaihoek die de schildpad maakt 120° graden is, ofwel $360^{\circ}/3 = 120^{\circ}$. Voor het vierkant geldt dat de draaihoek 90° graden is, ofwel $360^{\circ}/4 = 90^{\circ}$.

Er is dus een patroon terug te vinden. De draaihoek van de schildpad is 360° gedeeld door het aantal hoeken van de figuur.

Voor de vijfhoek geldt dus $360^{\circ}/5 = 72^{\circ}$ graden.

In de broncode hoef je het aantal graden niet op te schrijven, je laat dit door de computer zelf uitrekenen.

De PSD voor de vijfhoek ziet er dus zo uit:





Basis Figuren

De broncode voor de vijfhoek wordt:

```
1 For n = 1 To 5
2   Turtle.Move(100)
3   Turtle.Turn(360/5)
4 EndFor
```

Deze figuren worden n-hoeken genoemd om het aantal hoeken dat ze hebben. De PSD en de broncode blijft bijna helemaal hetzelfde. Alleen het aantal iteratie stappen en het aantal hoeken waardoor gedeeld moet worden om de juiste draaihoek voor de schildpad te vinden veranderd.

De cirkel

De cirkel is een aparte figuur daar het "geen" hoeken heeft.

In de wiskundeles heb je geleerd dat de omtrek van een cirkel berekend wordt met de formule:

$$\text{Omtrek} = 2 \times \pi \times r$$

De diameter is twee keer de straal. Dus je zou dit kunnen herschrijven als de volgende formule:

$$D(\text{diameter}) = 2 \times r$$

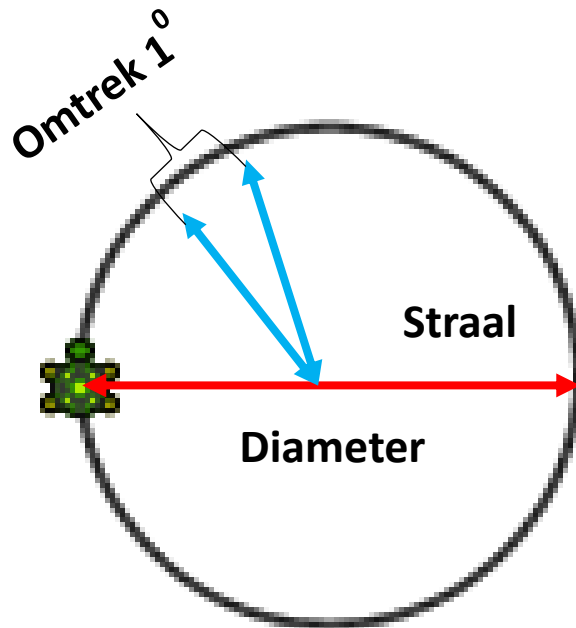
Dan wordt de formule voor de omtrek:

$$\text{Omtrek} = \pi \times D$$

De omtrek van 1° graden is dan:

$$\text{Omtrek } 1^{\circ} \text{ graden} = (\pi / 360) \times \text{Diameter}$$

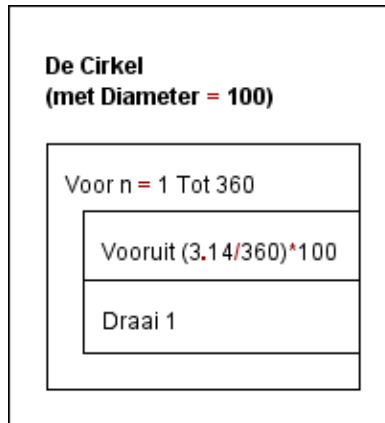
Dit is belangrijk om cirkels met de juiste diameter te kunnen tekenen.





Basis Figuren

De PSD voor een cirkel met diameter van 100 stappen, ziet er dus zo uit:



De broncode is dan:

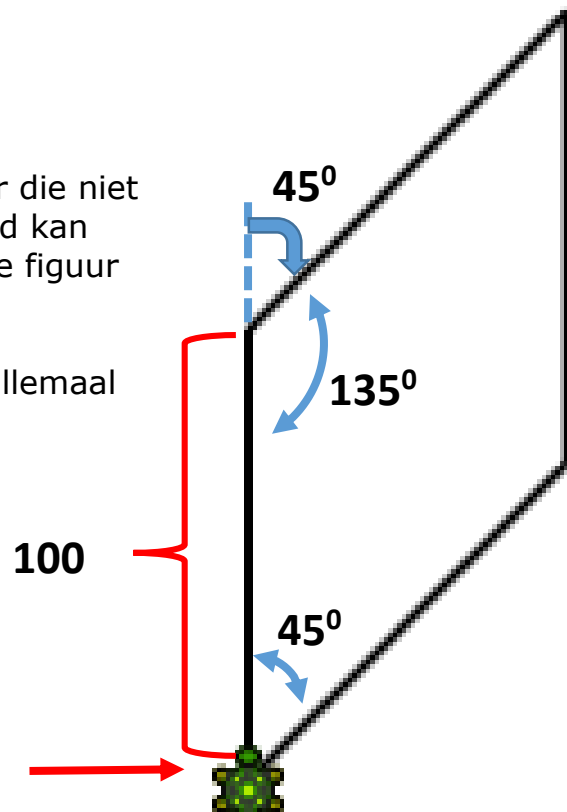
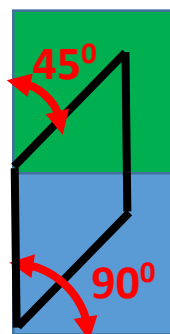
```
1 For n=1 To 360
2   Turtle.Move((3.14/360)*100)
3   Turtle.Turn(1)
4 EndFor
```

De parallellogram

De parallellogram is de enige basis figuur die niet in één keer met een iteratie-lus getekend kan worden. Dit omdat de hoeken bij dit type figuur niet allemaal even groot zijn.

De maten van de lijnen zijn echter wel allemaal even lang.

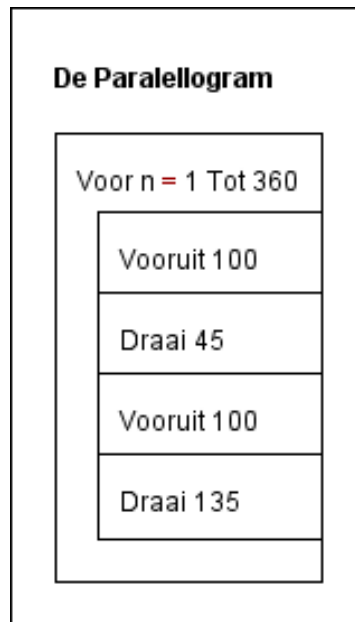
De hoek van 45° is omdat de schuine lijn de hoek van 90° precies door het midden deelt.





Basis Figuren

De PSD voor een parallellogram ziet er als volgt uit:



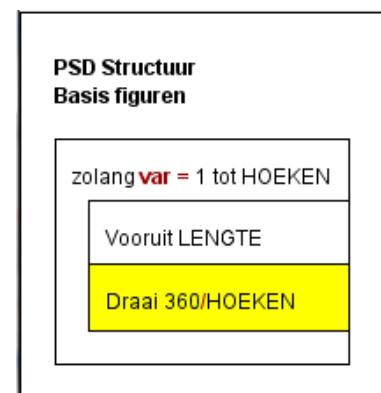
De bijbehorende broncode is:

```
1 For n = 1 To 2
2   Turtle.Move(100)
3   Turtle.Turn(45)
4   Turtle.Move(100)
5   Turtle.Turn(135)
6 EndFor
```

Basis PSD

Wanneer je alle figuren bekijkt is er een basis structuur te zien voor alle figuren behalve de parallellogram. Deze ziet er als volgt uit:

- Aantal HOEKEN van de figuur.
- LENGTE van de zijden van de figuur.





Basis Figuren

Volledige Analyse stappen

Door het gebruik van basis figuren moet het aantal stappen bij de analyse uitgebreid worden. De analyse stappen zijn nu als volgt:

1. Neem tekening over op papier
2. Bepaal de basisfiguren
 - a. Maak de PSD's van de basis figuren
3. Bepaal de begin positie van de Schildpad.
 - a. Kies altijd de meest efficiënte plek om te beginnen.
 - b. Kijk naar de begin positie van de Schildpad bij de basis figuren. Kies de begin positie zodanig dat de basis figuren eenvoudige getekend kunnen worden.
4. Geef de beginpositie en de verschillende draai richtingen aan van de Schildpad.
5. Bepaal de lengtes van de zijden van de figuur. *Alleen de verschillende lengtes aangeven.*
6. Bepaal en geef aan de grootte van de verschillende draaihoeken van de Schildpad.
7. Bepaal en geef eventueel aan de grootte van de verschillende hoeken van de figuur. *Indien dit nodig is om de draaihoeken te bepalen.*

Technieken voor samengestelde figuren

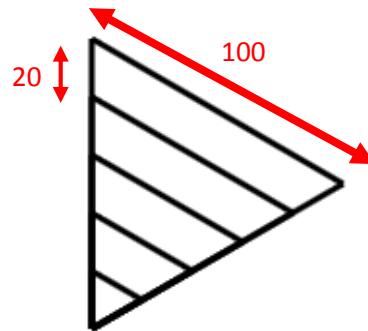
In les 4 staan er voorbeelden van verschillende (*samengestelde*) figuren die met verschillende technieken gemaakt zijn.

- Draai figuren
 - Dit zijn figuren die ontstaan door een basis figuur om zijn eigen as te laten draaien. Of een bepaald punt in de figuur.
- Samengestelde figuren
 - Deze zijn opgebouwd uit twee of meerdere basis figuren.
 - Een basis figuur kan linksom of rechtsom om een andere figuur heen of er door heen getekend worden.
- Groei – Krimp figuren
 - Dit zijn figuren die van klein naar groot worden getekend of vieze versa van groot naar klein. Dit wordt gedaan door de variabele in de herhaal lus te gebruiken als de maat van de zijden van de figuur. Men gebruikt in de herhaal dus niet meer het aantal hoeken van de figuur maar de lengtes van de zijden die men wil hebben. Bijvoorbeeld van 10 naar 100. Hiervoor is de stap noodzakelijk.



Basis Figuren

Voorbeeld Groei-Krimp techniek:



```
1 Turtle.Speed = 100
2 For X = 20 to 100 Step 20
3   For Y = 1 To 3
4     Turtle.Move(X)
5     Turtle.Turn(360/3)
6   EndFor
7 EndFor
8 Turtle.Hide()
```

De waarde van X in de buitenste herhaal-lus wordt gebruikt als maat voor de lengte van de zijden. De waarde van X wordt elke keer met 20 opgehoogd. Dus de zijden worden 20, 40, 60, 80 en 100.