

## Opdrachten week 1 les 3 – Tekenen met een schildpad

In de vorige lessen heb je kennis gemaakt met de programmeertaal Python en de IDE Thonny. Je kunt eenvoudige Python programma's schrijven met behulp van variabelen, expressies, statements, ingebouwde functies en zelf geschreven functies. Je kunt deze programma's debuggen met behulp van Thonny. In deze les ga je deze kennis toepassen in een uitgebreidere opdracht. Je leert hoe je:

- een verzameling functies kan ontwerpen die samenwerken om een bepaald doel te bereiken;
- gebruik kunt maken van encapsulation, generalization, keyword arguments, heldere interfaces en refactoring om betere Python programma's te schrijven.

In de volgende les leer je om beslissingen te nemen in een Python programma en hoe je dingen kunt herhalen door recursieve functies toe te passen.

**Lees nu hoofdstuk 4 tot en met paragraaf 4.1 van het boek<sup>1</sup>.**

**1.3.1** Voer de in [paragraaf 4.1](#) van het boek beschreven opdrachten uit. Ga pas verder als je programma de uitvoer zoals gegeven in [figuur 1](#) produceert.

**Lees nu paragraaf 4.2 van het boek.**

Wij hebben in de vorige les al kennis gemaakt met het **for**-statement.

**1.3.2** Als je nog geen **for**-statement had gebruikt bij het tekenen van het vierkant in [opdracht 1.3.1](#), pas je programma dan aan zodat je wel een **for**-statement gebruikt.

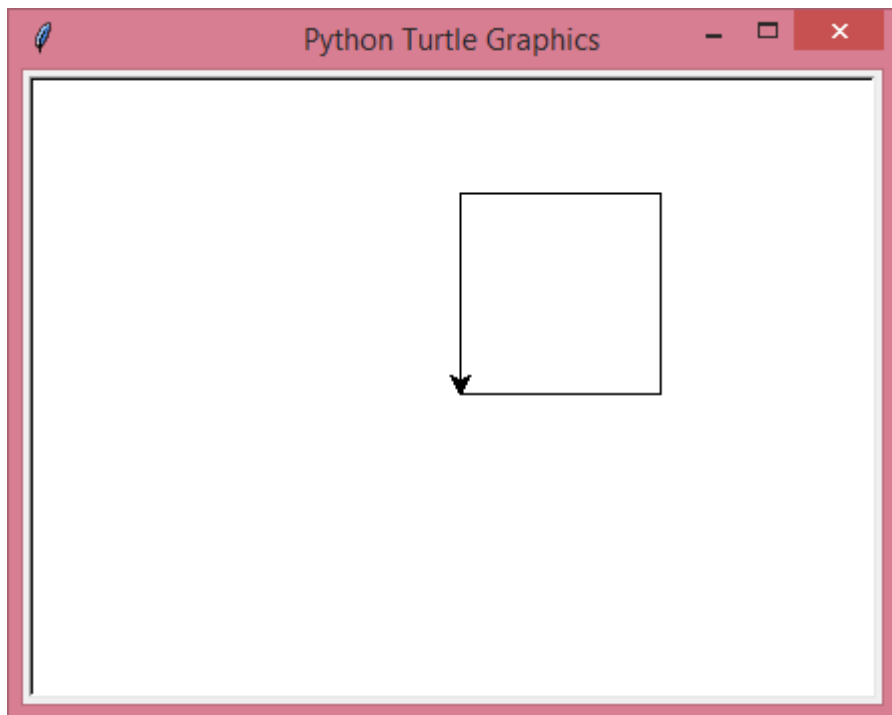
**1.3.3** Maak nu alle opdrachten die staan beschreven in [paragraaf 4.3](#) van het boek.

**1.3.4** Gebruik de testcode gegeven in [listing 1](#). Bij een goede implementatie van de functies `square`, `polygon`, `circle` en `arc` tekent deze code het figuur dat weergegeven is in [figuur 2](#).

**Lees nu paragraaf 4.3, 4.4, 4.5, 4.6, 4.7 en 4.8 van het boek.**

---

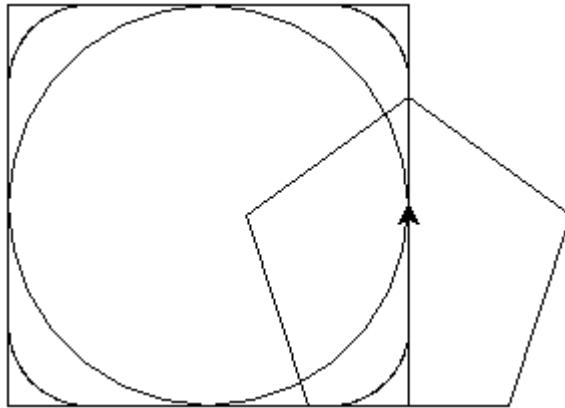
<sup>1</sup> Allen B. Downey. *Think Python: How to Think Like a Computer Scientist*. 2de ed. Green Tea Press, 2016. ISBN: 978-1-4919-3936-9. URL: <http://greenteapress.com/wp/think-python-2e/>.



**Figuur 1:** Een vierkant getekend door de turtle bob.

```
import turtle
bob = turtle.Turtle()
polygon(bob, n=5, length=100)
bob.fd(10)
for i in range(4):
    arc(bob, r=40, angle=90)
    bob.fd(120)
bob.fd(40)
bob.lt(90)
square(bob, 200)
bob.fd(100)
bob.circle(100)
turtle.mainloop()
```

**Listing 1:** De testcode waarmee de functies die je geschreven hebt in [opdracht 1.3.3](#) getest worden.



**Figuur 2:** De uitvoer van de testcode gegeven in [listing 1](#).

- 1.3.5** Vergelijk de code die gegeven is in het boek met je eigen code. Welke code vind je beter en waarom?
- 1.3.6** Beschrijf het begrip *encapsulation* in je eigen woorden en noem twee voordelen van het toepassen van encapsulation.
- 1.3.7** Beschrijf het begrip *generalization* in je eigen woorden en noem een voordelen van het toepassen van generalization.
- 1.3.8** De auteur van de testcode in [listing 1](#) heeft gebruik gemaakt van zogenoemde *keyword arguments* bij het aanroepen van de functies `polygon` en `arc`. Waarom denk je dat hier in dit geval voor gekozen is?
- 1.3.9** Beschrijf het begrip *interface van een functie* in je eigen woorden.
- 1.3.10** Beschrijf het begrip *refactoring* in je eigen woorden. Refactor de code die je hebt geschreven bij [opdracht 1.3.3](#) naar aanleiding van wat je in het boek hebt gevonden.

**Lees nu paragraaf 4.9 en 4.10 van het boek.**

- 1.3.11** Voorzie alle functies die je hebt geschreven bij [opdracht 1.3.3](#) van een *docstring* waarin het nut en de interface van de functie wordt beschreven.
- 1.3.12** Beschrijf de *preconditions* en *postconditions* van de functies die je hebt geschreven bij [opdracht 1.3.3](#).

[Paragraaf 4.11](#) van het boek definieert de verschillende (vak)termen die in hoofdstuk 4 gebruikt worden. Het kan handig zijn om deze paragraaf te raadplegen als je niet meer weet wat met een bepaalde (vak)term bedoeld wordt.

[Paragraaf 4.12](#) van het boek bevat enkele oefeningen. Deze oefeningen zijn behoorlijk pittig, maar het is wel leuk om te zien hoe je bloemen en taarten kunt tekenen met een schildpad. Studenten die een extra uitdaging zoeken kunnen zich uitleven met deze opdrachten, maar het is niet noodzakelijk om ze te maken.

**Als je de stof van deze week lastig vindt, vergeet dan niet om te oefenen met de flashcards van les 1 en 2!**