



Iteraties (Herhalingen) met While

Leerdoelen:

- While loop: Hoe je iets heel vaak uitvoert zonder dat je van te voren precies weet hoe vaak
- Conditie met een logische vlag
- Eindeloze loops



While loop

```
while < boolean expressie >:  
    < acties >
```

*“Zolang een voorwaarde waar is,
voer de acties uit”*



While loop: herhalingen

- Zolang een voorwaarde waar is, voer de acties uit.

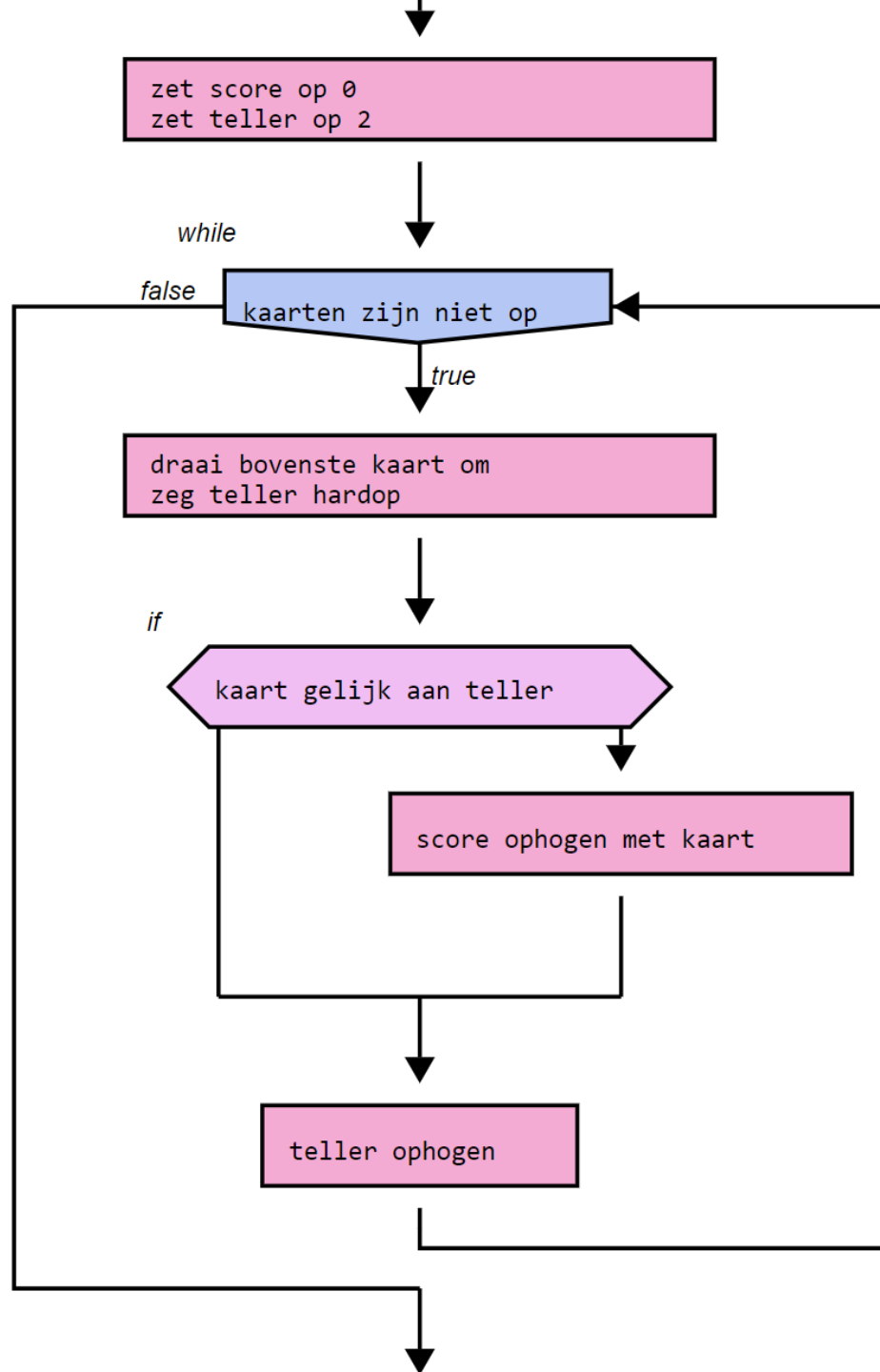
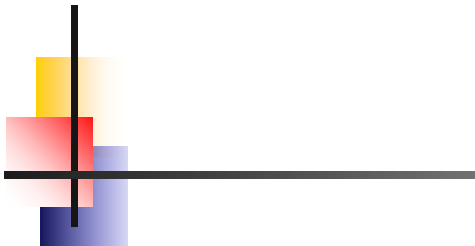
- Structuur:

```
while < voorwaarde >:  
    < acties >
```

- Bijvoorbeeld:

```
while ikHebHonger:  
    eet()
```

```
while not wachtWoordJuist:  
    voerWachtwoordIn()
```





Strafregels schrijven

Opdracht:

Druk 3 keer af: “Ik zal voortaan de opdrachten maken.”

```
while < voorwaarde >:  
    < acties >
```

Uitvoer:

1^e keer: “Ik zal voortaan de opdrachten maken.”

2^e keer: “Ik zal voortaan de opdrachten maken.”


3^e keer: “Ik zal voortaan de opdrachten maken.”

Het algoritme:

- Zolang nog geen drie keer geschreven
 - Druk strafregel af



Standaard opbouw while



Op de volgende
sheets kijken we
hoe dat precies
gaat

Vaak met een teller

De variabele:

1a) Wat is de variabele? Dus wat verandert er steeds?

1b) Geef de variabele een begin waarde (meestal is dit 0)

De herhaling `while`:

2a) Bepaal de **voorwaarde** voor herhaling: “Zolang **voorwaarde** herhaal ... “

2b) Bepaal wat je steeds wilt herhalen, de **acties**.

2c) Verander de variabele (meestal verhogen met 1).

```
<variabele initialiseren>
```

```
while <voorwaarde>:
```

```
    <acties>
```

```
    <variabele aanpassen>
```



Strafregels schrijven

Het algoritme:

- Zolang nog geen drie keer geschreven
 - Druk strafregel af

Je moet wel bijhouden hoe vaak je al geschreven hebt!

De variabele:

1a) Wat moet je onthouden? Wat verandert er steeds?

`aantal_strafregels_geschreven`

1b) Variabele beginwaarde

`aantal_strafregels_geschreven = 0`



Strafregels schrijven

Het algoritme:

- Zolang nog geen drie keer geschreven
 - Druk strafregel af

De herhaling `while`:

2a) Bepaal de **voorwaarde** voor herhaling:

```
aantal_strafregels_geschreven < 3
```

2b) Bepaal wat je steeds wilt herhalen, de **acties**.

```
print( "Ik zal voortaan de opdrachten maken." )
```

2c) Verander de variabele (meestal verhogen met 1).

```
aantal_strafregels_geschreven += 1
```




Strafregels

```
<variabele initialiseren>  
while <voorwaarde>:  
    <acties>  
    <variabele aanpassen>
```

Opdracht:

Druk 3 keer af: "Ik zal voortaan de opdrachten maken."

Code:

```
aantal_strafregels_geschreven = 0  
  
while aantal_strafregels_geschreven < 3:  
    print( "Ik zal voortaan de opdrachten maken." )  
    aantal_strafregels_geschreven += 1
```

Tracing tabel: wat gebeurt er eigenlijk?

```
aantal_strafregels = 0

while aantal_strafregels < 3:
    print( "Ik zal voortaan de opdrachten maken." )
    aantal_strafregels += 1
print( "Klaar!" )
```

Code	Waarde van aantal_strafregels	Wat geprint wordt
aantal_strafregels = 0	0	
while aantal_strafregels < 3)		
print ("Ik zal ... ")		"Ik zal ..."
aantal_strafregels += 1		
while aantal_strafregels < 3)		
print ("Ik zal ... ")		"Ik zal ..."
aantal_strafregels += 1		
while aantal_strafregels < 3)		
print ("Ik zal ... ")		"Ik zal ..."
aantal_strafregels += 1		
print("Klaar!")		"Klaar!"



Tellen van 0 tot 10

Het algoritme:

- Begin bij 0
- Zolang nog niet bij tien
 - Druk het getal af

De variabele:

1a) Wat moet je onthouden? Wat verandert er steeds?

teller

1b) Variabele beginwaarde

teller = 0



Tellen van 0 tot 10

Het algoritme:

- Begin bij 0
- Zolang nog niet bij tien
 - Druk het getal af

De herhaling `while`:

2a) Bepaal de **voorwaarde** voor herhaling:

```
teller < 10
```

2b) Bepaal wat je steeds wilt herhalen, de **acties**.

```
print( teller )
```

2c) Verander de variabele (meestal verhogen met 1).

```
teller += 1
```



Opgave 7.1.1.0: Tellen van 0 tot 10

De variabele:

1a) Wat moet je onthouden? Wat verandert er steeds?

```
teller
```

1b) Variabele beginwaarde

```
teller = 0
```

De herhaling `while`:

2a) Bepaal de **voorwaarde** voor herhaling:

```
teller < 10
```

2b) Bepaal wat je steeds wilt herhalen, de **acties**.

```
print( teller )
```

2c) Verander de variabele (meestal verhogen met 1).

```
teller += 1
```

Het algoritme:

- Begin bij 0
- Zolang nog niet bij tien
 - Druk het getal af



Even getallen afdrukken

Het algoritme:

- Begin bij 0
- Zolang nog niet bij tien
 - Druk het getal af als hij even is

De variabele:

1a) Wat moet je onthouden? Wat verandert er steeds?

teller

1b) Variabele beginwaarde

teller = 0



Even getallen afdrukken

Het algoritme:

- Zolang nog niet bij tien
 - Druk het getal af als hij even is

De herhaling `while`:

2a) Bepaal de **voorwaarde** voor herhaling

```
teller < 10
```

2b) Bepaal wat je steeds wilt herhalen, de **acties**.

```
if( teller % 2 == 0):
```

```
    print(str(teller) + " is even.")
```

2c) Verander de variabele (meestal verhogen met 1).

```
teller += 1
```

getal % 2

is steeds 2 aftrekken
dit niet meer kan

Dus als $\text{getal} \% 2$
gelijk is aan 0,
dan is **getal even**

Anders is getal oneven



Even getallen afdrukken

Opdracht:

Druk alle even getallen onder de 10 af.

<variabele initialiseren>

while <voorwaarde>:

 <acties>

 <variabele aanpassen>

Code:

```
teller = 0

while teller < 10:
    if( teller % 2 == 0):
        print(str(teller) + " is even.")
    teller += 1
```




Standaard opbouw while

Vaak met een teller

Probeer nu opgave 7.1.2.1 zelf

De variabele:

- 1a) Wat is de variabele? Dus wat verandert er steeds?
- 1b) Geef de variabele een begin waarde (meestal is dit 0)

De herhaling `while`:

- 2a) Bepaal de **voorwaarde** voor herhaling: “Zolang **voorwaarde** herhaal ... “
- 2b) Bepaal wat je steeds wilt herhalen, de **acties**.
- 2c) Verander de variabele (meestal verhogen met 1).

```
<variabele initialiseren>
while <voorwaarde>:
    <acties>
    <variabele aanpassen>
```



Met een Logische Vlag.

De variabele:

1a) Wat is de conditie **vlag**: `spel_is_afgelopen`

1b) Geef de vlag een begin waarde:

`spel_is_afgelopen = False`

De herhaling `while`:

2a) Bepaal de **voorwaarde** voor herhaling: “Zolang **voorwaarde** herhaal ... “

`while spel_is_afgelopen == False`

2b) Bepaal wat je steeds wilt herhalen, de **acties**.

2c) Verander in de juiste geval de **vlag** waarde

`spel_is_afgelopen = True`



Met een Logische Vlag.

De variabele:

1a) Wat moet je onthouden? Wat verandert er op een later moment?

```
pincode_juist
```

1b) Variabele beginwaarde

```
pincode_juist = False
```

De herhaling `while`:

2a) Bepaal de **voorwaarde** voor herhaling:

```
pincode_juist == False
```

2b) Bepaal wat je steeds wilt herhalen, de **acties**.


```
invoer = input( "Voer pincode in: " )
gebruikers_antwoord = int( invoer )
if gebruikers_antwoord == PINCODE:
    goed_geraaden = True
else: ...
```

2c) Verander de variabele

```
pincode_juist = True
```

Het algoritme:

- Zolang pincode niet juist
 - Vraag pincode
 - Controleer pincode



Hooguit 3 pogingen

```
PINCODE = "1234"

goed_gerden = False #om te beginnen staat de vlag op False

while goed_gerden == False: #herhaal zolang vlag nog op False staat
    invoer = input( "Voer pincode in: " ) #vraag de wachtwoord
    gebruikers_antwoord = int( invoer )

    if gebruikers_antwoord == PINCODE: #goed gerden
        goed_gerden = True #de vlag wordt op True gezet,
                           #hierna zal de while loop
                           #niet opnieuw uitgevoerd gaan worden
    else: #fout gerden, voer loop weer uit en vraag opnieuw
        print("Fout... probeer het opnieuw")
```

Breidt nu de code uit zodat je maximaal 3 pogingen hebt.



Welk uitvoer hoort bij deze code?

```
teller = 2
while teller < 8:
    print( teller )
    teller += 1
```

ANTWOORD:

D

- A 7, 6, 5, 4, 3, 2.
- B 2, 3, 4, 5, 6, 7, 8.
- C 8, 7, 6, 5, 4, 3, 2, 1, 0.
- D 2, 3, 4, 5, 6, 7.



Wat doet deze code?

```
teller = 0
while teller < 100:
    print( "teller: " + str(teller) )
```

Drukt af:

teller: 0

teller: 0

teller: 0

teller: 0

...

--stopt dus nooit: CRASH!! -



Welk van de volgende stukken code drukt de getallen 1 t/m 3 af?

ANTWOORD:

D

```
A. numToPrint = 1
   while numToPrint > 4:
       print(numToPrint)
       numToPrint +=1

B. numToPrint = 0
   while numToPrint < 3:
       print(numToPrint)
       numToPrint += 1

C. numToPrint = 1
   while numToPrint > 3:
       print(numToPrint)
       numToPrint += 1

D. numToPrint = 1
   while numToPrint < 4:
       print(numToPrint)
       numToPrint += 1
```