



Python

Informatica

Renske Smetsers



Hoofdstuk 6: Condities

Leerdoelen:

- ❑ Wat boolean expressies zijn
- ❑ Boolean waarden ``True`` en ``False``
- ❑ Vergelijkingen met ``<``, ``<=``, ``==``, ``>``, ``>=``, en ``!=``
- ❑ De ``in`` operator- Logische operatoren ``and``, ``or``, en ``not``
- ❑ Conditionele statements met ``if`` en ``else``, ``elif``
- ❑ Stroomdiagrammen
- ❑ Inspringen
- ❑ Geneste condities



Terugvinden van je eigen werk

- In linker kolom: rechtermuisknop:
‘Show in Explorer’

Hoofdstuk 5 (Eenvoudige functies)

-> in map lesson 6

Turtle Graphics (functies en parameters)

-> in map lesson 7



Vergelijken: True of False

- Is gelijk aan?

`gewicht == 60`

- Is kleiner dan?

`gewicht < 60`

- Is kleiner of gelijk aan?

`gewicht <= 60`

- Is groter dan?

`gewicht > 60`

- Is groter of gelijk aan?

`gewicht >= 60`

- Is ongelijk aan?

`gewicht != 60`

**Het resultaat is altijd
True of False**



BELANGRIJK: Verschil tussen **vergelijken** en **toekennen**

TOEKENNEN

- `gewicht = 60`
 - **Opdracht:** gewicht **wordt** 60

VERGELIJKEN

- `gewicht == 60`
 - **Vraag:** is gewicht gelijk aan 60?

Kiezen met Kaarten





Voorbereiding

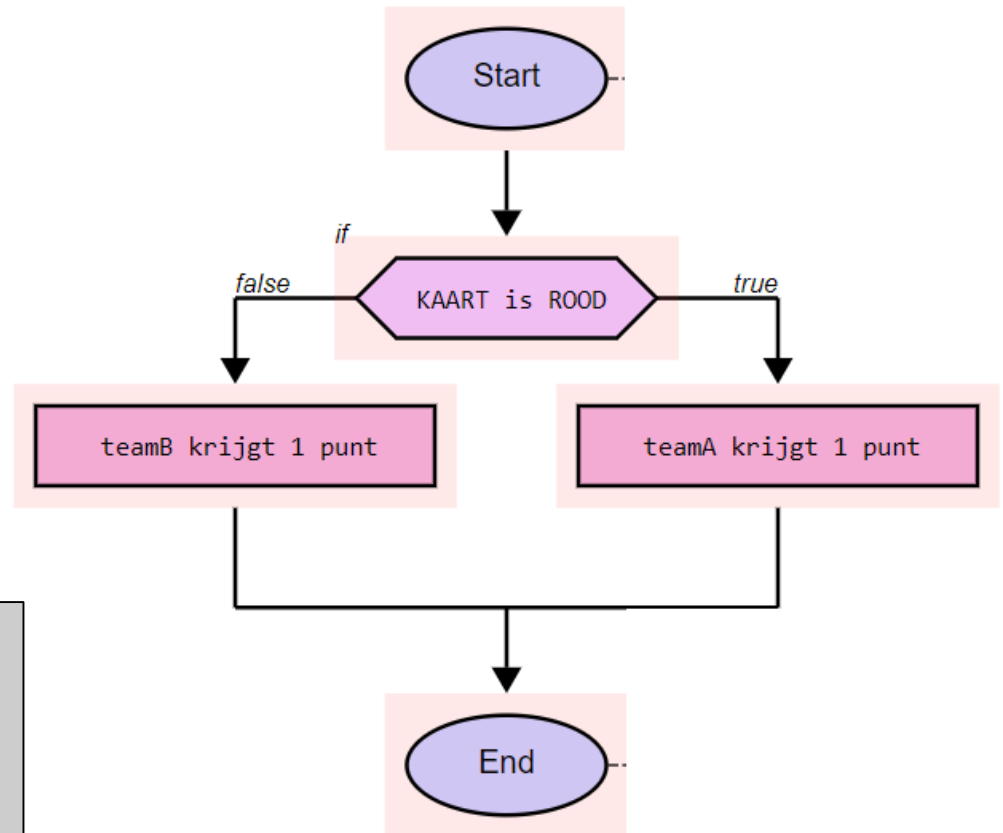
- Maak tweetallen
- Ga bij een ander tweetal zitten
- Spreek per viertal af welk tweetal *team A* heet, en welke *team B*
- Elk viertal krijgt (ongeveer) 15 kaarten van verschillende kleuren en alleen de getallen (2 t/m 9)



Inleiding

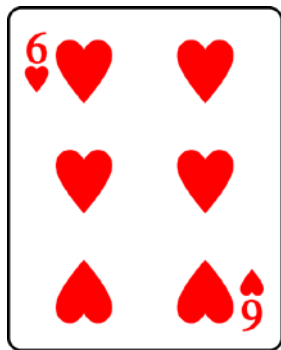
- Vandaag gaan we een simpel kaartspelletje spelen.
- Twee teams spelen tegen elkaar.
- Elk team bestaat uit twee spelers.
- Je trekt steeds gezamenlijk één kaart van de stapel.
- Op basis van het algoritme bepaal je welk team de punten krijgt en hoeveel. Het algoritme komt op het scherm en wordt soms gewijzigd.
- Het team dat als eerste tien punten of meer heeft wint het spel.

Voorbeeld



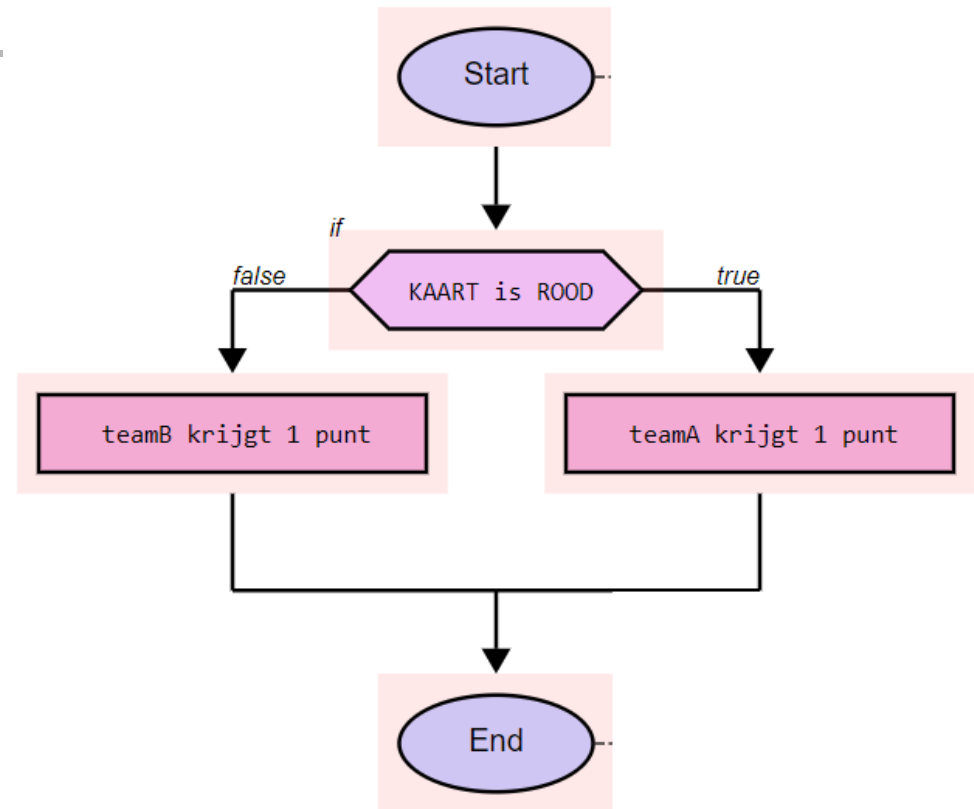
```
if (KAART is ROOD):  
    teamA krijgt 1 punt  
Else:  
    teamB krijgt 1 punt
```

Voorbeeld



TeamA = 1 punt
TeamB = 0 punten

```
if (KAART is ROOD):  
    teamA krijgt 1 punt  
else:  
    teamB krijgt 1 punt
```

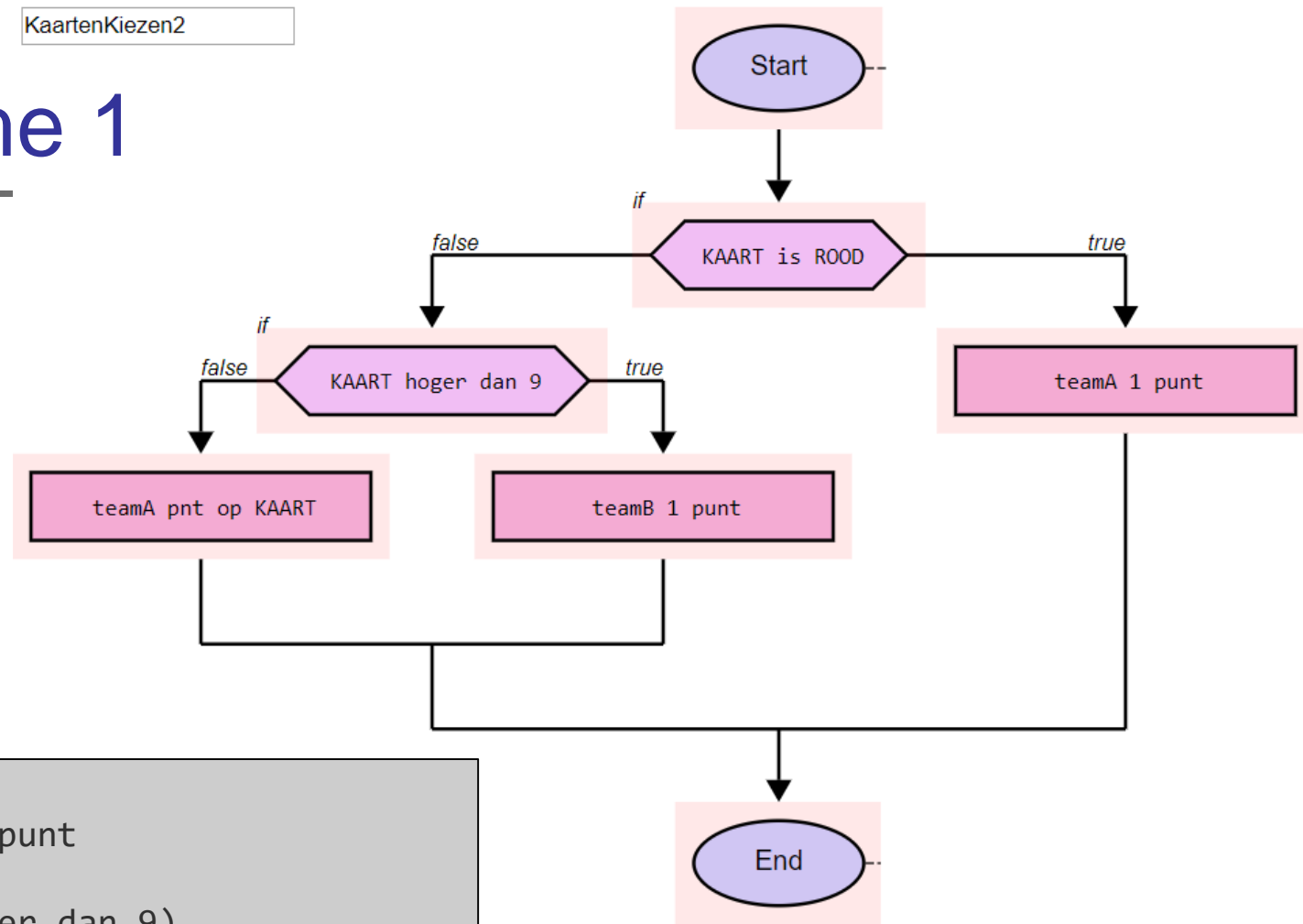




We gaan nu spelen...

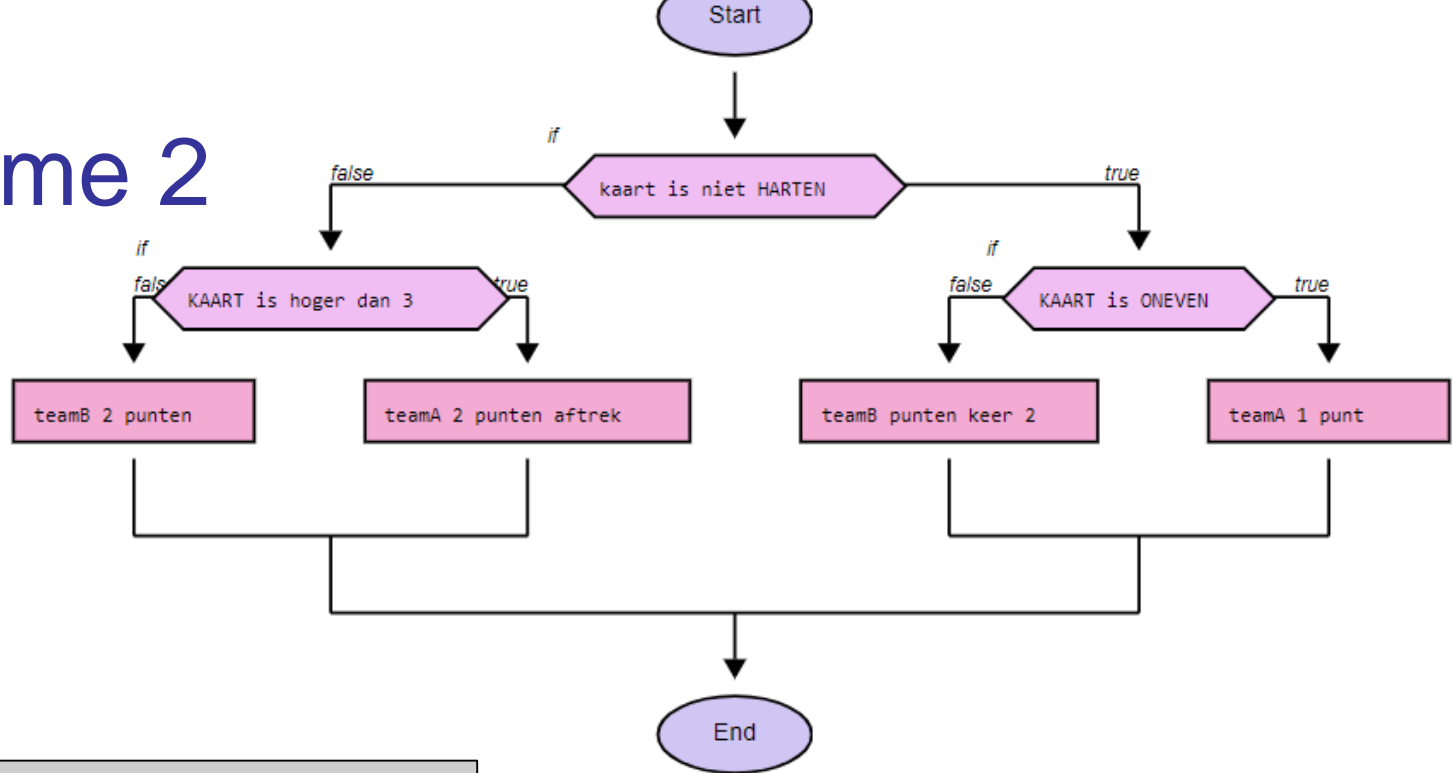
- Jullie zijn in teams verdeeld die tegen elkaar spelen.
- Ik laat steeds één kaart zien en jullie bepalen gezamenlijk welk team de punten krijgt (jullie gebruiken nog niet je eigen setje).
- Het team dat als eerste tien punten heeft wint het spel.
- Daarna begin je een nieuw spel, beide teams hebben weer nul punten.

Algoritme 1



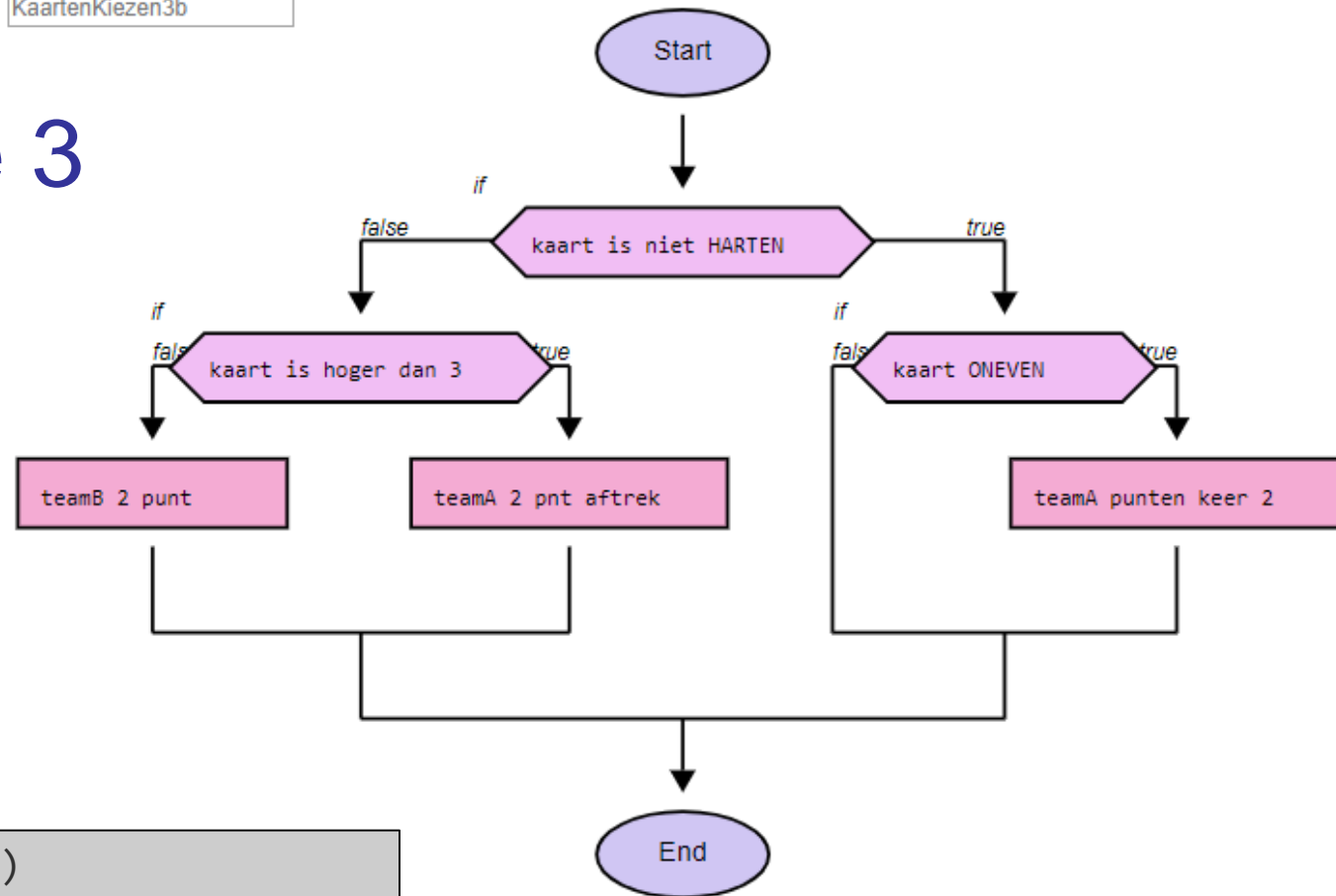
```
if (KAART is ROOD)
    team A krijgt 1 punt
else
    if (KAART is hoger dan 9)
        team B krijgt 1 punt
    else
        team A krijgt aantal punten
        dat op de kaart staat
```

Algoritme 2



```
if (KAART is niet HARTEN)
    if (KAART is ONEVEN)
        team A krijgt 1 punt
    else
        team B krijgt punten keer 2
else
    if (KAART is hoger dan 3)
        team A krijgt 2 punten aftrek
    else
        team B krijgt 2 punten
```

Algoritme 3

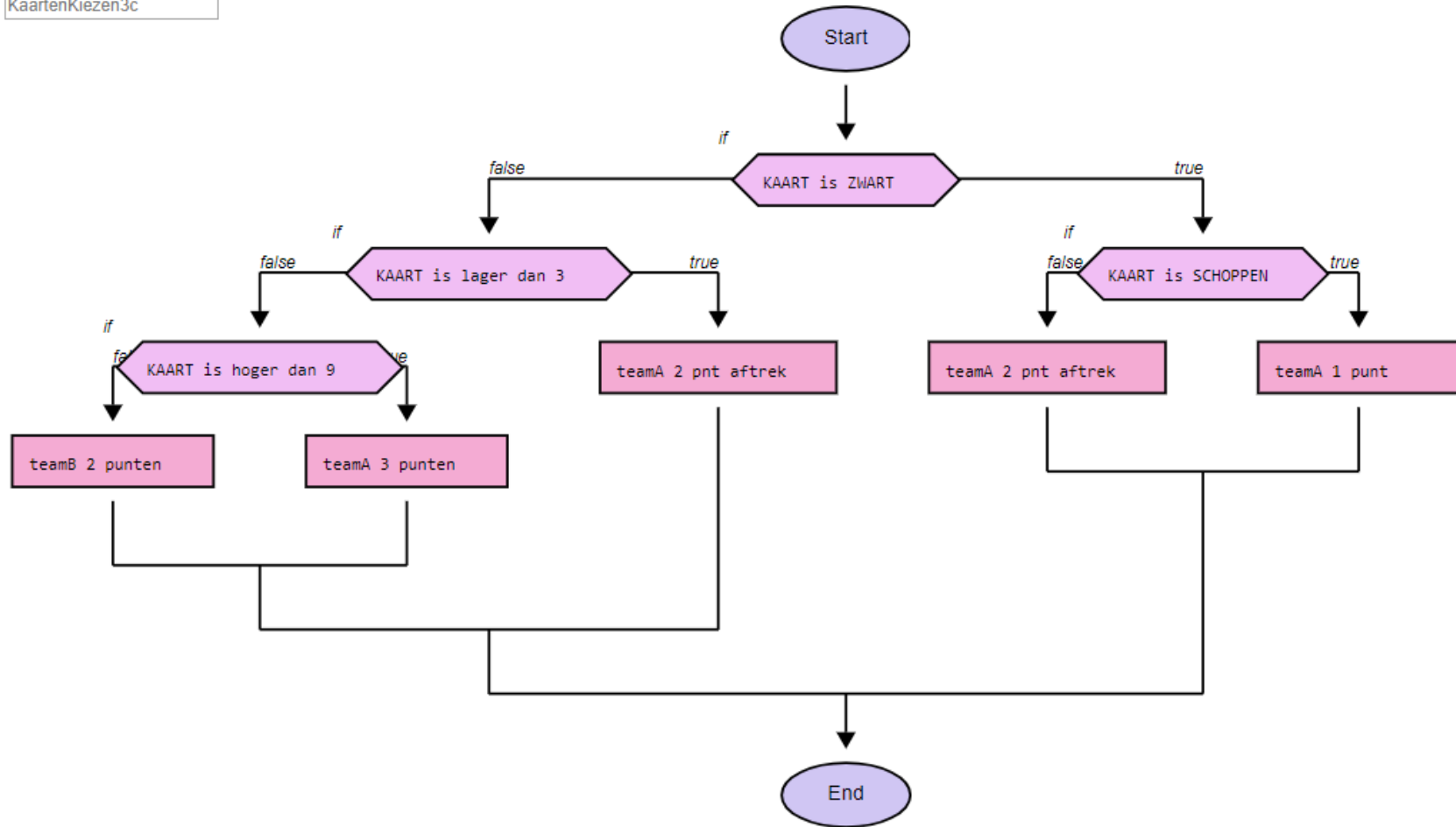


```

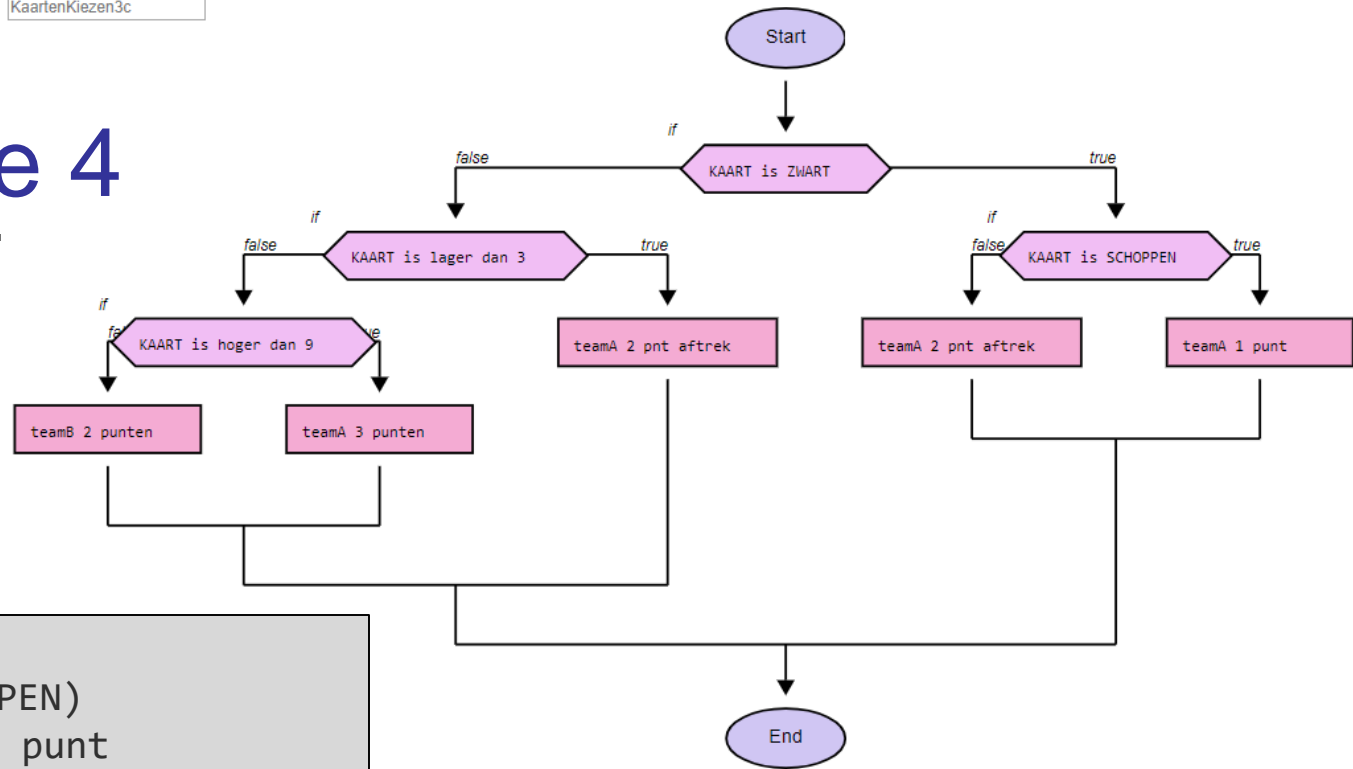
if (KAART is niet HARTEN)
    if (KAART is ONEVEN)
        team A krijgt punten keer 2
    else
        if (KAART is hoger dan 3)
            team A krijgt 2 punten aftrek
        else
            team B 2 punten
  
```

Algoritme 4

KaartenKiezen3c



Algoritme 4



```

if (KAART is ZWART)
    if (KAART is SCHOPPEN)
        teamA krijgt 1 punt
    else
        teamA krijgt 2 punten aftrek
else
    if (KAART is lager dan 3)
        teamA krijgt 2 punten aftrek
    else
        if (KAART hoger dan 9)
            teamA krijgt 3 punten
        else
            teamB krijgt 2 punten
  
```




Na afloop van het spel

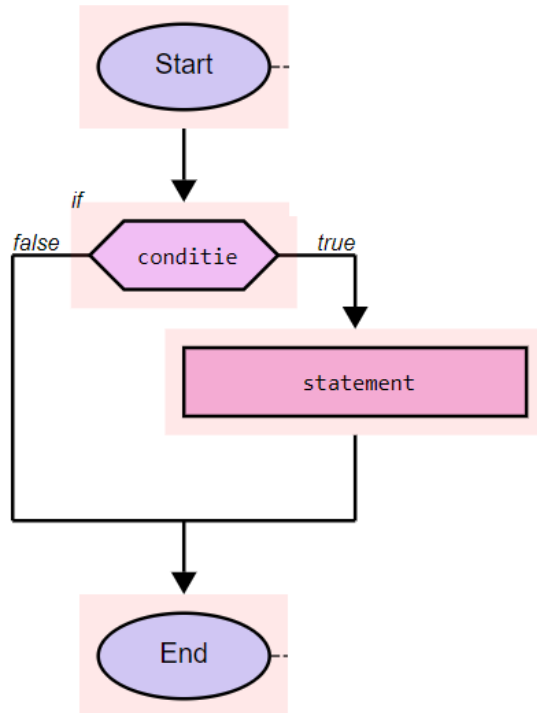
In dit spel heb je bepaald hoeveel punten je kreeg met behulp van een *algoritme*. Dit algoritme werd weergegeven in een *stroomdiagram* en in *pseudocode*. In elk algoritme zaten één of meerdere *keuzes* (=selectie).

In een stroomdiagram zie je zo'n keuze terug als een ruit, waarin een *conditie* staat: een vraag die je kan worden beantwoorden met ja of nee.

In pseudocode en in programmeercode zie je zo'n keuze terug als een *if-then-constructie* of een *if-then-else-constructie*. Keuzes kunnen ook *genest* zijn, dan komt er na een keuze gelijk nog een keuze.

Keuze met één pad

Stroomdiagram



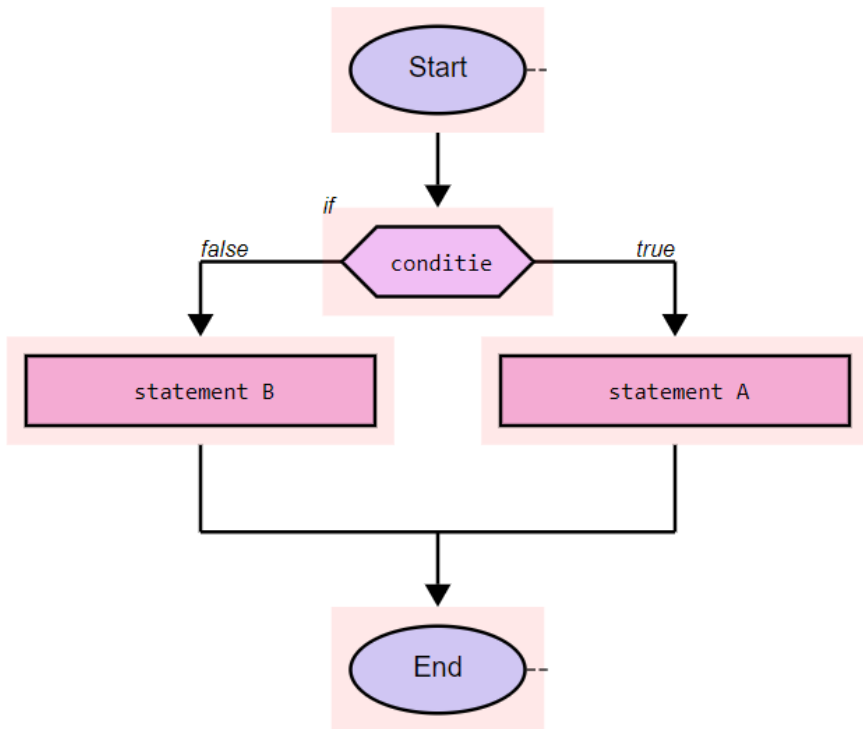
Pythoncode

```
if conditie:  
    statement
```

Let op:
inspringen

Keuze met twee paden

Stroomdiagram



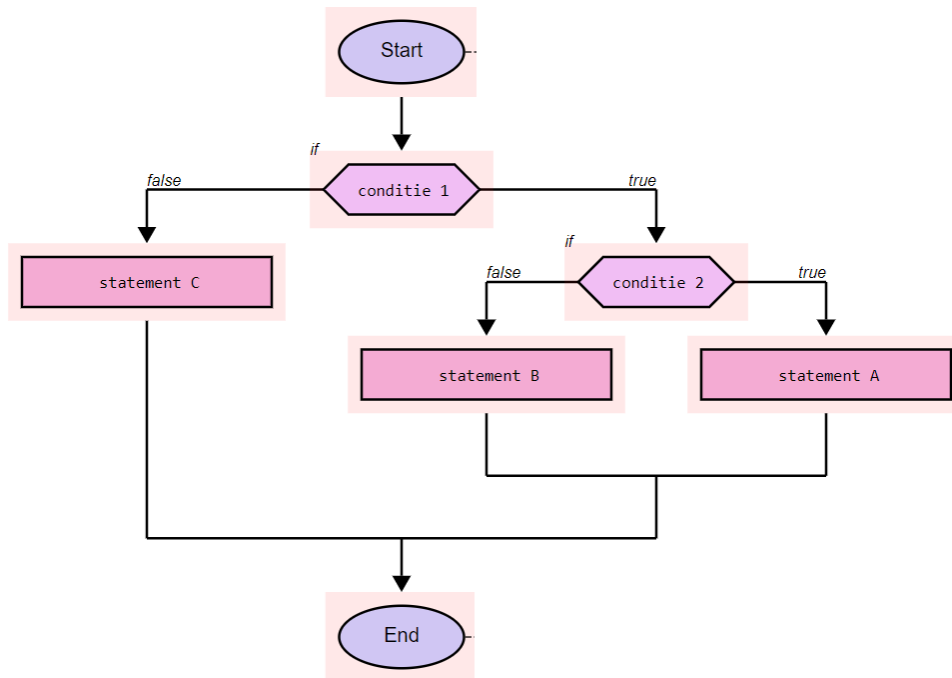
Pythoncode

```
if conditie:  
    statement A  
else:  
    statement B
```

Let op:
inspringen

Geneste Keuze

Stroomdiagram



Pythoncode

```
if conditie 1:  
    if conditie 2:  
        statement A  
    else:  
        statement B  
else:  
    statement C
```

Let steeds op niveau
van inspringen!



Keuzes

Keuzes met: if

```
if aantal_uren_geslapen <= 6  
    print("Ga terug naar bed!")
```

Keuzes met: if..then..else

```
if beltegoed == 0:  
    print( "Je beltegoed is op!" )  
    print( "Dat is jammer voor je.")  
else:  
    print ( "Je hebt nog beltegoed.")
```



Keuzes met: if... elif...else

← 25 jaar? laat je
legitimatie zien!

← 18 jaar verkopen
wij geen alcohol


NIX18

```
if leeftijd < 18:  
    print("Je mag geen bier kopen.")  
elif leeftijd < 25:  
    print ("Je mag wel bier kopen, maar je moet je legitimeren.")  
else:  
    print ("Je hoeft geen legitimatie te laten zien.")
```



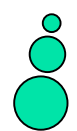
Samengestelde condities

- not
- and
- or

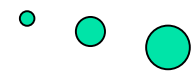


Deze ken je nog
uit de eerste
periode!


- Gebruik haakjes!



$(a \text{ or } b) \text{ and } c$
is anders dan
 $a \text{ or } (b \text{ and } c)$



Toen heb je ook
geleerd dat haakjes
belangrijk zijn.



$(\text{cola or sinas}) \text{ and chips}$
is anders dan
 $\text{cola or } (\text{sinas and chips})$




NIX18: samengestelde condities

De supermarkten in Nederland willen geen alcohol en tabak verkopen aan jongeren onder de 18 jaar. Aan jongeren tot 25 jaar die alcohol of tabak willen kopen wordt daarom gevraagd om klaar te staan met een geldig legitimatiebewijs.

« 25 jaar? laat je
legitimatie zien!
« 18 jaar verkopen
wij geen alcohol

NIX18

```
if (leeftijd < 25 and leeftijd >= 18):  
    print("Geen probleem, maar toon wel legitimatie.")  
  
elif leeftijd < 18:  
    print("Je krijgt NIX mee!")  
  
else:  
    print("Geen probleem.")
```



Trouwens: hoe
zou jij testen op
correctheid?



Testen: pak grensgevallen

« 25 jaar? laat je
legitimatie zien!

« 18 jaar verkopen
wij geen alcohol

NIX18

Test met grensgevallen:

17
18
19
24
25
26

```
leeftijd = 18

if (leeftijd < 25 and leeftijd >= 18):
    print("Geen probleem, maar toon wel legitimatie.")

elif leeftijd < 18:
    print("Je krijgt NIX mee!")

else:
    print("Geen probleem.")
```