Les 8 voorbereiding

# Theorie

## Screencast onderwerp 5: ArrayList

<http://www.youtube.com/playlist?list=PLpd9jJvk1PjmJddDbml4Yh_s99TUOJmtx>

## Reader over uml klassendiagrammen en sequentiediagrammen

Les 08 Reader UML class en sequence diagrams

## Boek

### Hoofdstuk 4

4.1 tot en met 4.9 pagina 103 tot en met 124 (bovenaan)

4.11 pagina 131 tot en met 134

### Hoofdstuk 5

5.6.1 en 5.6.2 pagina 189 tot en met 191

5.7 pagina 193 tot en met 195

# Opgave isBenedenScherm

In de screencast 6.1 over arraylist op tijdstip 8:25 – 8:40 wordt de methode isBenedenScherm aan het hoofdprogramma toegevoegd.

Kan deze methode niet beter in de klasse Deeltje staan? Leg uit waarom wel of niet. Moet je de methode dan nog aanpassen?

Gebruik in de uitleg het begrip: verbergen van informatie (Boek 5.11 pagina 200 tot en met 203).

# Opgave for-lus om elementen te verwijderen

## Opgave A

Laat zien dat een for-lus die van 0 tot de grootte van de ArrayList loopt niet gebruikt kan worden om elementen te verwijderen.

Gebruik daarvoor het onderstaande programma en teken elke keer dat de tweede for lus doorlopen wordt (regel 8 tot en met 11) het geheugenmodel van de lijst. Zie de reader op #OnderwijsOnline (bij les 6) om te zien hoe je een ArrayList tekent in het geheugenmodel.

public static void main(String[] args) {  
 ArrayList<String> lijst = new ArrayList<String>();  
   
 for (int i = 0; i < 4; i++) {  
 lijst.add("element: " + i);  
 }  
   
 for (int i = 0; i < lijst.size(); i++) {  
 String s = lijst.get(i);  
 lijst.remove(s);  
 }  
 }

## Opgave B

Hoe komt het dat in screencast 6.1 op tijdstip 9:00 toch alle Deeltjes uit de ArrayList worden verwijderd?

## Opgave C

Maak een programma met for-lus die terugtelt van de grootte van de ArrayList naar 0 en laat zien dat deze lus wel alle elementen uit de ArrayList verwijdert.

# Opgave foreach voor verwijderen

Maak een kort programmatje waarmee je onderzoekt of je met een foreach lus alle elementen uit een ArrayList kunt verwijderen.