Les 2 voorbereiding

# Theorie

## Screencast onderwerp 1 het geheugenmodel

<http://www.youtube.com/playlist?list=PLpd9jJvk1PjmtR_LDjx6Ao8ddS5Q_-30a>

# Opgave Geheugenmodel volgorde

Hieronder staan een aantal acties die in het geheugenmodel plaats kunnen vinden. Zet ze in de juiste volgorde.

1. Lokale variabelen een waarde geven.
2. Stack frame verwijderen.
3. Returnwaarde kopiëren.
4. Stack frame plaatsen.
5. Globale variabelen plaatsen.
6. Lokale variabelen plaatsen.

*Sorry ik ga niet nog een keer drie kwartier kijken voor deze opdracht ik heb wel gekeken.*

Tags: geheugenmodel, stack, stack frame, return,

# Opgave Pijl in het geheugenmodel

## Onderdeel A

Geef zo duidelijk mogelijk aan wat een pijl in het geheugenmodel precies betekent.

*De pijl verwijst naar een array of een andere plek waar een variabele wordt aangepast in het filmpje.*

## Onderdeel B

Geef aan waar deze pijl precies moet beginnen en waar deze pijl precies naar wijst.

*De pijl begint op het vakje waar je een variabele wilt hebben en wijst naar waar je naar verwijst.*

Tags: geheugenmodel, variabele, array, referentie, geheugenadres

# Opgave Geheugenmodel van doeKeerTwee

## Onderdeel A

In onderstaande code is de functie doeKeerTwee gegeven:

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10 | int testGetal = 5;  void setup() {  doeKeerTwee(testGetal);  println(testGetal);  }  void doeKeerTwee(int getal) {  getal = 2 \* getal;  } |

Teken het geheugenmodel op het moment dat doeKeerTwee op regel 4 is uitgevoerd, maar het stack frame van deze functie nog niet is verwijderd.

## Onderdeel B

Hieronder staat de functie doeKeerTwee

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13 | int[] testGetallen = {5, 5};  void setup() {  doeKeerTwee(testGetallen);    println(testGetallen);  }  void doeKeerTwee(int[] getallen) {  for (int i = 0; i < getallen.length; i++) {  getallen[i] = 2 \* getallen[i];  }  } |

Teken het geheugenmodel op het moment dat doeKeerTwee op regel 4 is uitgevoerd, maar het stack frame van deze functie nog niet is verwijderd.

## Onderdeel C

Vergelijk de geheugenmodellen uit Onderdeel A en Onderdeel B met elkaar en verklaar aan de hand van deze modellen waardoor de globale variabele testGetal niet van waarde is veranderd, maar de testGetallen wel.

Tags: stap voor stap doorlopen, geheugenmodel, referentievariabele, primitieve variabele, lokale variabele, globale variabele.