Les 3 lesprogramma

# Opgave Damstenen deel 2

Individueel, programmeervaardigheid

## Onderdeel A

Maak in het hoofdprogramma een array met vier damstenen: twee voor de witte speler en twee voor de zwarte speler. Initialiseer deze array en vul de array met vier nieuwe damsteenobjecten.

## Onderdeel B

Pas de code in de draw-lus zo aan dat alle damstenen uit de array getekend worden.

Tags: array met objecten

# Opgave Persoonsverandering

Klassikaal, begrip

## Onderdeel A

Gegeven onderstaande code:

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17 | class Persoon {  String naam;    Persoon(String naam) {  this.naam = naam;  }  }  void setup() {  Persoon p1 = new Persoon("han");  Persoon p2 = new Persoon("ica");    p1.naam = p2.naam;  p2.naam = "kareltje";    println(p1.naam);  } |

Teken het geheugenmodel op het moment dat println(p1.naam) op regel 16 wordt uitgevoerd en verklaar de uitvoer in de console (je hoeft het stack frame van println niet te tekenen).

## Onderdeel B

Het programma wordt een beetje aangepast (zie vetgedrukte code) :

|  |  |
| --- | --- |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17 | class Persoon {  String naam;    Persoon(String naam) {  this.naam = naam;  }  }  void setup() {  Persoon p1 = new Persoon("han");  Persoon p2 = new Persoon("ica");    **p1 = p2;**  p2.naam = "kareltje";    println(p1.naam);  } |

Teken het geheugenmodel op het moment dat println(p1.naam) op regel 16 wordt uitgevoerd (zonder stack frame van println) en verklaar het verschil van de uitvoer in de console met onderdeel A.

Tags: geheugenmodel, primitieve types, referentietypes

Reflectieopgaven

# Opgave Testen

Bij verschillende opgaven is er gevraagd om een test te schrijven voor een functie.

Op welke plek, of plekken schrijf je in een processingprogramma die test?

Geef een schets van de code die je maakt, als je een test moet schrijven.

Wanneer is de test van goede kwaliteit?

# Opgave Foutmeldingen

Er zijn een aantal belangrijke foutmeldingen besproken.

Welke foutmeldingen zijn dit? Waardoor wordt elke foutmelding veroorzaakt? Geef per foutmelding de oplossing.

# Opgave Programmeerstijl

Het hanteren van een goede programmeerstijl is een paar keer teruggekomen. Zoek de opgaven op waarin programmeerstijl aan de orde is geweest. Geef een paar vuistregels voor een goede programmeerstijl.

# Opgaven Begrippen verbinden

Hieronder staan verschillende groepen met begrippen. Maak een zin waarin je elk begrip verbind door de constructie: “bevat/kan bevatten”, “is/kan zijn”, “wordt weergegeven door/kan weergegeven worden door”, “zorgt voor/kan zorgen voor.

Je mag de begrippen meervoud maken als dit de zin ten goede komt.

Bijvoorbeeld:

Rechthoek, Stack frame, Geheugenmodel

Levert

Het geheugenmodel kan stack frames bevatten die elk worden weergegeven door een rechthoek.

* Runtime administratie, geheugenmodel, variabelen.
* Stack frame, geheugenmodel, functieaanroep.
* Primitieve variabele, referentievariabele, object, array, integer
* Geheugenadres, geheugenmodel, pijl.
* Lokale variabele, parameter
* Globale variabele declaratie, lokale variabele declaratie, functiedefinitie
* Functiedefinitie, functieaanroep.

# Opgave Opgaven en tags

## Onderdeel A

Bekijk nogmaals elke opgave en bekijk welke opgaven je hebt gemaakt, welke opgaven je denkt te beheersen en van welke opgave je onderdelen nog niet begrijpt.

## Onderdeel B

Bij elke opgave staan tags genoemd. Benoem de rol van elke tag in de opgave, of vraag dit na. Ga ook na of er een tag bij de opgave mist.