



Cursusomvang (nominale waarden; effectieve waarden kunnen verschillen per opleiding)

Studiepunten 6.0 Studietijd 180 u Contacturen 60.0 u

Aanbodsessies en werkvormen in academiejaar 2017-2018

A (semester 1)	Nederlands	hoorcollege	30.0 u
		werkcollege: PC- klasoefeningen	30.0 u

Lesgevers in academiejaar 2017-2018

Coolsaet, Kris WE02 Verantwoordelijk lesgever

Aangeboden in onderstaande opleidingen in 2017-2018

	stptn	aanbodsessie
Bachelor of Arts in de wijsbegeerte	6	A
Bachelor of Science in de informatica	6	A
Bachelor of Science in de wiskunde	6	A

Onderwijstalen

Nederlands

Trefwoorden

Java, programmeren, objectgericht

Situering

Kennen en toepassen van de basisbeginselen van het objectgericht programmeren met behulp van de programmeertaal Java. Praktische programmeerervaring opdoen in dit domein en zelf programma's kunnen schrijven. Kunnen evalueren of code van goede kwaliteit is, en de kwaliteit verbeteren indien nodig. Het belang onderkennen (en er actief naar handelen) dat een programmeur of software-ontwikkelaar nog zelden een volledig programma eigenhandig ontwikkelt, maar dat programmeren in de eerste plaats bestaat uit het hergebruiken van bestaande programmamodules en het inpluggen van kleine programmaonderdeeltjes in bestaande software-raamwerken.

Inhoud

- Kernconcepten van objectgericht programmeren. Hierbij komen onder andere de volgende concepten aan bod: objecten, klassen en interfaces, polymorfisme, dynamische binding, overerving, velden, constructoren en methoden.
- Ontwerpen van eenvoudige klassen en interfaces (welke attributen, methoden heb je nodig), onderverdelen van programma's in klassen, klassen gebruiken waarvan enkel de specificatie gekend is, interfaces implementeren op basis van een specificatie.
- Schrijven van heldere code (programmeren met 'stijl'): naamkeuze, indentatie, commentaar, opsplitsen in methoden.
- Het correct afhandelen van fouten tijdens de uitvoering van een programma door middel van uitzonderingen (exceptions).
- Het opsplitsen van een opdracht in deelopdrachten om het probleem beheersbaar te maken.
- Het schrijven van eenvoudige algoritmes die gebruik maken van lussen en eenvoudige gegevensstructuren zoals tabellen (arrays) en lijsten.
- Kernconcepten voor het schrijven van software van een hoge kwaliteit met een focus op uitbreidbaarheid en leesbaarheid. Concepten zoals verantwoordelijkheid, encapsulatie, koppeling, cohesie, en 'refactoring'. De begrippen 'mutator' en 'accessor'. Ontwerp van een geschikte klassenstructuur voor een bepaald probleem.
- Het toepassen van deze concepten en principes in the programmeertaal Java, waarin bijkomend een aantal Java-specifieke elementen nader bestudeerd worden, waaronder
 - Primitieve gegevenstypes en elementaire bewerkingen.

- 1- en meer-dimensionale arrays, en collectie klassen.
- Afschermen van de toegang tot methodes en velden: privaat vs. publiek (vs. protected, package), pakketten (packages).
- Checked en unchecked exceptions.
- Syntactic overloading

Begincompetenties

Enige vertrouwdheid met de computer is gewenst. Voorafgaande programmeerervaring is niet nodig.

Eindcompetenties

- 1 Het kunnen ontleden van een opdracht in natuurlijke taal, en deze opdracht opsplitsen in een aantal deelopdrachten.
- 2 Praktische en theoretische kennis hebben van de basisprincipes van het objectgericht programmeren (objecten, klassen, overerving, interfaces. implementatie, polymorfisme,...)
- 3 Kunnen omzetten van een opdracht gesteld in natuurlijke taal naar een object-gericht programma, of onderdeel van een programma, geschreven in de programmeertaal Java. Wanneer het over een korte eenvoudige opdracht gaat die enkel courante bibliotheekmodules gebruikt, dan moet de student dit kunnen zonder gebruik te maken van een computer en zonder de documentatie te raadplegen.
- 4 Bij het implementeren van een probleem moet de student goede keuzes kunnen maken tussen verschillende alternatieven op basis van de aangeleerde kwaliteitscriteria.
- 5 Schrijven van een programma-onderdeel dat voldoet aan vooropgegeven specificaties.
- 6 Schrijven van één programma-onderdeel dat moet ingepast worden in een groep van interagerende programma-onderdelen die door andere programmeurs werden geschreven en waarvan alleen specificaties en documentatie voorhanden is (maar geen broncode).
- 7 Een programma-onderdeel kunnen debuggen en testen.

Creditcontractvoorwaarde

Toelating tot dit opleidingsonderdeel via creditcontract is mogelijk mits gunstige beoordeling van de competenties

Examencontractvoorwaarde

Dit opleidingsonderdeel kan niet via examencontract gevolgd worden

Didactische werkvormen

Hoorcollege, werkcollege: PC-klasoefeningen

Toelichtingen bij de didactische werkvormen

Hoorcolleges en begeleide practica aan de computer. Forum- en webpaginafaciliteit van de elektronische leeromgeving. Het gevolgde leerboek is ook uitermate geschikt voor (bijkomende) zelfstudie.

Leermateriaal

Programmeren in Java met BlueJ (6de editie), Barnes & Kölling, Pearson Education, ISBN 9789043034999

Prijs: 60 EUR

Daarnaast wordt ook bijkomend materiaal gratis ter beschikking gesteld via de elektronische leeromgeving.

Referenties

Java(TM) Language Specification (Java SE 8 Edition) James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley. Ook online raadpleegbaar.

Vakinhoudelijke studiebegeleiding

Doorlopende begeleiding tijdens de oefeningensessies in de PC-klassen, mogelijkheid tot het stellen van vragen aan de lesgever, mondeling of per e-mail. Forum- en webpaginafaciliteit van de elektronische leeromgeving.

Evaluatiemomenten

periodegebonden evaluatie

Evaluatievormen bij periodegebonden evaluatie in de eerste examenperiode

Schriftelijk examen met open vragen

Evaluatievormen bij periodegebonden evaluatie in de tweede examenperiode

Schriftelijk examen met open vragen

Evaluatievormen bij niet-periodegebonden evaluatie

Tweede examenkans in geval van niet-periodegebonden evaluatie

Niet van toepassing

Eindscoreberekening

100% examen