UNIVERSITEIT GENT

Studiefiche

Academiejaar 2020-2021 t.e.m. 2022-2023

Programmeren (C003080)

Wegens Covid19 kan mogelijk afgeweken worden van de onderwijs- en evaluatievormen. Dergelijke afwijkingen zullen via Ufora worden gecommuniceerd.

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

Studiepunten 5.0 **Studietijd 150 u** **Contacturen** 50.0u

Aanbodsessies en werkvormen in academiejaar 2021-2022

A (semester 1)	Nederlands	Gent	werkcollege: PC-klasoefeningen	30.0u
			online werkcollege: PC-klasoefeningen	0.0u
			hoorcollege	20.0u
			online hoorcollege	0.0u
B (semester 1)	Nederlands	Gent	online hoorcollege	0.0u
			werkcollege: PC-klasoefeningen	30.0u
			hoorcollege	20.0u
			online werkcollege: PC-klasoefeningen	0.0u
C (semester 1)	Engels	Gent	hoorcollege	30.0u
			online werkcollege: PC-klasoefeningen	0.0u
			online hoorcollege	0.0u
			werkcollege: PC-klasoefeningen	30.0u

Lesgevers in academiejaar 2021-2022

Dawyndt, Peter	WE02	Verantwoordelijk lesgever

Aangeboden in onderstaande opleidingen in 2021-2022	**stptn**	**aanbodsessie**
Bachelor of Science in de bestuurskunde en het publiek management	5	A, C
Bachelor of Science in de biochemie en de biotechnologie	5	B
Bachelor of Science in de chemie	5	B
Bachelor of Science in de geografie en de geomatica	5	B
Bachelor of Science in de geologie	5	B
Bachelor of Science in de ingenieurswetenschappen: architectuur	5	A, C
Educatieve Master of Science in de maatschappijwetenschappen(afstudeerrichting communicatiewetenschappen)	5	A
Master of Science in de communicatiewetenschappen(afstudeerrichting nieuwe media en maatschappij)	5	A
Master of Science in de algemene economie	5	A, C
Uitwisselingsprogramma faculteit Wetenschappen (niveau Bachelor)	5	C
Uitwisselingsprogramma politieke en sociale wetenschappen	5	C
Uitwisselingsprogramma wiskunde (niveau master)	5	C
Schakelprogramma tot Master of Science in Chemistry(afstudeerrichting (Bio)Organic and Polymer Chemistry)	5	B
Schakelprogramma tot Master of Science in Chemistry(afstudeerrichting Analytical and Environmental Chemistry)	5	B
Schakelprogramma tot Master of Science in Chemistry(afstudeerrichting Materials and Nano Chemistry)	5	B
Schakelprogramma tot Master of Science in Bioinformatics(afstudeerrichting Systems Biology)	5	B
Schakelprogramma tot Master of Science in de geografie en de geomatica	5	B

Universiteitsbrede keuzevakken	5	A, C
Universiteitsbrede keuzevakken	5	A, C
Universiteitsbrede keuzevakken	5	A, C
Universiteitsbrede keuzevakken	5	A, C
Universiteitsbrede keuzevakken	5	A, C
Universiteitsbrede keuzevakken	5	A, C
Universiteitsbrede keuzevakken	5	A, C
Universiteitsbrede keuzevakken	5	A, C
Universiteitsbrede keuzevakken	5	A, C
Universiteitsbrede keuzevakken	5	A, C
Universiteitsbrede keuzevakken	5	A, C
Voorbereidingsprogramma tot Master of Science in Chemistry(afstudeerrichting Analytical and Environmental Chemistry)	5	B
Preparatory Course Master of Science in Business Engineering	5	C
Voorbereidingsprogramma tot Master of Science in de geografie en de geomatica	5	B

Onderwijstalen

Engels, Nederlands

Trefwoorden

programmeervaardigheden, probleemoplossend denken, programmeertaal Python

Situering

Bij het verwerken en analyseren van informatie met behulp van een computer worden onderzoekers vaak geconfronteerd met tijdrovende en repetitieve taken: verzamelen van gegevens uit webpagina's, omzetten van bestanden naar een ander formaat, analyseren, samenvatten en grafisch voorstellen van gegevens. Door de omvang van de stroom aan nieuwe informatie mag — met het oog op het versnellen van het dagelijkse routinewerk — van een moderne onderzoeker dan ook verwacht worden dat hij of zij deze taken kan automatiseren. In dit opleidingsonderdeel zal je leren hoe je een taak gesteld in natuurlijke taal kunt omzetten naar een programma dat door een computer kan uitgevoerd worden. Hierbij zal je praktische programmeerervaring opdoen door te leren denken in de programmeertaal Python.

Inhoud

Programmeren is het proces om broncode voor computerprogramma's te ontwerpen, schrijven, testen, debuggen en onderhouden. Dit vereist in de eerste plaats kennis van de syntaxis en semantiek van een programmeertaal en de vaardigheid om computerprogramma's in die programmeertaal te schrijven. Als je leert programmeren moet je echter in de eerste plaats ook leren denken als een programmeur. In dit opleidingsonderdeel wordt dan ook heel wat nadruk gelegd op dit denkproces, of anders gezegd: programmeren wordt gezien als een manier om probleemoplossend te leren denken. In het bijzonder zullen we gebruik maken van de programmeertaal Python om oplossingen voor problemen te leren bedenken in termen van

- basiscomponenten: instructies, variabelen, gegevenstypes en operatoren
- controlestructuren: voorwaardelijke opdrachten, controlelussen en functies
- datastructuren: strings, lijsten, tuples, dictionaries, verzamelingen, bestanden en modules
- objectgericht programmeren: objecten, klassen, attributen, methoden, overerving, polymorfisme en uitzonderingen

Via programmavoorbeelden zal aangetoond worden dat de aangeleerde programmeervaardigheden ook generiek kunnen toegepast worden in andere programmeertalen.

Begincompetenties

Enige vertrouwdheid met de computer is gewenst. Het is echter helemaal niet nodig om voorafgaande programmeerervaring te hebben.

Eindcompetenties

1 Een taak omschreven in natuurlijke taal kunnen omzetten naar een programma in de programmeertaal Python en in staat zijn om dit programma door een computer te laten uitvoeren om een correct resultaat te genereren.

2 Een programma kunnen testen en debuggen.
3 Goede keuzes kunnen maken tussen verschillende alternatieven bij het implementeren van een programma, rekening houdend met efficiëntie, helderheid en oog voor detail.
4 Praktische kennis hebben van de basisprincipes van objectgericht programmeren.

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

Online hoorcollege, Online werkcollege: pc-klasoefeningen, Hoorcollege, Zelfstandig werk, Werkcollege: pc-klasoefeningen

Toelichtingen bij de didactische werkvormen

Interactieve hoorcolleges en begeleide oefeningen aan de computer in de PC-klas. Gebruik van de elektronische leeromgeving Pythia (pythia.ugent.be) voor automatische evaluatie van programmeeroefeningen. Interactieve faciliteiten van de elektronische leeromgeving Ufora (Ufora.ugent.be). Het gevolgde leerboek is ook uitermate geschikt voor (bijkomende) zelfstudie.

Leermateriaal

Handboek: The Practice of Computing using Python (derde editie)
Auteurs: William Punch, Richard Enbody
Uitgeverij: Pearson
Taal: Engels
ISBN-13: 978-1292166629
Richtprijs: €70

De presentaties die tijdens de hoorcolleges getoond worden, worden samen met bijkomend leermateriaal (achtergronddocumenten en koppelingen naar relevante webpagina's) ter beschikking gesteld via de elektronische leeromgeving Ufora.

Referenties

- Mark Lutz (2009). Learning Python: Powerful Object-Oriented Programming (4th edition). O'Reilly Media, ISBN-13: 978-0596158064.
- Mark Pilgrim (2009). Dive Into Python. CreateSpace, ISBN-13: 978-1441413024. (free download @ http://diveintopython.org)
- Hans Peter Langtangen (2009). A Primer on Scientific Programming with Python. Springer, ISBN-13: 978-3642024740.
- Tony Gaddis (2009). Starting Out with Python. Pearson Eduction - Addison Wesley, ISBN-13: 978-0321549419.
- Michael H. Goldwasser (2007). Object-Oriented Programming in Python. Prentice Hall, ISBN-13: 978-0136150312.
- Jason Kinser (2008). Python For Bioinformatics. Jones & Bartlett Publishers, ISBN-13: 978-0763751869.
- Sebastian Bassi (2009). Python for Bioinformatics. Chapman & Hall, ISBN-13: 978-1584889298.

Vakinhoudelijke studiebegeleiding

Syntaxis en semantiek van de programmeertaal Python worden aangebracht in het handboek en de slides, en moeten grotendeels via zelfstudie verworven worden. Tijdens de hoorcolleges worden voorbeelden uitgewerkt hoe de programmeertechnieken in praktijk kunnen gebracht worden. Tijdens de werkcolleges leren studenten al doende zelf computerprogramma's schrijven op basis van een reeks opgelegde oefeningen, die daarna zelfstandig verder moeten uitgewerkt worden. Via een elektronische leeromgeving krijgen studenten ogenblikkelijk feedback op de door hen ingediende programmeeroefeningen, en worden bijkomende opgaven voor vrij oefenen aangeboden. Gebruik van het forum in de elektronische leeromgeving Ufora wordt gestimuleerd, waardoor studenten, begeleiders en lesgever elkaar buiten de contacturen kunnen helpen bij het omzetten van taken naar computerprogramma's. Niet-periodegebonden evaluaties van opgelegde oefeningen en evaluatie-oefeningen worden tegelijkertijd ook gebruikt als feedbackmoment, en na de wekelijkse deadlines en de evaluaties worden voorbeeldoplossingen van de opgelegde oefeningen ter beschikking gesteld.

Evaluatiemomenten

periodegebonden en niet-periodegebonden evaluatie

Evaluatievormen bij periodegebonden evaluatie in de eerste examenperiode

Vaardigheidstest, Openboekexamen

Evaluatievormen bij periodegebonden evaluatie in de tweede examenperiode

Vaardigheidstest, Openboekexamen

Evaluatievormen bij niet-periodegebonden evaluatie

Vaardigheidstest, Openboekexamen

Tweede examenkans in geval van niet-periodegebonden evaluatie

Examen in de tweede examenperiode is niet mogelijk

Toelichtingen bij de evaluatievormen

De punten worden voor 20% (4/20) berekend op basis van de niet-periodegebonden evaluatie (dagelijks werk) en voor 80% (16/20) op basis van de periodegebonden evaluatie (examen). De niet-periodegebonden evaluatie (dagelijks werk) bestaat uit twee componenten die samen de score van de niet-periodegebonden evaluatie bepalen.

Voor het eerste deel van de niet-periodegebonden evaluatie krijgen de studenten een reeks van 60 oefeningen opgelegd. Op basis van de behandelde programmeertechnieken worden deze oefeningen onderverdeeld in 10 reeksen, waarbij elke reeks bestaat uit 6 opgelegde oefeningen. De eerste oefening van elke reeks is telkens een variant op het werken met ISBN-nummers. Een voorbeeldoplossing van deze oefening is gegeven op Ufora, en in een bijhorende instructievideo wordt uitgelegd hoe we tot deze voorbeeldoplossing gekomen zijn. Met deze ISBN-oefening zetten we uiteen hoe de nieuw aangebrachte programmeertechniek van de oefeningenreeks in de praktijk kan gebracht worden. Met deze voorbereiding kunnen de studenten zelf aan de slag om de programmeertechniek in de praktijk te brengen via het oplossen van de andere vijf opgelegde oefeningen uit de reeks. Studenten moeten hun oplossingen voor de opgelegde oefeningen van elke reeks (inclusief de ISBN-oefening) indienen via het online leerplatform Pythia tegen vooraf vastgelegde deadlines (telkens op de dinsdag om 22:00 volgend op de week van het werkcollege gewijd aan de oefeningenreeks). Studenten kunnen gebruik maken van de Pythia scoresheet om op te volgen voor welke oefeningen ze reeds een correcte oplossing hebben ingediend. De scoresheet geeft immers een handig overzicht van de opgelegde oefeningen uit de opeenvolgende reeksen, de bijhorende deadlines en de tijd waarop voor het eerste een correcte oplossing voor de oefening werd ingediend.

Voor het tweede deel van de niet-periodegebonden evaluatie organiseren we twee evaluatiemomenten tijdens de werkcolleges die volgen op het afwerken van vijf oefeningenreeksen. Bij deze evaluaties krijgen de studenten telkens twee uur de tijd om twee nieuwe programmeeroefeningen op te lossen. Hierbij kunnen ze gebruik maken van het online leerplatform Pythia om de correctheid van hun oplossingen na te gaan. De ingediende oplossingen van deze evaluatie-oefeningen worden echter met de hand nagekeken door de lesgever en zijn begeleiders, en beoordeeld op correctheid, gebruikte programmeerstijl, keuzes gemaakt bij het gebruik van de verschillende programmeertechnieken en de kwaliteit van de oplossingsmethode. De moeilijkheidsgraad van deze evaluatie-oefeningen is lager dan deze van de oefeningen die tijdens de periodegebonden evaluatie (het examen) worden voorgelegd, aangezien we op dit tijdstip van het semester voornamelijk willen nagaan of de studenten de basisvaardigheden van het programmeren beheersen. Bovendien volgen deze evaluatiemomenten dezelfde procedure die gebruikt wordt tijdens de periodegebonden evaluatie, zodat studenten op basis van hun ervaring eventueel hun werkmethode kunnen bijsturen in voorbereiding naar het examen.

De score van de niet-periodegebonden evaluatie wordt bepaald volgens de formule s * c / a. Hierbij staat s voor de score die de studenten behaald hebben op basis van hun ingediende evaluatie-oefeningen (uitgedrukt op 20), c staat voor het aantal opgelegde oefeningen dat correct werd ingediend tegen de wekelijkse deadlines, en a staat voor het aantal opgelegde oefeningen (30 per evaluatiereeks). Een student die bijvoorbeeld 16/20 behaald heeft voor zijn evaluatie-oefeningen en alle 30 opgelegde oefeningen correct heeft ingediend voorafgaand aan de wekelijkse deadlines, krijgt voor die evaluatiereeks een score van 16 * 30/30 = 16 op 20. Als die student echter nog steeds 16/20 had behaald voor zijn evaluatie-oefeningen, maar slechts 18/30 opgelegde oefeningen correct had ingediend voorafgaand aan de wekelijkse deadlines, dan houdt die student slechts een score van 16 * 18/30 = 9.6 op 20 over.

Studenten krijgen zo snel mogelijk na een evaluatie per email hun score toegestuurd. Tijdens het daaropvolgende werkcollege krijgen de studenten ook de oplossingen die ze ingediend

hebben tijdens de evaluatie terug, voorzien van feedback die aangeeft waar er verbeterpunten zijn die ze kunnen meenemen bij het verder oplossen van oefeningen of naar het examen.

De niet-periodegebonden evaluatie kan niet hernomen worden tijdens de tweede examenperiode. Voor de tweede examenperiode wordt enerzijds een score berekend rekening houdend met de punten behaald voor de niet-periodegebonden evaluatie (dus punten dagelijks werk gequoteerd op 4 samen met het examen gequoteerd op 16, zoals dat ook geldt voor de eerste examenperiode) en wordt anderzijds ook een score berekend zonder rekening te houden met de punten behaald voor de niet-periodegebonden evaluatie (examenresultaat rechtstreeks gequoteerd op 20). De eindscore voor de tweede examenperiode is het maximum van de voorgaande twee berekeningen.

Tijdens de periodegebonden evaluatie (examen) krijgen de studenten 3.5 uur om drie programmeeroefeningen op te lossen. Daarvoor kunnen de studenten opnieuw gebruik maken van de Pythia omgeving om hun oplossingen in te dienen en feedback te krijgen op de correctheid en de programmeerstijl ervan. Voor het bepalen van de examenscore worden de ingediende oplossingen evenwel opnieuw met de hand nagekeken en beoordeeld.

Eindscoreberekening

De punten worden voor 20% (4/20) berekend op basis van de niet-periodegebonden evaluatie (dagelijks werk) en voor 80% (16/20) op basis van de periodegebonden evaluatie (examen).

De niet-periodegebonden evaluatie kan niet hernomen worden tijdens de tweede examenperiode. Voor de tweede examenperiode wordt enerzijds een score berekend rekening houdend met de punten behaald voor de niet-periodegebonden evaluatie (dus punten dagelijks werk gequoteerd op 4 samen met het examen gequoteerd op 16, zoals dat ook geldt voor de eerste examenperiode) en wordt anderzijds ook een score berekend zonder rekening te houden met de punten behaald voor de niet-periodegebonden evaluatie (examenresultaat rechtstreeks gequoteerd op 20). De eindscore voor de tweede examenperiode is het maximum van de voorgaande twee berekeningen.