



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

Studiepunten 6.0 Studietijd 180 u Contacturen 60.0 u

Aanbodsessies en werkvormen in academiejaar 2018-2019

A (semester 2)	Nederlands	practicum	15.0 u
		werkcollege: geleide	5.0 u
		hoorcollege	20.0 u

Lesgevers in academiejaar 2018-2019

De Sutter, Bjorn	TW06	Verantwoordelijk lesgever
------------------	------	---------------------------

Aangeboden in onderstaande opleidingen in 2018-2019

	stptn	aanbodsessie
Master of Science in de informatica	6	A
Master of Science in de industriële wetenschappen: informatica	6	A
Master of Science in de ingenieurswetenschappen: computerwetenschappen	6	A
Master of Science in Computer Science Engineering	6	A

Onderwijstalen

Nederlands

Trefwoorden

contextvrije grammatica, afleidingsboom, top-down en bottom-up parsers, abstracte syntax bomen, semantische analyse, basisblokken, instructieselectie, registertoewijzing, dataflow analyse, optimaliserende codegeneratie

Situering

Een computer werkt met machinetaal terwijl een programmeur schrijft in een hogere programmeertaal. Compilers vormen de verbinding tussen de programmeur en de computer. Een compiler zal het bronprogramma op een correcte en efficiënte manier vertalen naar machine-instructies. Dit gebeurt in twee grote stappen: in het front-end gedeelte worden lexicaal en parsing technieken gebruikt om het programma om te vormen naar een intermediaire voorstelling. In het back-end gedeelte worden verschillende optimalisatietechnieken toegepast om het programma zo goed mogelijk af te beelden op de onderliggende architectuur. In dit vak worden de theoretische concepten gekoppeld aan de implementatie van een mini-compiler. De doelstellingen zijn: een goed begrip krijgen van de bouw en de werking van lexicaal analyzers en parsers, een inzicht verwerven in de kenmerken van een geschikte intermediaire voorstelling, de vaardigheden ontwikkelen om de complexiteit van een compilerproject te beheersen en ervaring opdoen met de implementatie en evaluatie van optimaliserende codegeneratie.

Inhoud

- Inleiding: Datastructuren voor compilers, Overzicht van fasen in de compilatie
- Lexicale analyse: Tokens en reguliere expressies, Eindige automaten
- Parsing: Contextvrije grammatica, Predictieve en LR-parsers
- Abstracte syntax: Semantische acties, Abstracte syntaxboom
- Semantische analyse: Symbooltabellen, Typechecking
- Activatierecords: Stapelvensters en locale variabelen, Doorgeven van parameters
- Intermediaire code: Voorstellingsboom, Vertaling
- Basisblokken en sporen: Canonische bomen, Voorwaardelijke sprongen
- Instructieselectie: Algoritmen voor instructieselectie
- Liveness analyse en registerallocatie: Levensduurvergelijkingen, Graafkleuring en registertoewijzing

- Datastroomanalyse: Datastructuren voor datastroomvoorstelling, Datastroomalgoritmen
- Lusoïptimisaties: Reduceerbare grafen, Lustransformaties
- Compiler implementatie: Parsing, Codegeneratie, Code optiïmisatie

Begincompetenties

Kennis van progammeertalen, programmeren in C

Eindcompetenties

- 1 De betekenis van de verschillende fasen in een compiler begrijpen.
- 2 Inzicht hebben in de constructie van automata voor de generatie van lexicale analyzers.
- 3 Het gebruik van verschillende parsertechnieken (o.m. LL, LR) beheersen.
- 4 De software voor de generatie van lexicale analyzers en grammaticale parsers gebruiken.
- 5 Typechecking inbouwen.
- 6 Een abstracte syntaxboom interpreteren.
- 7 Controle- en datastroomgrafen opbouwen.
- 8 Afhankelijkheidsanalyse gebruiken voor levensduur-analyse en code-optiïmisatie.
- 9 Instructieselectie-algoritmen vergelijken en toepassen.
- 10 Registers toewijzen.
- 11 Intermediaire voorstellingen hanteren voor de generatie van machine-onafhankelijke code.
- 12 Lustransformaties analyseren voor code-optiïmisatie.
- 13 Dataverloop analyseren en code transformeren voor optimalisatie.

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, practicum, werkcollege: geleide oefeningen

Toelichtingen bij de didactische werkvormen

De studenten krijgen programmeeropdrachten die ze deels in begeleide sessies in de PC-lassen, deels thuis moeten uitvoeren. Ze moeten hun software, alsook soms een klein verslag, indienen. Voor elke opdracht kan een aantal studenten gekozen worden die ook mondeling bepaalde aspecten van de ingediende code dienen te verduidelijken, tijdens een volgende sessie in de PC-klas.

Leermateriaal

Modern compiler implementation in C, Andrew W. Appel with Maia Ginsburg, Cambridge University Press, ISBN 052158390X, LCCN QA76.73.C15.A63, prijs: 51 Euro

Referenties

- Modern compiler implementation in C, Andrew W. Appel with Maia Ginsburg, Cambridge University Press, ISBN 052158390X, LCCN QA76.73.C15.A63
- JFLAP: An Interactive Formal Languages and Automata Package, Susan H. Rodger en Thomas W. Finley, Jones and Bartlett Publishers, ISBN 0763738344, LCCN QA267.3.R63
- The Definitive ANTLR Reference: Building Domain-Specific Languages, Terence Parr, The Pragmatic Bookshelf, ISBN 0978739256, LCCN QA76.7.P37
- Principles of compiler design, Alfred V. Aho, Jeffrey D. Ullman, Addison-Wesley, ISBN 0201000229, LCCN QA76.6.A365

Vakinhoudelijke studiebegeleiding

Evaluatiemomenten

periodegebonden en niet-periodegebonden evaluatie

Evaluatievormen bij periodegebonden evaluatie in de eerste examenperiode

Schriftelijk examen, openboekexamen

Evaluatievormen bij periodegebonden evaluatie in de tweede examenperiode

Schriftelijk examen, openboekexamen

Evaluatievormen bij niet-periodegebonden evaluatie

Mondeling examen, werkstuk, verslag

Tweede examenkans in geval van niet-periodegebonden evaluatie

Examen in de tweede examenperiode is enkel mogelijk in gewijzigde vorm

Toelichtingen bij de evaluatievormen

De niet-periodegebonden evaluatie bestaat uit een zestal werkstukken met verslagen die gespreid zijn over het semester. Het verslag van elk werkstuk moet voor de start van het volgende practicum of op een opgegeven datum ingediend worden. Na het indienen van het verslag volgt er mogelijk ook een mondelinge evaluatie waarop de student ondervraagd wordt over de gebruikte methodiek en de behaalde resultaten. Voor de werkstukken werken de studenten in groepjes van twee. Aan de hand van een peer assessment over hun bijdrage aan het werkstuk en verslag, en eventueel na een gesprek hierover bij tegenstrijdige assessments, wordt bepaald of beide studenten al dan niet dezelfde punten krijgen.

Niet op tijd ingediende verslagen zonder geldige reden van afwezigheid (zoals een doktersbriefje) worden met een nul gekwoteerd voor dat onderdeel van de niet-periodegebonden evaluatie.

De niet-periodegebonden evaluatie verloopt gelijkaardig voor de tweede examenperiode. De student wordt dan geëvalueerd op basis van nieuwe werkstukken met een gelijkaardige belasting als in de eerste periode. De verslagen ervan zullen een week voor het examen ingediend moeten worden. Vlak na dat schriftelijk periodegebonden examen volgt dan een mondelinge evaluatie over de werkstukken. De periodegebonden evaluatie is schriftelijk en omvat een oefeningengedeelte met open boek.

Eindscoreberekening

Verdeling van de punten: 3/5 voor de periodegebonden evaluatie en 2/5 voor de niet-periodegebonden evaluatie. Speciale voorwaarde: men moet minstens 10/20 halen op de niet-periodegebonden evaluatie en op de periodegebonden evaluatie om voor dit opleidingsonderdeel te kunnen slagen. Indien dit niet het geval is voor één of beide delen van de score, wordt dat deel de eindscore.