UNIVERSITEIT GENT

Studiefiche

Vanaf academiejaar 2026-2027

Parallelle en gedistribueerde softwaresystemen (E017930)

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

Studiepunten 6.0 **Studietijd 180 u**

Aanbodsessies en werkvormen in academiejaar 2026-2027

A (semester 1)	Engels	Gent	zelfstandig werk hoorcollege werkcollege
B (semester 1)	Nederlands	Gent	

Lesgevers in academiejaar 2026-2027

De Turck, Filip	TWO5	Verantwoordelijk lesgever
Fostier, Jan	TWO5	Medelesgever

Aangeboden in onderstaande opleidingen in 2026-2027	**stptn**	**aanbodsessie**
Educatieve Master of Science in de wetenschappen en technologie(afstudeerrichting informatica)	6	A
Brugprogramma Master of Science in Bioinformatics(afstudeerrichting Engineering)	6	A
Brugprogramma Master of Science in Computer Science Engineering	6	A
Master of Science in Electrical Engineering(afstudeerrichting Communication and Information Technology)	6	A
Master of Science in Electromechanical Engineering(afstudeerrichting Control Engineering and Automation)	6	A
Master of Science in Electromechanical Engineering(afstudeerrichting Electrical Power Engineering)	6	A
Master of Science in Bioinformatics(afstudeerrichting Engineering)	6	A
Master of Science in Electromechanical Engineering(afstudeerrichting Maritime Engineering)	6	A
Master of Science in Electromechanical Engineering(afstudeerrichting Mechanical Construction)	6	A
Master of Science in Electromechanical Engineering(afstudeerrichting Mechanical Energy Engineering)	6	A
Master of Science in Computer Science Engineering	6	A
Master of Science in de informatica	6	A
Master of Science in de ingenieurswetenschappen: computerwetenschappen	6	B
Uitwisselingsprogramma informatica (niveau master)	6	A

Onderwijstalen

Engels, Nederlands

Trefwoorden

Parallelle en gedistribueerde software, communicatie, coördinatie, synchronisatie, efficiëntie, programmeermodellen, hoog-performant rekenen, belangrijke software-architecturen, fouttolerantie.

Situering

Deze cursus behandelt de verschillende aspecten van het ontwerp en de implementatie van parallelle en gedistribueerde software. De studenten krijgen een state-of-the-art overzicht van parallelle en gedistribueerde software systemen, en leren verschillende algoritmen en software-engineeringtechnieken voor deze belangrijke klasse van software-systemen. De nadruk ligt op algoritmische en softwarematige aspecten, met aandacht voor de impact van hardware-

architecturen op performantie.

Inhoud

- Basisbegrippen, definities, classificaties.
- Performantiemetrieken en beperkende factoren: versnelling, efficiëntie, schaalbaarheid, beschikbaarheid, Amdahl's en Gustafson's wet, CAP-theorema, netwerkmodellen, falen van nodes.
- Gedistribueerde software: message passing, RPC, RMI, marshalling, message-oriented middleware en services.
- Algoritmen voor coördinatie, consensus en anonieme netwerken.
- Identificeren en definiëren van functionele vereisten en niet-functionele vereisten voor gedistribueerde software systemen.
- Ontwerp van een softwarearchitectuur die voldoet aan de vooropgestelde kwaliteitsvereisten met behulp van de Attribute-Driven Designmethodologie en het gebruik van softwarepatronen en ontwerpstactieken.
- Cloudcomputingmodellen: service-, deployment- en betalingsmodellen; cloudplatforms en programmeermodellen. [enkel voor studenten Master Informatica]
- Microservices en containers [enkel voor studenten Master Informatica]
- Hoog-performant gedistribueerd rekenen: MPI, communicatiepatronen, probleemdecompositie, casestudies.
- Gedeeld-geheugenprogrammering: data-consistentie, concurrency, synchronisatie, deadlocks, thread-safety.
- Monitoring en beheer van grootschalige systemen: metingen, autonoom beheer, control-theory-gebaseerd beheer.
- Acceleratoren voor artificiële intelligentie: kernels, thread- en geheugenhiërarchie, control flow.

Begincompetenties

- Basiskennis programmeren in C/C++.
- Basiskennis van datastructuren en algoritmen.
- Basiskennis van besturingssystemen.

Eindcompetenties

1 De belangrijkste algoritmische problemen in parallelle en gedistribueerde software herkennen en begrijpen.
2 De functies van middleware en de voornaamste architecturen kennen voor parallelle en gedistribueerde systemen.
3 De verschillende parallelle en gedistribueerde softwaremodellen kunnen verklaren.
4 Basisstrategieën kunnen toepassen om algoritmische problemen op te lossen.
5 Het vertalen van de vereisten van een softwareproject naar technische en niet-technische vereisten en het ontwerp van een softwarearchitectuur die aan deze specificaties voldoet.
6 Een basisontwerp voor parallelle of gedistribueerde applicaties kunnen ontwikkelen en de performantie inschatten.
7 Rekening houden met schaalbaarheid en performantie bij ontwerpbeslissingen.
8 Algoritmes voor standaardproblemen kunnen evalueren en toepassen.
9 Alternatieven kritisch afwegen vóór implementatie.

Creditcontractvoorwaarde

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

Examencontractvoorwaarde

Dit opleidingsonderdeel kan niet via examencontract gevolgd worden

Didactische werkvormen

Werkcollege, Hoorcollege, Zelfstandig werk

Toelichtingen bij de didactische werkvormen

De theoretische concepten worden behandeld in hoorcolleges. In werkcolleges passen studenten deze concepten toe in programmeeroefeningen en casestudies. Zelfstandig werk omvat voorbereidende taken en opdrachten, o.a. huistaken.

Studiemateriaal

Type: Andere

Naam: Aangewezen literatuur en online referenties
Richtprijs: Gratis of betaald door opleiding
Optioneel: nee
Bijkomende info: Er zijn geen verplichte handboeken met kostprijs. Gebruik van aanbevolen literatuur en online referenties via het elektronisch leerplatform.

Referenties

- George Coulouris et al., Distributed Systems: Concepts and Design (5th ed.), Pearson.
- Rajkumar Buyya et al., Cloud Computing: Principles and Paradigms, Wiley.
- Peter Pacheco, An Introduction to Parallel Programming, Morgan Kaufmann.
- Ian Fostier, Designing and Building Parallel Programs, Addison-Wesley.
- Online referenties: Cloud programming, MPI: The Complete Reference.

Vakinhoudelijke studiebegeleiding

Practica worden begeleid door assistenten. Bijkomende informatie wordt voorzien via het elektronische leerplatform.

Evaluatiemomenten

periodegebonden en niet-periodegebonden evaluatie

Evaluatievormen bij periodegebonden evaluatie in de eerste examenperiode

Schriftelijke evaluatie

Evaluatievormen bij periodegebonden evaluatie in de tweede examenperiode

Schriftelijke evaluatie

Evaluatievormen bij niet-periodegebonden evaluatie

Vaardigheidstest, Participatie, Werkstuk

Tweede examenkans in geval van niet-periodegebonden evaluatie

Examen in de tweede examenperiode is niet mogelijk

Toelichtingen bij de evaluatievormen

Periodegebonden evaluatie: schriftelijk examen met gesloten boek.
Niet-periodegebonden evaluatie: beoordeling van huistaken.

Eindscoreberekening

75% examen
25% huistaken
Indien het examencijfer lager is dan 8/20, bedraagt de eindscore maximaal 8/20.
In de tweede examenperiode worden punten voor practica en huistaken enkel meegerekend indien ze de eindscore verhogen.