

RTU studiju kurss "Programmēšanas valodas"

02C60 Rīgas Biznesa skola

Vispārējā informācija

Kods	BS0030
Nosaukums	Programmēšanas valodas
Studiju kursa statuss programmā	Obligātais/Ierobežotās izvēles
Atbildīgais mācītspēks	Jānis Lazovskis - Doktors, Docents
Apjoms daļās un kredītpunktos	1 daļa, 8.0 kredītpunkti
Studiju kursa īstenošanas valodas	EN
Anotācija	Studiju kurss iepazīstina studentus ar dažādiem programmēšanas veidiem, tiem atbilstošām programmēšanas valodām un tajās izmantotajām fundamentālajām programmu veidošanas konstrukcijām, tādējādi bagātinot tālākās mācībās un darbā izmantojamo valodu līdzekļu klāstu.
Mērķis un uzdevumi, izteikti kompetencēs un prasmēs	Studiju kursa galvenais mērķis ir rosināt studentus apgūt programmēšanas valodu fundamentālos jēdzienus un droši pielietot valodu rīkus gan pašmācībā, gan ikdienas darbos. Studiju kursa uzdevumi ir: - iepazīstināt studentus ar programmēšanas valodas jēdzienu un sniegt priekšstatu par programmēšanas valodas apraksta līdzekļiem: sintaksi un semantiku; - sniegt priekšstatu par dažādiem fundamentāliem programmēšanas veidiem (paradigmām) - imperatīvo, funkcionālo, loģisko, objekt-orientēto, paralēlo un uz kopām balstīto, tiem atbilstošajām programmēšanas valodām un tajās izmantotajām galvenajām programmu veidošanas konstrukcijām, kā arī pamata pieejām uzdevumu risināšanai; - ļaut izmēģināt dažādas programmēšanas paradigmas un valodas darbībā, veidojot vienkāršas programmas.
Patstāvīgais darbs, tā organizācija un uzdevumi	Studenti patstāvīgi analizē studiju kursa literatūru, gatavoja praktiskajiem darbiem, pārbaudes darbiem un eksāmeniem, veic mājasdarbus.
Literatūra	Obligātā/Obligatory: 1. Sebesta, Robert W. Concepts of Programming Languages (11th ed.). Addison-Wesley, ISBN-13: 978-1-292-10055-5, 2015, 800 p. Papildu/Additional: 1. Louden, K.C., Lambert, K.A., Programming Languages. Principles and Practice. 3rd Edition. Cengage Learning 2012, 704 p. 2. Nielson, H.R., Nielson, F., Semantics with Applications. An Appetizer. Springer Undergraduate Topics in Computer Science. 2007, 274 p. Citi informācijas avoti/Other sources of information: 1. An Example-Based Introduction to Finite-State Machines, https://levelup.gitconnected.com/an-example-based-introduction-to-finite-state-machines-f908858e450f 2. Download GNAT Community Edition. https://www.adacore.com/download 3. Finite State Machines and Regular Expressions, https://www.gamedev.net/tutorials/_/technical/general-programming/finite-state-machines-and-regular-expressions-r3176/ 4. Haskell. An advanced, purely functional programming language. https://www.haskell.org/ 5. Nielson, H.R., Introduction to Semantics, http://www.imm.dtu.dk/courses/02240/lecture1A.pdf 6. Racket. https://racket-lang.org/ 7. SWI Prolog. https://www.swi-prolog.org/ 8. SQL Tutorial, https://www.w3schools.com/sql/ 9. Welcome to Squeak/Smalltalk. https://squeak.org/ 10. Concepts of Programming Languages" (10th ed.) by Robert W. Sebesta. Addison-Wesley, ISBN-13: 978-0131395312
Nepieciešamās priekšzināšanas	Datu struktūras.

Studiju kursa saturs

Saturs	Pilna un nepilna laika klātienē studijas		Nepilna laika neklātienē studijas	
	Kontakt stundas	Patstāv. darbs	Kontakt stundas	Patstāv. darbs
Programmēšanas valodas jēdziens un programmēšanas valodu vēsturiskā attīstība.	10	10	0	0
Sintakse un semantika: abstraktās un konkrētās sintakses jēdzieni, gramatikas, sintakses analīze, dabiskā, operacionālā un aksiomātiskā semantika.	10	10	0	0
Funkcionālā programmēšana. Haskell. Scheme.	9	10	0	0
Loģiskā programmēšana. Prolog.	8	10	0	0
Imperatīvā un objekt-orientētā programmēšana. C, C++, C#, Smalltalk.	12	10	0	0
Vārdi un to atribūti: piesaiste, darbības apgabali, tipu pārbaude.	8	10	0	0
Datu tipi programmēšanas valodās.	9	10	0	0
Operatoru un procedūru līmeņa vadības struktūras.	12	10	0	0
Paralēlā programmēšana: pamata konstrukcijas un to pieejamība programmēšanas valodās.	10	10	0	0
Datubāzu programmēšana. SQL.	12	10	0	0

Kopā:	100	100	0	0
-------	-----	-----	---	---

Sasniedzamie studiju rezultāti un to vērtēšana

Sasniedzamie studiju rezultāti	Rezultātu vērtēšanas metodes
Spēj izskaidrot programmēšanas valodas jēdzienu un raksturo programmēšanas valodu attīstības vēstures posmus.	Starpeksāmens, eksāmens.
Prot raksturot dažādas programmēšanas paradigmas, to galvenās iezīmes.	Starpeksāmens, eksāmens.
Prot izskaidrot programmēšanas valodu konstrukcijas un to izmantošanas iespējas programmēšanas uzdevumu risināšanā.	Starpeksāmens, eksāmens.
Spēj izskaidrot programmēšanas valodu sintakses un semantikas jēdzienus, galvenos sintakses un semantikas uzdošanas veidus.	Starpeksāmens, eksāmens.
Izprot un veido sintakses un semantikas aprakstus programmēšanas valodu konstrukcijām.	Mājasdarbi, ikdienas pārbaudījumi, eksāmens.
Prot veidot vienkāršas programmas, izmantojot dažādas programmēšanas valodas un to konstrukcijas.	Mājasdarbi, ikdienas pārbaudījumi, eksāmens.
Spēj salīdzināt dažādu programmēšanas valodu īpašības.	Mājasdarbi, ikdienas pārbaudījumi, eksāmens.

Studiju rezultātu vērtēšanas kritēriji

Kritērijs	% no kopējā vērtējuma
Mājasdarbi	70
Starpeksāmens	8
Eksāmens	10
Ikdienas pārbaudījumi	12
Kopā:	100

Studiju kursa plānojums

Daļa	KP	Stundas			Pārbaudījumi		
		Lekcijas	Prakt d.	Laborat	Ieskaite	Eksām.	Darbs
1.	8.0	40.0	60.0	0.0		*	