

RTU studiju kurss "Objektorientētā programmēšana"

33000 Datorzinātnes, informācijas tehnoloģijas un enerģētikas fakultāte

Vispārējā informācija

Kods	DPI230
Nosaukums	Objektorientētā programmēšana
Studiju kursa statuss programmā	Obligātais/Ierobežotās izvēles
Atbildīgais mācītbspēks	Uldis Sukovskis - Doktors, Vecākais pasniedzējs
Mācītbspēks	Pāvels Rusakovs - Doktors, Docents Gundars Alksnis - Doktors, Asociētais profesors Ojārs Bāliņš - Docētājs Igoris Ščukins - Doktors, Docents Kristaps Rāvalds - Pasniedzējs
Apjoms daļās un kredītpunktos	1 daļa, 4.5 kredītpunkti
Studiju kursa īstenošanas valodas	LV, EN
Anotācija	Studiju kursā tiek apgūta objektorientētās programmēšanas (OOP) koncepcija, izmantojot valodu C++ kā piemēru. Tiek apskatītas OOP pamatkonceptijas – abstrakcija, iekapsulēšana, mantošana, polimorfisms u.c. Tiek apgūta klašu hierarhijas veidošana, virtuālo funkciju izmantošana, izņēmuma situāciju apstrāde, šablonu izmantošana, kā arī praktiskās programmēšanas jautājumi, tajā skaitā grafiskā lietotāja interfeisa veidošanas pamati integrētajā programmatūras izstrādes vidē. Būtisku studiju kursa daļu veido laboratorijas darbu izpilde, kuros studenti izmanto lekcijās iegūtās zināšanas un nostiprina OOP prasmes. Pēc sekmīgas šā studiju kursa apgūšanas studentiem jāizprot OOP pamatprincipi un jāspēj izveidot programmatūras risinājumu OOP valodā.
Mērķis un uzdevumi, izteikti kompetencēs un prasmēs	Studiju kursa mērķis ir apgūt OOP paradigmu un tās praktisko lietojumu, lai studiju kursa nobeigumā studenti būtu kompetenti to izmantot programmatūras izstrādē un uzturēšanā. Studiju kursa galvenie uzdevumi: - izskaidrot OOP pamatkonceptijas (abstrakcija, iekapsulēšana, mantošana, modularitāte, polimorfisms u.c.); - nostiprināt prasmes izprast un pilnveidot esošu OOP pirmkodu; - attīstīt prasmes risināt uzdevumus, izmantojot OOP paradigmas realizāciju valodā C++; - nostiprināt programmatūras izstrādes un atklādošās rīku lietošanas prasmes.
Patstāvīgais darbs, tā organizācija un uzdevumi	Patstāvīgo darbu studenti veic ārpus nodarbību laika, izmantojot savus personīgos datorus vai datorklasē. Darbs ar literatūras avotiem un lekciju materiāliem. Lekcijā apskatīto pirmkoda piemēru patstāvīga kompilēšana, pilnveidošana, atklādošana un izpilde. Laboratorijas darbu individuālo uzdevumu sagatavošana un gatavošanās to aizstāvēšanai.
Literatūra	Obligātā / Obligatory: 1. C++ Get Started! / Internet: https://isocpp.org/get-started 2. Bjarne Stroustrup. A Tour of C++ Addison-Wesley, 2014 Papildu / Additional: 1. Ray Lischner. Exploring C++20, Apress, 2020 2. J. Burton Browning, Bruce Sutherland, "C++20 Recipes", Apress, 2020 3. Scott Meyers. Effective Modern C++ O'Reilly Media, 2014 4. Bjarne Stroustrup. The C++ Programming Language 4th Edition Addison-Wesley, 2013 5. Grady Booch, etc. Object-Oriented Analysis and Design with Applications 3rd Ed. Addison-Wesley, 2007 Citi / Other: 1. RTU E-library's ProQuest Ebook Central (keywords: C++; object-oriented programming) 2. Online Videos, e.g.: https://www.youtube.com/playlist?list=PLlRAtfBNZ98dudnM48yFGUldqGD0S4FFb , https://www.youtube.com/playlist?list=PL43pGnjiVwgQHLpnuH9ch-LhZdwcM8Tq
Nepieciešamās priekšzināšanas	Pamatzināšanas un prasmes strukturālajā programmēšanā (vēlams C saimes programmēšanas valodā).

Studiju kursa saturs

Saturs	Pilna un nepilna laika klātienes studijas		Nepilna laika neklātienes studijas	
	Kontakt stundas	Patstāv. darbs	Kontakt stundas	Patstāv. darbs
Ievads. Objektorientētas programmas piemērs. Darbs ar Microsoft Visual Studio. C++ preprocesors.	2	2	0	0
Objektorientētās programmēšanas pamatjēdzieni - abstrakcija, iekapsulēšana, mantošana un polimorfisms u.c. Klases. Klases locekļu iekapsulēšana.	2	2	0	0
Iegultās (inline) metodes. Konstruktori un destruktori.	2	2	0	0
Hierarhija. Mantošana un atvasinātās klases. Daudzkārtējā mantošana.	2	2	0	0
Virtuālās metodes. Tīrās virtuālās metodes un abstraktās klases.	2	2	0	0
Izņēmumu apstrāde valodā C++.	2	2	0	0
Statiskie klases locekļi. Klases draugi - friend.	2	2	0	0
Grafiskās saskarnes veidošana Microsoft Visual Studio vidē, izmantojot klašu bibliotēkas.	2	4	0	0
Polimorfisms. Operāciju definēšana klasēm.	4	4	0	0

Šabloni (templates). Šablonu bibliotēkas.	2	2	0	0
OO programmu pirmkoda riski un to mazināšanas paņēmieni.	6	4	0	0
Starppārbaudījumi, eksāmens un gatavošanās tiem.	6	10	0	0
Laboratorijas darbu izpilde un aizstāvēšana.	19	16	0	0
Kopā:	53	54	0	0

Sasniedzamie studiju rezultāti un to vērtēšana

Sasniedzamie studiju rezultāti	Rezultātu vērtēšanas metodes
Spēj izskaidrot OOP pamatkonceptijas (abstrakcija, iekapsulēšana, mantošana, polimorfisms) un to realizāciju programmēšanas valodā C++.	Kontroldarbi, eksāmens un laboratorijas darbu uzdevumi. Kritēriji: spēj izskaidrot OOP pamatkonceptiju realizāciju programmēšanas valodā.
Spēj patstāvīgi veidot uzdevumu risinājumus valodā C++, izmantojot objektorientētās programmēšanas principus, izskaidrot to darbību un izmantotās valodas C++ iespējas.	Laboratorijas darbi un to aizstāvēšana. Kritēriji: spēj izskaidrot uzdevuma realizāciju valodā C++.
Spēj analizēt un izskaidrot gatavu pirmkodu un izskaidrot tajā izmantoto valodas C++ konstrukciju darbību.	Kontroldarbi, eksāmens. Kritēriji: spēj izprast pirmkoda darbību, konstatēt nepilnības, un zina, kā tās novērst.
Prot izmantot integrēto programmatūras izstrādes vidi programmu izveidei un atklādošanai.	Laboratorijas darbi un to aizstāvēšana. Kritēriji: spēj patstāvīgi veidot uzdevuma risinājumu programmēšanas vidē, veikt atklādošanu.

Studiju rezultātu vērtēšanas kritēriji

Kritērijs	% no kopējā vērtējuma
Laboratorijas darbu izstrāde un aizstāvēšana	35
Kontroldarbu vērtējums	25
Eksāmens	40
Kopā:	100

Studiju kursa plānojums

Daļa	KP	Stundas			Pārbaudījumi		
		Lekcijas	Prakt d.	Laborat	Ieskaite	Eksām.	Darbs
1.	4.5	2.0	0.0	1.0		*	