

RTU studiju kurss "Objektorientētā programmēšana"**0L000 Liepājas akadēmija****Vispārējā informācija**

Kods	LA0628
Nosaukums	Objektorientētā programmēšana
Studiju kursa statuss programmā	Obligātais/Ierobežotās izvēles
Atbildīgais mācītspēks	Dzintars Tomsons - Lektors
Apjoms daļās un kredītpunktos	1 daļa, 3.0 kredītpunkti
Studiju kursa īstenošanas valodas	LV, EN
Anotācija	Kursā tiek sniegtas zināšanas par objektorientētās programmēšanas pieeju un izstrādes galvenajiem principiem. Kursa teorētiskajā daļā tiek aplūkoti objektorientētās programmēšanas pamatjēdzieni – mantošana, iekapsulēšana un polimorfisms. Tiek dots arī ieskats par draugu funkcijām, abstraktām klasēm un šabloniem. Praktiskās nodarbības veltītas, lai attīstītu programmēšanas iemaņas valodās C++ un C# un sistēmu modelēšanas prasmes valodā UML.
Mērķis un uzdevumi, izteikti kompetencēs un prasmēs	Studiju kursa mērķis ir sniegt zināšanas par objektorientētās programmēšanas vispārīgām iezīmēm un paradigmām. Studiju kursa uzdevumi: 1. Iepazīstināt studentus ar objektorientēto pieeju programmatūras izstrādē un tās pielietojumiem; 2. Sniegt iespēju apgūt prasmes un iemaņas objektorientētās programmēšanas valodā C++ un C#; 3. Veidot sistēmu modelēšanas prasmes un iemaņas modelēšanas valodā UML.
Patstāvīgais darbs, tā organizācija un uzdevumi	1. Klašu un objektu veidošana Sastādīt C++ programmu, kurā definēta vismaz viena klase, kas īstenota vairākos objektos; 2. Mantošana Sastādīt C++ programmu, kurā definētas mantotājklares, kas īstenotas vairākos objektos; 3. UML klašu diagrammas Izveidot kādas sistēmas UML klašu diagrammu un uzrakstīt atbilstošas C++ klašu definīcijas; 4. Virtuālās funkcijas Sastādīt programmu, iekļaujot tajā virtuālās funkcijas; 5. Studiju projekts objektorientētajā modelēšanā un programmēšanā Izveidot studiju projektā veidotās sistēmas UML klašu diagrammu un uzrakstīt atbilstošas C# klašu definīcijas; 6. Studiju projekta UML lietojumgadījumu diagrammas Izveidot studiju projekta prasību aprakstu un tam atbilstošas UML lietojumgadījumu diagrammas; 7. Sistēmas dinamikas modelēšana Izveidot studiju projektam UML secību, komunikāciju, aktivitāšu un stāvokļu diagrammas; 8. UML modeļa īstenošana Izveidot C# programmu komplektu, kas īsteno studiju projekta UML diagrammas.
Literatūra	Obligātā/ Obligatory: 1. Dz. Tomsons. DatZP077: Objektorientētā programmēšana/Object-Oriented Programming [Elektroniskais resurss] – Tiešsaites raksts. – [Liepāja., b.i. 2022.] – Pieejas veids: tīmeklis www. URL: https://estudijas.liepu.lv/course/view.php?id=677 Papildu/ Additional: 1. Irvine Kip R. C++ and Object-Oriented Programming / Kip R. Irvine. - Upper Saddle River, New Jersey : Prentice-Hall, Inc., 1997. - XVII Citi informācijas avoti/ Other sources of information: 1. UML Resource Group, http://www.uml.org/ 2. UML Tutorial, https://www.tutorialspoint.com/uml/index.htm
Nepieciešamās priekšzināšanas	Algoritmi un datu struktūras

Studiju kursa saturs

Saturs	Pilna un nepilna laika klātienes studijas		Nepilna laika neklātienes studijas	
	Kontakt stundas	Patstāv. darbs	Kontakt stundas	Patstāv. darbs
Objektorientētā pieeja programmatūras izstrādē	2	4	0	0
Klašu un objektu definēšana programmēšanas valodā C++	2	4	0	0
Mantošana	2	4	0	0
Ievads modelēšanas valodā UML. Klašu diagrammas	2	4	0	0
Studiju projekts objektorientētajā modelēšanā un programmēšanā	2	4	0	0
Norādes uz klases objektiem, objektu masīvi	2	4	0	0
Virtuālās funkcijas	2	4	0	0
Abstraktās klases, funkciju un klašu veidnes, statiskas funkcijas, vārdtelpas	2	4	0	0
Programmatūras prasību analīze un UML lietojumgadījumu diagrammas	2	4	0	0

Klašu identificēšana un to raksturošana, izmantojot UML un C#	2	2	0	0
Sistēmas dinamikas modelēšana, izmantojot modelēšanas valodu UML	2	4	0	0
UML secību diagrammas un komunikāciju diagrammas; to īstenošana programmēšanas valodā C#	2	4	0	0
UML aktivitāšu diagrammas un stāvokļu diagrammas; to īstenošana programmēšanas valodā C#	2	2	0	0
Objektorientētā programmēšana un datubāzes	6	4	0	0
Kopā:	32	52	0	0

Sasniedzamie studiju rezultāti un to vērtēšana

Sasniedzamie studiju rezultāti	Rezultātu vērtēšanas metodes
Zināšanas: Zina un izprot objektorientētās programmēšanas pamatprincipus;	Jāizpilda visi patstāvīgā darba uzdevumi; jāsagatavo programmatūras sistēmas UML modeļi un jāizveido tiem atbilstoši C++ un C# programmu komplekti.
Prasmes: Prot kodēt un atklūdot programmas; Prot veidot programmas, izmantojot objektorientēto pieeju; Prot pielietot projektējuma shēmas un diagrammas; Prot izvēlēties uzdevumu risināšanai adekvātus līdzekļus;	Jāizpilda visi patstāvīgā darba uzdevumi; jāsagatavo programmatūras sistēmas UML modeļi un jāizveido tiem atbilstoši C++ un C# programmu komplekti.
Spēj kodēt, lasot un analizējot svešus programmu tekstus; Spēj kodēt, lasot un saprotot programmatūras projektējuma aprakstus; Spēj kodēt, rakstot programmas kodu saskaņā ar projektējumu un kodēšanas vadlīnijām; Spēj kodēt, dokumentējot kodu; Spēj projektēt, veidojot un aprakstot programmatūras arhitektūru; Spēj projektēt, veidojot realizācijas modeli (klašu un/vai funkciju hierarhiju); Spēj projektēt, sagatavojot programmatūras projektējuma apraksta dokumentu;	Jāizpilda visi patstāvīgā darba uzdevumi; jāsagatavo programmatūras sistēmas UML modeļi un jāizveido tiem atbilstoši C++ un C# programmu komplekti.
Kompetences: Prot lietot labu programmēšanas stilu; Prot veikt darbu patstāvīgi; Spēj iesaistīties projektos, kur tiek izmantota objektorientētā programmēšanas pieeja.	Jāizpilda visi patstāvīgā darba uzdevumi; jāsagatavo programmatūras sistēmas UML modeļi un jāizveido tiem atbilstoši C++ un C# programmu komplekti.

Studiju rezultātu vērtēšanas kritēriji

Kritērijs	% no kopējā vērtējuma
Patstāvīgie darbi	100
Kopā:	100

Studiju kursa plānojums

Daļa	KP	Stundas			Pārbaudījumi		
		Lekcijas	Prakt d.	Laborat	Ieskaite	Eksām.	Darbs
1.	3.0	16.0	16.0	0.0	*		