



## DALYKO (MODULIO) APRAŠAS

Modulio pavadinimas	Kodas
Objektinis programavimas C++	

### Anotacija

Kursas apima objektinio programavimo kertines koncepcijas bei principus, skatinančius abstrakcijomis paremtą mąstymą bei iš informacijos slėpimo principo išplaukiančią programinio kodo kokybę, sąlygojančią jo skaitomumą, lengvesnį palaikymą bei atsparumą pokyčiams. Kurse susipažįstama su įvairiais irankiais bei technologijomis, taip pat nagrinėjami objektiškai orientuoto projektavimo šablonai, leidžiantys gauti standartinius sprendimus tipinėms probleminėms situacijoms. Kursas skirtas turintiems programavimo pagrindus ir yra laikomas nuosekliu procedūrinio programavimo pagrindų kurso tęsiniu.

Dėstytoja(s)	Padalinys
<b>Koordinuojanti(s):</b> Irmantas Radavičius <b>Kiti:</b> Rokas Astrauskas, Pijus Kasparaitis	Informatikos katedra Matematikos ir informatikos fakultetas Vilniaus Universitetas

Studijų pakopa	Dalyko tipas
Pirmoji	Privalomasis

Įgyvendinimo forma	Vykdyimo laikotarpis	Vykdyimo kalba
Auditorinė	2 semestras	Lietuvių, anglų

### Reikalavimai studijuojančiam(-ai)

<b>Išankstiniai reikalavimai:</b> Programavimo pagrindai	<b>Gretutiniai reikalavimai:</b> nėra
--	---------------------------------------

Modulio apimtis kreditais	Visas studento darbo krūvis	Kontaktinio darbo valandos	Savarankiško darbo valandos
5	134	66	68

### Modulio tikslas: studijų programos ugdomos kompetencijos

#### Šio kurso paskirtis:

- Supažindinti su C++ programavimo kalba, STL ir kitomis C++ bibliotekomis
- Supažindinti su objektinio ir apibendrinto programavimo paradigmomis
- Supažindinti su objektiškai orientuotos analizės bei projektavimo pagrindais, naudojant UML kalbą

**Bendrosios kompetencijos:** Žinias pritaikyti praktikoje (BK2), Organizuoti ir planuoti darbus, dirbti individualiai ir grupėje (BK3). **Dalykinės kompetencijos:** Programavimo (DK6), Programų sistemų inžinerijos (DK8).

Modulio studijų siekiniai	Studijų metodai	Vertinimo metodai
<b>Studentai gebės:</b> <ul style="list-style-type: none"><li>• gebės taikyti objektiškai-orientuoto programavimo technikas realaus pasaulio situacijoms modeliuoti</li><li>• gebės suprasti, modifikuoti bei rašyti programinį kodą C++ kalba</li><li>• gebės kurti, testuoti, dokumentuoti C++ programas</li></ul>	Paskaitos Laboratoriniai darbai Savarankiškas darbas	Laboratorinių darbų užduotys Egzaminas (raštu)

Temos	Kontaktinio darbo valandos						Savarankiškų studijų laikas ir užduotys	
	Paskaitos	Konsultacijos	Seminarai	Pratybos	Laboratoriniai darbai	Visas kontaktinis darbas	Savarankiškas darbas	Užduotys
Kurso apžvalga. Įvadas. Objektinis programavimas. Objektinio programavimo kalbos. C++ kalbos istorija. C++ kaip „geresnis C“.	2				2	4	4	Savarankiškas skaitymas Laboratorinių darbų užduotys
Klasės ir objektai. Laukai ir metodai. Prieigos kontrolė. Objekto gyvavimo ciklas. Konstruktoriai ir destruktoriai. Statiniai klasių elementai	2				2	4	4	
Daugiafailės programos. Vardų erdvės. Antraštės failai. Klaidų paieška ir taisymas. Vienetų testai. Tarpiniai išvedimai. Klaidų apdorojimas. Tvirtinimai. Išimtinės situacijos.	2				2	4	4	
Metodai, metodų perkrovimas. Konstantos, konstantiniai laukai ir metodai. Parametrų perdavimas. Rodyklės ir sinonimai. Operatorių perkrovimas. Draugiškos klasės ir funkcijos.	2				2	4	4	
Kompozicija. Kompozicijos rūšys. Objektų paviršutinis ir gilus kopijavimas. Vidinės klasės. Masyvai. Standartiniai konteineriai.	2				2	4	4	
Objektiškai orientuota analizė ir projektavimas. UML kalba. UML panaudojimo atvejų, veiklos, klasių, sekų diagramos.	2				2	4	4	
Apibendrintas programavimas. Funkcijų ir klasių šablonai.	2				2	4	4	
STL konteineriai. STL iteratoriai.	2				2	4	4	
STL funkciniai objektai. STL algoritmai.	2				2	4	4	
Paveldėjimas. Metodų perrašymas. Virtualūs metodai. Polimorfizmas. Tipų konversija	2				2	4	4	
Paveldėjimo būdai. Matomumo specifikatoriai. Paveldėjimo tipai. „Deimanto problema“. Virtualus paveldėjimas.	2				2	4	4	
Abstrakčios klasės. Projektavimo šablonai	2				2	4	4	
Objektų kūrimas ir kopijavimas. Projektavimo šablonai.	2				2	4	4	
Objektiškai orientuota analizė ir projektavimas. Projektavimo šablonai	2				2	4	4	
Resursų valdymas. Išimtytys ir saugus kodas	2				2	4	4	
Pasirengimas egzaminui	2				2	4	8	
Egzaminas (raštu)						2		
<b>Iš viso:</b>	32				32	66	68	

Vertinimo strategija	Svoris (proc.)	Atsiskaitymo laikas	Vertinimo kriterijai
Laboratorinių darbų užduotys	50	Semestro bėgyje kurso dalyviams pateikiamos savaitinės užduotys, kurios nuolat atsiskaitomos pagal semestro pradžioje paskelbtą tvarką	Visos savaitės laikomos vienodai vertingomis, ir jas atsiskaičius laiku ir tinkamai, už temą/užduotį galima gauti 5/N taško, kur N – temų/užduočių skaičius. Kad būtų leidžiama laikyti egzaminą, semestro bėgyje turi būti surinkta bent 40% taškų (2 balai).
Egzaminas (raštu)	50	Egzaminas laikomas sesijos metu	Egzamino metu kurso dalyviai sprendžia įvairaus tipo ir sudėtingumo užduotis. Egzaminą sudaro dvi dalys: testas (jo metu dalyviai sprendžia užduotis, atsakinėdami į įvairaus tipo klausimus) bei kodo rašymas (jam pasibaigus dalyviai turi pateikti programinį kodą, atitinkantį jam keliamus reikalavimus). Kad gauti teigiamą įvertinimą, iš egzamino privalu surinkti bent 40% taškų (2 balai).
Eksternu		Dalykas gali būti kartojamas eksternu, jeigu prieš tai buvo dalyvauta jame pilna apimtimi, ir studijuojantį tenkina anksčiau surinktas taškų skaičius, kuris tokiu atveju užskaitomas ir kartojamas tik egzaminas. Eksternu dalyką kartojantis studentas privalo apie tai informuoti dėstytoją semestro pradžioje, ir gauti raštišką patvirtinimą, kuriame nurodomas įskaitomas taškų skaičius. Jeigu studijuojantis nesurinko minimalaus šiame apraše įvardinto taškų skaičiaus, arba surinktas taškų skaičius netenkina, dalyko kartoti eksternu negalima.	

Autorius	Metai	Pavadinimas	Leid.	Vieta
<b>Privaloma literatūra</b>				
Bjarne Stroustrup	2013	The C++ Programming Language	4th ed.	Addison-Wesley
Paul J. Deitel, Harvey M. Deitel	2016	C++ How to Program	10th ed.	Pearson
<b>Papildoma literatūra</b>				
Bruce Eckel	2000	Thinking in C++	2nd ed.	The Benjamin/Cummings Publishing Company, Inc.
Bruce Eckel	2003	Thinking in C++, Volume Two: Practical Programming	1st ed.	<a href="https://archive.org/details/TI_CPP2ndEdVolOne">https://archive.org/details/TI_CPP2ndEdVolOne</a>
Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides	1994	Design Patterns: Elements of Reusable Object-Oriented Software	1st ed.	<a href="https://archive.org/details/TI_CPP2ndEdVolTwo">https://archive.org/details/TI_CPP2ndEdVolTwo</a>