



MODULIO APRAŠAS

Modulio pavadinimas	Kodas
Programavimo metodikos pagrindai	

Dėstytojas	Padalinys
Koordinuojantis: Saulius Gražulis Kitas (-i):	Matematinės informatikos katedra Matematikos ir informatikos fakultetas Vilniaus universitetas

Studijų pakopa	Dalyko tipas
Pirmoji	Privalomasis

Įgyvendinimo forma	Vykdyimo laikotarpis	Vykdyimo kalbos
Auditorinė	5 semestras	Lietuvių

Reikalavimai studijuojančiajam
Išankstiniai reikalavimai: Perlo programavimo kalba, Informatikos pagrindai, duomenų struktūros Pageidautina žinoti: tiesinė algebra, operacinės sistemos, kompiuterių architektūra

Modulio apimtis kreditais	Visas studento darbo krūvis	Kontaktinio darbo valandos	Savarankiško darbo valandos
5	130	50	80

Modulio tikslas: studijų programos ugdomos kompetencijos		
<p>Modulio tikslas – suteikti studentams bazinius organizuoto programavimo įgūdžius: išmokyti juos naudotis versijų valdymo sistemomis, naudoti automatines programų testavimo priemones, duoti pradines žinias apie programų verifikavimą, suteikti žinių apie programavimo stilius ir pamokyti dirbti į Unix sistemą panašiose aplinkose (pvz. GNU/Linux sistemoje).</p> <p>Šis kursas numatytas kaip įvadinis kursas tolimesniai struktūrinės bioinformatikos kursui, kur toliau bus naudojamos išmoktos programavimo priemonės.</p> <p>Bendrosios kompetencijos:</p> <ul style="list-style-type: none"> Gebėjimas ieškoti duomenų informacijos šaltiniuose, analizuoti, vaizduoti ir sisteminti gautus duomenis. (BK1). Žinias pritaikyti praktikoje (BK2). Gebėjimas organizuoti ir planuoti darbus, dirbti individualiai ir grupėje, gebėjimas bendradarbiauti su kitų profesinių sričių žmonėmis. (BK3). <p>Dalykinės kompetencijos:</p> <ul style="list-style-type: none"> Algoritmų ir duomenų struktūrų (DK5). Programavimo modelių ir interneto technologijų (DK6). Programų sistemų inžinerijos (DK8). Bioinformatikos duomenų gavybos, vaizdavimo ir analizės (DK11). 		
Modulio studijų siekiniai	Studijų metodai	Vertinimo metodai
<p>Išmokti savarankiškai efektyviai dirbti į Unix sistemą panašiose aplinkose.</p> <p>Suvokti pagrindinius versijų kontrolės (valdymo) sistemų principus ir uždavinius;</p> <p>išmokti efektyviai dirbti su Subversion versijų kontrolės sistema ir analogiškomis sistemomis.</p>	<p>Paskaitos, praktikos darbai, individualių programavimo užduočių įgyvendinimas</p>	<p>Apklausa virtualioje mokymo sistemoje; egzaminas (uždavinių sprendimas, ese); studentų darbų pristatymas raštu ir žodžiu; studentų programavimo darbų</p>

Suvokti programų testavimo būtinumą, suprasti testavimo teikiamas galimybes bei testavimo galimybių ribas. Išmokti savarankiškai konstruoti ir pritaikyti automatinius testus savo parašytoms programoms panaudojant GNU Make sistemą.		peržiūra ir priėmimo testai.
Įgyti pradines žinias apie programų verifikavimą.		
Išmokti tvarkingai rašyti lengvai skaitomą, nesunkiai lydimą programos kodą, nuosekliai naudojant pasirinktą kodavimo stilių.		
Sužinoti apie šiuolaikines programavimo metodikas – lankstųjį programavimą (agile development), testais paremtą programos vystymą (test-driven development), ekstremalų programavimą (XP, extreme programming) ir išmokti taikyti jų elementus praktikoje.		

Temos	Savarankiškų studijų laikas ir užduotys						Savarankiškų studijų laikas ir užduotys	
	Paskaitos	Konsultacijos	Seminarai	Pratybos	Laboratoriniai darbai (LD)	Visas kontaktinis darbas	Savarankiškas darbas	Užduotys
1. Unix architektūros principai, failų sistema, komandos	4			2		6	7	
2. Versijų kontrolė (valdymas) ir Subversion	4			2		6	14	
3. Programų kūrimo aplinka Unix bei GNU Linux sistemose ir jos galimybės	4			4		8	14	
4. Programų testavimas, programų surinkimas ir procesų automatizavimas GNU Make sistema	8			4		12	14	
5. Unix tipo aplinkos ir GNU Make sistemos pritaikymas duomenų apdorojimui	4			4		8	14	
6. Programų analizė ir teisingumo įrodymai	4					4	7	
7. Unix, GNU ir Linux sistemų papildomos galimybės, istorija, savybių analizė	4					4	7	
8. Pasiruošimas egzaminui, egzaminas	2					2	3	
Iš viso	34			16		66	72	

Vertinimo strategija	Svoris proc.	Atsiskaitymo laikas	Vertinimo kriterijai
Darbas paskaitose (paskaitų apklausos)	10	10 min. prieš kiekvieną praktikos darbą	Testas iš 4 klausimų, apimančių jau išklaustyti paskaitų medžiagą. Sumuojamas balų už teisingai atsakytus klausimus skaičius; maksimali visų tokių apklausų balų suma – 100 balų. Paprastai organizuojamas virtualioje mokymosi aplinkoje, bet gali būti panaudotas testas raštu.
Tarpinis kontrolinis	15	semestro vidurys (po maždaug pusės paskaitų)	Testas (virtuali mokymosi aplinka) iš klausimų, apimančių iki tol praeitas temas. Maksimali balų už atsakymus suma – iki 150 balų.
Praktikos darbų atsiskaitymai	50	Po kiekvieno praktikos darbo pagal nustatytą grafiką.	Studentai įkelia į virtualią mokymosi aplinką savo praktinio darbo rezultatus. Vertinimo kriterijai: teisingas programos veikimas, programavimo stilius ir programos teksto tvarkingumas; bendros dalyko žinios. Atliktos užduotys vertinamos subtraktyvia sistema: idealiai padaryta užduotis vertinama 100%; už kiekvieną padarytą klaidą atimamas klaidos svarbą atitinkantis balų skaičius (su paaiškinimu apie klaidos esmę ir svarbą). Vertinimui gali būti pasitelktos automatinės kodo analizės priemonės. Klaidų pataisymui gali būti skiriamos papildomos (neprivalomos) užduotys.
Praktikos darbo ataskaitos pristatymas	10	Paskutinė semestro savaitė	Studentai įkelia į virtualią mokymosi aplinką savo praktinio darbo ataskaitą ir perskaito 5–10 min. pranešimą grupėje. Darbo ataskaitą būtina pateikti pagal VU darbų apipavidalinimo taisykles. Vertinimo metofika: naudojant Moodle sistemos rubrikas arba subtraktyvią sistemą.
Egzaminas	15	Egzaminų sesija (galutinio atsiskaitymo laikotarpis)	<p>Testas (virtuali mokymosi aplinka) klausimų, apimančių visas kurso temas.</p> <p>Kad studentai būtų prileisti prie egzamino, jie turi:</p> <ol style="list-style-type: none"> 1. Atlikti bent vieną praktikos darbą ir surinkti teigiamą (didesnį už nulį) pratybų balą; 2. Sukaupti suminį balą už darbą per semestrą (iš praktikos darbų, tarpinio kontrolinio, darbo paskaitose, galimai kitų dėstytojų paskirtų užduočių), kad, parašius egzamino kontrolinį, būtų įmanoma pasiekti patenkinamo pažymio balą (t. y. balą, užtikrinantį bent pažymį „5“); <p>Egzamino kontrolinis yra būtinas visiems nepriklausomai nuo surinkto balų skaičiaus. Studentams, neatvykusiems į egzaminą, žiniaraštyje bus žymima „neatvyko“. Egzamine būtina surinkti bent 50% egzamino balo.</p>
Viso	100		Galutinis pažymys yra suminis paskaitų apklausų, tarpinio kontrolinio, praktikos darbų rezultatų, praktikos darbo ataskaitos ir egzamino balas (iki 1000 balų), padalintas iš 100, apvalinamas iki artimiausio <i>didesnio</i> sveiko skaičiaus (pvz., 901 balas apvalinamas iki galutinio įvertinimo 10).

Reikalavimai dalyko vertinimui eksterno būdu

Įvertinimas galimas eksterno būdu: nenumatytas

Autorius	Leidi mo metai	Pavadinimas	Periodinio leidinio Nr. ar leidinio tomas	Leidimo vieta ir leidykla ar internetinė nuoroda
Privalomoji literatūra				
Ben Collins-Sussman, Brian W. Fitzpatrick, C. Michael Pilato	2011	Version control with Subversion		O'Reilly Media, Inc., http://shop.oreilly.com/ produc t/9780596004484.do , ISBN 978-0596510336, http://svnbook.red- bean.com/
Richard M. Stallman, Roland McGrath, Paul D. Smith	2010	GNU Make		Free Software Foundation, http://www.gnu.org/soft- ware/ make/manual/
Kernighan, Brian W.	1984	The UNIX programming environment		Prentice-Hall, Inc.; ISBN 0-13-937681-X
Бриан Керниган	1992	UNIX-универсальная среда программирования		Москва: Финансы и статистика
Papildoma literatūra				
Wikipedia	2013 – 2021	Test-driven development		http://en.wikipedia.org/ wiki/Test- driven_development
Kent Beck	2003	Test-Driven Development By Example		Addison-Wesley, Boston, ISBN-13: 978- 0321146533
Kent Beck, Erich Gamma	2005	Extreme Programming Ex- plained: Embrace Change, 2nd Edition (The XP Series)		Addison-Wesley, Boston, ISBN- 13: 978-0321278654
Bourne, S. R.	1983	The UNIX system		Addison-Wesley, Boston, ISBN 0-201-13791-7
С. Баурн	1986	Операционная система UNIX		Москва: Мир