UČNI NAČRT PREDMETA / COURSE SYLLABUS (leto / year 2016/17)			
Predmet:	Programiranje		
Course title:	Programming		
Študijski program in stopnja **Study programme and level**	**Študijska smer** **Study field**	**Letnik** **Academic year**	**Semester** **Semester**
Interdisciplinarni magistrski študijski program Računalništvo in matematika	ni smeri	1	prvi
Interdisciplinary Master's study programme Computer Science and Mathematics	none	1	first

Vrsta predmeta / Course type	obvezni / compulsory
Univerzitetna koda predmeta / University course code:	M2818

Predavanja **Lectures**	**Seminar** **Seminar**	**Vaje** **Tutorial**	**Klinične vaje** **work**	**Druge oblike študija**	**Samost. delo** **Individ. work**	**ECTS**
45	**10**	**20**			**105**	**6**

Nosilec predmeta / Lecturer:	prof. dr. Zoran Bosnić

Jeziki / Languages:	**Predavanja / Lectures:**	slovenski / Slovene, angleški / English
	Vaje / Tutorial:	slovenski / Slovene, angleški / English

Pogoji za vključitev v delo oz. za opravljanje študijskih obveznosti:	**Prerequisites:**
Vpis v letnik študija.	Enrolment in the programme.
Vsebina:	**Content (Syllabus outline):**

Predmet vsebuje različne teme s področja programiranja, ki so priporočene v ACM in IEEEjevem kurikulumu za računalništvo. Konkretna vsebina se bo letno prilagajala trendom. Trenutno aktualne teme z ustreznimi programskimi jeziki in/ali okolji, so, na primer: Funkcijsko programiranje (Lisp/Scheme ali Haskell ali Python ali JavaScript) Deklarativno programiranje (SQL, podatkovne baze, Prolog) Nerelacijske podatkovne baze (CouchDB) Programiranje v oblaku (npr. Google App Engine) Paralelizem v oblaku (MapReduce, Python, Java, C++, Php) Razlika med programiranjem v prevajanih in interpretiranih jezikih (Java ali C v primerjavi z jeziki Python ali Ruby ali R ali Php ali JavaScript) Sistemi po načelu Model-Pogled-Kontrola (Joomla ali Django in Php ali Python) Programiranje v objektno usmerjenih jezikih s prvorazrednimi funkcijami (Python ali Lisp/Scheme) Dogodkovno vodeni programi (npr. Python s Qtjem ali C++ s Qtjem) Programiranje s shranjevanjem sej (npr. spletno programiranje brez uporabe že narejenih ogrodij, JavaScript s Phpjem ali Pythonom) Programiranje vzorcev (template) (C++, Python) Programiranje z dogovori (contract) (Smalltalk) Posebnosti programiranja mobilnih naprav (Android z Javo)	The course will include topics in programming recommended in the ACM/IEEE curriculum for CS. Concrete topics will change each year according to trends in computer science and industry. Potential topics and the corresponding programming languages and/or environments at the moment are, for example: Functional programming (Lisp/Scheme or Haskell or Python or JavaScript) Declarative programming (SQL, databases, Prolog) Non-relational databases (Couch DB) Cloud Programming (e.g. Google App Engine) Parallelism in cloud programming (MapReduce, Pytho, Java, C++, Php) Difference between programming in compiled and interpreted languages (Java or C in comparison with Python or Ruby or R or Php or JavaScript) Systems in Model-View-Controller paradigm (Joomla or Django with Php or Python, respectively) Programming in pure object oriented language with first-class functions (Python or Lisp/Scheme) Event-driven programming (e.g. Python or C++ with Qt) Programming with sessions (e.g. web programming without frameworks, JavaScript with Php or Python) Programming templates (C++, Python) Programming with contracts (Smalltalk) Programming for mobile devices (Android and

Paralelizem z uporabo grafičnih procesorjev (CUDA)	Java) Parallel programming using graphics processing units (CUDA)

Temeljni literatura in viri / Readings:

Splošna:

S. Oualline: Practical C++ Programming, 2nd Edition, O'Reilly Media, 2009.

R. K. Dybvig: The Scheme Programming Language, 4th Edition, MIT Press 2009.

J. Demšar: Python za programerje, Založba FE in FRI, Ljubljana, 2009.

C. Heilmann: Beginning JavaScript with DOM Scripting and Ajax, Apress, 2006

Dodatni viri, specifični za posamezne teme:

A. Holovaty, J. Kaplan-Moss: The Definitive Guide to Django, 2nd Edition, Apress, 2009.

C. Chung: Pro Objective-C Design Patterns for iOS, Apress, 2011.

J. Sanders: CUDA by Example: An Introduction to General-Purpose GPU Programming, Addison-Wesley Professional, 2010.

R. Meier: Professional Android 2 Application Development, 2nd Edition, Wrox, 2010.

Cilji in kompetence:	**Objectives and competences:**
Študenti, ki so dokončali prvostopenjski študij RI, so opravili predmete s področja osnov programiranja, pri drugih predmetih pa spoznali različne pristope in paradigme programiranja. Cilj tega predmeta je združiti implicitna znanja v strnjen okvir sledeč priporočilom ACM and IEEE. Študenti bodo spoznali različne tehnike v njihovih relevantnih kontekstih in z ustreznimi programskimi jeziki. Študenti brez predznanja ali s pomanjkljivim znanjem programiranja bodo morali za sledenje predmetu vložiti več truda in po potrebi	Students who finished the undergraduate study of computer science already completed courses on basics of programming and used various programming approaches and paradigms within other subjects. The objective of this course is to present this implicit knowledge within a unified perspective following the recommendations of ACM and IEEE. Students will be exposed to various techniques within their relevant contexts and programming languages.Students lacking the sufficient skills in programming will need to

obiskovati ustrezne predmete prvostopenjskega študija.	put in some extra effort and also attend the undergraduate courses if needed.

Predvideni študijski rezultati:	**Intended learning outcomes:**
Znanje in razumevanje: Študent bo poznal in znal uporabljati različne pristope k programiranju v odvisnosti od konkretnih kontekstov. Uporaba: predmet bo študentom predstavljal osnove različnih področij programiranja, ki so aktualne za potrebe računalniške industrije. Predmet bo od študenta poleg prilagajanja različnim paradigmam zahteval tudi hitro učenje različnih jezikov in ga s tem pripravljal na delo v sodobni računalniški industriji. Refleksija: Poleg konkretnih znanj bodo študenti dobili tudi teoretičen pregled nad različnimi področji programiranja, kar jim bo omogočilo tudi boljše prilagajanje potrebam industrije v prihodnosti. Prenosljive spretnosti - niso vezane le na en predmet: Znanje programiranja je potrebno za večino drugih predmetov študija.	Knowledge and understanding: The student will understand and be able to apply different approaches to programming suitable to various contexts.Application: The course will present various areas of pogramming relevant to the current trends. The subject will, on purpose, require quick adaptations to various paradigms and languages, which will prepare the students for successful work in modern computer industry.Reflection: Besides the practical knowledge, the students will gain a theoretical insight into various forms of programming, which will enable them for faster adaptations to new techniques that will appear in the future.Transferable skills: Programming is the basic skill and an implicitly required prerequisit for most other courses.

Metode poučevanja in učenja:	**Learning and teaching methods:**
Predavanja in domača naloge. Poseben poudarek je na individualnem delu študentov.	Lectures and homeworks with special emphasis on individual work.

Načini ocenjevanja:	Delež (v %) / Weight (in %)	**Assessment:**

Način (pisni izpit, ustno izpraševanje, naloge, projekt): Sprotno preverjanje (domače naloge) Končno preverjanje (pisni in ustni izpit) Ocene: 6-10 pozitivno, 1-5 negativno (v skladu s Statutom UL)	 50% 50%	Type (examination, oral, coursework, project):Continuing (homeworks)Final (written and oral exam) Grading: 6-10 pass, 1-5 fail (according to the rules of University of Ljubljana)

Reference nosilca / Lecturer's references:

BOSNIĆ, Zoran, VRAČAR, Petar, RADOVIĆ, Miloš D., DEVEDŽIĆ, Goran, FILIPOVIĆ, Nenad D., KONONENKO, Igor. Mining data from hemodynamic simulations for generating prediction and explanation models. IEEE transactions on information technology in biomedicine, ISSN 1089-7771. [Print ed.], Mar. 2012, vol. 16, no. 2, str. 248-254, ilustr. [COBISS.SI-ID 9026900]

POGORELC, Bogdan, BOSNIĆ, Zoran, GAMS, Matjaž. Automatic recognition of gait-related health problems in the elderly using machine learning. Multimedia tools and applications, ISSN 1380-7501, 2012, vol. 58, no. 2, str. 333-354, graf. prikazi. [COBISS.SI-ID 8773460]

VOGRINČIČ, Sergeja, BOSNIĆ, Zoran. Ontology-based multi-label classification of economic articles. Computer science and information systems, ISSN 1820-0214. [Print ed.], 2011, vol. 8, no. 1, str. 101-119, ilustr. [COBISS.SI-ID 8235860]

BOSNIĆ, Zoran, KONONENKO, Igor. Correction of regression predictions using the secondary learner on the sensitivity analysis outputs. Computing and informatics, ISSN 1335-9150, 2010, vol. 29, no. 6, str. 929-946, graf. prikazi. [COBISS.SI-ID 8027988]

ČEHOVIN, Luka, BOSNIĆ, Zoran. Empirical evaluation of feature selection methods in classification. Intelligent data analysis, ISSN 1088-467X. [Print ed.], 2010, vol. 14, no. 3, str. 265-281, ilustr. [COBISS.SI-ID 7732564]