

UČNI NAČRT PREDMETA / COURSE SYLLABUS (leto / year 2017/18)						
Predmet:		Programiranje 1				
Course title:		Programming 1				
Študijski program in stopnja Study programme and level		Študijska smer Study field		Letnik Academic year	Semester Semester	
Enoviti magistrski študijski program Pedagoška matematika		ni smeri		3 ali 4	prvi	
Integrated Master's study programme Pedagogical Mathematics		none		3 or 4	first	
Vrsta predmeta / Course type				izbirni / elective		
Univerzitetna koda predmeta / University course code:				M0534		
Predavanja Lectures	Seminar Seminar	Vaje Tutorial	Klinične vaje work	Druge oblike študija	Samost. delo Individ. work	ECTS
30		30			90	5
Nosilec predmeta / Lecturer:		prof. dr. Andrej Bauer				
Jeziki / Languages:		Predavanja / Lectures:		slovenski / Slovene		
		Vaje / Tutorial:		slovenski / Slovene		
Pogoji za vključitev v delo oz. za opravljanje študijskih obveznosti:				Prerequisites:		
Vpis v letnik študija.				Enrolment in the programme.		
Opravljen predmet Uvod v programiranje.				Completed course Introduction to programming.		
Vsebina:				Content (Syllabus outline):		

<p>Strukturirano programiranje: podatkovni tipi in abstraktne podatkovne strukture. Implementacija in uporaba osnovnih podatkovnih tipov in struktur v izbranem programskem jeziku. Sezname, drevesa, slovarji in zgoščene tabele.</p> <p>Osnove analize algoritmov. Iskanje z bisekcijo, urejanje z vstavljanjem, hitro urejanje. Uporaba regularnih izrazov.</p> <p>Simbolno računanje: prepisovalna pravila, poenostavljanje izrazov, reševanje enačb, grafični prikaz matematičnih objektov.</p>	<p>Structured programming: data types and abstract data structures. Implementation and use of basic data types and data structures in a selected programming language. Lists, trees, dictionaries and hash tables.</p> <p>Basics of algorithm analysis. Searching by bisection method, insertion sort, quicksort. Use of regular expressions.</p> <p>Symbolic computation: rewrite rules, simplifying expressions, solving equations, graphical representation of mathematical objects.</p>
--	---

Temeljni literatura in viri / Readings:

<p>Priročniki in učbeniki o programiranju v programskem jeziku, ki ga študenti spoznajo. Manuals and textbooks on programming in appropriate programming language.</p>
--

Cilji in kompetence:

<p>Študent spozna osnove strukturiranega programiranja in nekatere osnovne podatkovne strukture in algoritme.</p>

Objectives and competences:

<p>A student gets familiar with the basics of structured programming and certain basic data structures and algorithms.</p>
--

Predvideni študijski rezultati:

<p>Znanje in razumevanje: Osnovno znanje vsaj enega programskega jezika in osnovnih programerskih tehnik.</p> <p>Uporaba: Programerske veščine študent uporabi pri ostalih predmetih, ki zahtevajo reševanje problemov z računalnikom.</p> <p>Refleksija: Strukturirano programiranje omogoča abstrakten in konceptualno prečiščen</p>
--

Intended learning outcomes:

<p>Knowledge and understanding: Basic knowledge of at least one programming language and basic programming techniques.</p> <p>Application: A student can use the skills of programming at other courses that require problem solving using a computer.</p> <p>Reflection: Structured programming enables abstract and conceptually clean approach to</p>
--

pristop k programiranju, ki sloni na diskretnih strukturah.

Prenosljive spretnosti – niso vezane le na en predmet: Programerske sposobnosti so uporabne pri nadaljnjih računalniških predmetih, kakor tudi pri numeričnih metodah.

programming that is based on discrete structures.

Transferable skills: Programming capabilities are useful in further computer science courses as well as the courses on numerical methods.

Metode poučevanja in učenja:

Predavanja, vaje, domače naloge, konzultacije

Learning and teaching methods:

Lectures, exercises, homework, consultations

Načini ocenjevanja:	Delež (v %) / Weight (in %)	Assessment:
<p>Način (pisni izpit, ustno izpraševanje, naloge, projekt):</p> <p>2 kolokvija namesto izpita iz vaj, izpit iz vaj,</p> <p>izpit iz teorije</p> <p>ocene: 1-5 (negativno), 6-10 (pozitivno) (po Statutu UL)</p>	<p>50%</p> <p>50%</p>	<p>Type (examination, oral, coursework, project):</p> <p>2 midterm exams instead of written exam, written exam</p> <p>oral exam</p> <p>grading: 1-5 (fail), 6-10 (pass) (according to the Statute of UL)</p>

Reference nosilca / Lecturer's references:

Andrej Bauer:

BAUER, Andrej, STONE, Christopher A. RZ: a tool for bringing constructive and computable mathematics closer to programming practice. Journal of logic and computation, ISSN 0955-792X, 2009, vol. 19, no. 1, str. 17-43. [COBISS.SI-ID 15325785]

BAUER, Andrej, TAYLOR, Paul. The Dedekind reals in abstract Stone duality. Mathematical structures in computer science, ISSN 0960-1295, 2009, vol. 19, iss. 4, str. 757-838. [COBISS.SI-ID

15322201]

BAUER, Andrej, BIRKEDAL, Lars. Continuous functionals of dependent types and equilogical spaces. V: CLOTE, Peter G. (ur.). Computer science logic : 14th international workshop, CSL 2000, annual conference of the EACSL, Fischbachau, Germany, August 21-26, 2000 : proceedings, (Lecture notes in computer science, ISSN 0302-9743, 1862). Berlin [etc.]: Springer, 2000, vol. 1862, str. 202-216. [COBISS.SI-ID 10606681]

BAUER, Andrej. Uvod v programiranje v Javi. Ljubljana: [A. Bauer], 2008. 1 optični disk (CD-ROM). [COBISS.SI-ID 14629977]

BAUER, Andrej. Teorija programskih jezikov. Ljubljana: [A. Bauer], 2007. 100 str. [COBISS.SI-ID 14630489]