UČNI NAČRT PREDMETA / COURSE SYLLABUS	
Predmet:	Uvod v programiranje
Course title:	Introduction to Programming

Študijski program in stopnja Study programme and level	Študijska smer Study field	Letnik Academic year	Semester Semester
Informatika v sodobni družbi, visokošolski strokovni in univerzitetni študijski program prve stopnje	-	Prvi	Prvi
Informatics in Contemporary Society, first cycle Professional Study Programme and Academic Study programme	-	First	First

Vrsta predmeta / Course type	Obvezni / Obligatory
Univerzitetna koda predmeta / University course code:	1-ISD-VS,UN-UP-2019-05-13

Predavanja Lectures	Seminar Seminar	Vaje Tutorial	Klinične vaje work	Druge oblike študija	Samost. delo Individ. work	ECTS
30	-	45	-	-	135	7

Nosilec predmeta / Lecturer:

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

Pogoji za vključitev v delo oz. za opravljanje študijskih obveznosti:

Pogoj za vključitev v delo je vpis v 1. letnik študija, ustrezna prisotnost na vajah in zagovorjena seminarska naloga.

Prerequisits:

Enrollment into the first year of the study programme, appropriate presence during the lab work and finished student project.

Vsebina:

- Uvod: programiranje kot reševanje problemov, kratka zgodovina programiranja.
- Algoritem in program: diagrami poteka.
- Strukturirano in objektno usmerjeno programiranje: ključni koncepti objektno usmerjenega programiranja.
- Programski jezik Java: javanski virtualni računalnik.
- Osnovni podatkovni tipi.

Content (Syllabus outline):

- Introduction: programming as problem solving. Brief history of programming.
- Algorithm and program: flowcharts.
- Structured and object oriented programming:key concepts of object oriented programming.
- Java programming language: Java virtual machine.
- Basic data types.
- Declaring constants and variables.

• Deklaracije konstant in spremenljivk. • Prireditveni stavek, pisanje izrazov, operatorji. • Krmilni stavki. • Tabele, nizi. • Metode, razredi in objekti, konstruktorji, dedovanje. • Podprogrami, dogodki, izjeme. • Napotki za dobro programiranje.	• Assignments, expressions, operators. • Control flow statements. • Arrays. • Methods, classes and objects, constructors, inheritance. • Subroutines, events, exceptions. Good programming practices.

Temeljni literatura in viri / Readings:

- Mrhar, P. (2002). Java 2 - prvi korak. Šempeter pri Gorici: Flamingo.
- Mesojedec, U. (1997). Java: Programiranje za Internet. Ljubljana: Pasadena.
- Mesojedec, U., Fabjan B. (2004). Java2: temelji programiranja. Ljubljana: Pasadena.
- Barnes, D.J., Kolling, M. (2004). Objects First with Java - A Practical Introduction using BlueJ, 2nd Edition, Prentice Hall/Pearson Education.
- Wirth, N. (1985). Računalniško programiranje I. Ljubljana: DMFA.
- Wirth, N. (1985). Računalniško programiranje II. Ljubljana: DMFA.
- Učno razvojno okolje za Javo - BlueJ (URL: http://www.bluej.org/index.html).

Cilji in kompetence:	**Objectives and competences:**
Učna enota prispeva k razvoju naslednjih splošnih in predmetno specifičnih kompetenc: • usposobljenost za samostojno in avtonomno uporabo, nadzor in vzdrževanje informacijsko komunikacijske tehnologije v organizaciji • poznavanje tehnologij za spletno programiranje na strani klienta in strežnika ter razvoj aplikacij • sposobnost zapisati problem v obliki algoritma in pretvorba algoritma v računalniški program z uporabo sodobnih programskih orodij • razumevanje in uporaba računalniških sistemov in arhitektur	*The module contributes to the following general and module specific competences:* • competence for independent and autonomous use, monitoring and maintenance of information communication technology in an institution • knowledge of client and server side web programming technologies and applications development • the ability to write the problem in the form of an algorithm and converting the algorithm into a computer program using modern programming tools • understanding and use of computer systems and architectures

Predvideni študijski rezultati:	**Intended learning outcomes:**
Znanje in razumevanje: *Študent/študentka:* • razvije logično razmišljanja in sposobnosti načrtovanja programov • razume pomen načrtovanja in testiranja programske opreme • zmore dekompozicijo večjega problema na več manjših in obvladljivih	Knowledge and understanding: *The student:* • develops the ability of logical thinking and designing computer programs • understands the importance of software design and testing

• razume, kako računalnik deluje na logični ravni • operativno pozna programski jezik Java	• is able to decompose a bigger problem into a set of smaller ones that are easier to handle • understands the logical principles of computers • is able to write programs in Java

Metode poučevanja in učenja:	**Learning and teaching methods:**
• *predavanja* z aktivno udeležbo študentov (razlaga, diskusija, vprašanja, primeri, reševanje problemov) • *vaje,* kjer bodo študentje na konkretnih problemih ponovili, utrdili in dodatno osvetlili pojme in metode, spoznane na predavanjih • *kolokviji:* z njimi bodo študentje stimulirani, da sproti študirajo snov, ki bo obravnavana na predavanjih in vajah • *seminarska naloga* bo študente naučila samostojnega reševanja praktičnih problemov v programiranju	• *lectures* with active student participation (explanation, discussion, questions, examples, problem solving) • *lab work,* during which the students will use practical problems to repeat and strengthen the topics and methods presented at the lectures • *midterm exams* will stimulate the students to study concurrently with lectures and lab work • *student project* will prepare the students to autonomously solve practical programming problems

Načini ocenjevanja:	Delež (v %) / Weight (in %)	**Assessment:**
Način (pisni izpit, ustno izpraševanje, naloge, projekt): • pisni izpit	100	Type (examination, oral, coursework, project): • written exam