	UČNI NAČRT PREDMETA/COURSE SYLLABUS
Predmet	**Programiranje I**
Course title	**Programming I**

Študijski program in stopnja Study programme and level	**Študijska smer Study field**	**Letnik Academic year**	**Semester Semester**
Poslovna informatika / 1. stopnja	Računalništvo in informatika	1. letnik	1.
Business Informatics / 1st Cycle	Computer and Information Science	1st year	1st

Vrsta predmeta/Course type	obvezni/obligatory
Univerzitetna koda predmeta/University course code	1_RI_1_UN4

Predavanja Lectures	**Seminar Seminar**	**Sem. vaje Tutorial**	**Lab. vaje Laboratory work**	**Teren. vaje Field work**	**Samost. delo Individ. work**	**ECTS**
45			45		85	7

Nosilec predmeta/Lecturer:	doc. dr. Sebastian Lahajnar

Jeziki/ Languages:	**Predavanja/Lectures:**	slovenski/Slovenian
	Vaje/Tutorial:	slovenski/Slovenian

Pogoji za vključitev v delo oz. za opravljanje študijskih obveznosti:	**Prerequisites:**
• Vpis v prvi letnik študijskega programa. • Študent mora pred pristopom k izpitu izdelati vse programe, ki so predpisani v okviru laboratorijskih vaj.	• The prerequisite for inclusion is enrolment in the first year of study. • Students have to successfully develop all programs necessary in the scope of laboratory work before the examination.

Vsebina:	**Content (Syllabus outline):**
• *Uvod*: Splošno o programski opremi (v nadaljevanju PO), interakcija med nivoji PO v računalniškem sistemu, dokumentacija PO, knjižnice PO, standardi razvoja PO, kakovost PO, izdelava in implementacija PO. • *Osnove programiranja*: Splošno o programiranju, algoritem, proces	• *Introduction*: General information about software, interaction among levels of software in the computer system, documentation of software, libraries of software, standards of development of software, quality of software, preparation and implementation of software.

programiranja, značilnosti objektno usmerjenega programiranja. • *Predstavitev programskega jezika Java*: Splošno o Javi, izvajalno okolje Jave, značilnosti Jave, uporaba integriranih razvojnih okolij (v nadaljevanju IRO) za delo z Javo, namestitev IRO, osnove uporabe IRO. • *Temeljni elementi*: Spremenljivke, izrazi, stavki, bloki, simbolična imena, rezervirane besede, metode, dobesedne vrednosti. • *Osnovni podatkovni tipi*: Splošno o podatkovnih tipih, napoved, vzpostavitev in doseg, konstante, predstavitev celih in realnih števil, znakov ter logičnih vrednosti, implicitno in eksplicitno pretvarjanje podatkovnih tipov. • *Operatorji*: Prireditveni stavek, pisanje izrazov, tipi in prioriteta operatorjev. • *Usmerjanje izvajanja programa:* Izbirni stavki (if,switch), Ponavljalni stavki (do-while, while, for, for-each), nadzor izvajanja zank (break, continue). • *Razredi in objekti:* Splošno o objektih in razredih, stanje in obnašanje razreda, sporočila, osnovna zgradba razreda, konstruktorji, metode. • *Sklicni podatkovni tipi*: Splošno o sklicnih podatkovnih tipih, polja, nizi znakov, predmeti, ovijalni razredi. • *Napredne lastnosti razredov*: statični člani razreda, dosegljivost elementov razreda, preoblaganje metod in konstruktorjev, parametri metod, rekurzija, zapis stanja razreda kot niz, zaščita stanja ali enkapsulacija, primerjava predmetov, preoblaganje konstruktorjev razreda. • *Paketi*: Splošno o paketih, definiranje paketa, izvedba in vzdrževanje paketne strukture. • *Dedovanje*: Splošni pojmi dedovanja (podrazred, nadrazred), mnogoličnost ali polimorfizem, povožene metode, abstraktni razred in abstraktne metode, vmesnik, vgnezdeni razred, konstruktor podrazreda. • *Obravnavanje izjem*: Izjemni dogodek, razredna hierarhija izjem, Blok »try catch	• *Programming basics*: General information about programming, algorithm, programming process, characteristics of objectively-oriented programming. • *Presentation of the Java programming language:* General information about Java, Java execution environment, use of integrated development environments for working with Java, installation of integrated development environments, basics of using integrated development environments. • *Fundamental elements*: Variables, expressions, sentences, blocks, symbolic names, reserved words, methods, literal values. • *Basic data types*: General information about data types, forecast, initiation and reach, constants, presentation of integer and real numbers, characters and logical values, implicit and explicit conversion of data types. • *Operators*: Assignment statement, writing expressions, types and priority of operators. • *Coaching program implementation*: Conditional statements (if, switch), loops (do-while, while, for, for-each), control of loop implementation (break, continue). • *Classes and objects*: General information about objects and classes, status and behaviour of class, basic structure of class, constructors, methods. • *Reference data types*: General information about reference data types, fields, strings of characters, objects, wrapper classes. • *Advanced characteristics of classes*: Static members of class, reachability of class elements, overloading methods and constructors, method parameters, recursion, record of state of class as set, protection of state or encapsulation, comparison of objects, overloading constructors of class. • *Packages*: General information about packages, defining packages,

finally«, prepoznavanje izjem, uporaba rezerviranih besed throw in throws, uporaba eksplicitno definiranih izjem.	implementation and maintenance of package structure. • *Inheritance*: General terms about inheritance (subclass, superclass), polymorphism, overriding methods, abstract class and abstract methods, nested class, subclass constructor. • *Dealing with exceptions*: Exceptional event, class hierarchy of exceptions, "try catch finally" block, identification of exceptions, use of reserved words throw and throws, use of *explicitly defined exceptions.*

Temeljna literatura in viri/Readings:

Temeljna literatura/Basic literature

- Mesojedec, U. in Fabjan, B. (2004). *Java 2: Temelji programiranja.* Pasadena.
- Schildt, H. (2022). *Java: A Beginner's Guide, Ninth Edition*. McGraw-Hill Education.

Priporočljiva literatura/Recommended literature

- Schildt, H. (2018) *Java: The Complete Reference, Eleventh Edition*. McGraw-Hill Education.
- Farrell, J. (2014). *Java programming*. Course Technology, Cengage Learning.

Cilji in kompetence:	**Objectives and competences:**
Učna enota prispeva predvsem k razvoju naslednjih splošnih in specifičnih kompetenc: • poznavanje in razumevanje procesov v tehniško-tehnološkem ter poslovnem okolju in sposobnost za njihovo analizo, sintezo in predvidevanje rešitev ter njihovih posledic, • sposobnost definiranja, razumevanja in ustvarjalnega reševanja strokovnih izzivov na področjih računalništva in informatike, • usposobljenost za pridobivanje novih in poglabljanje pridobljenih strokovnih znanj računalništva in informatike, • usposobljenost za analizo in načrtovanje sistemov, • zmožnost opisati dano situacijo s pravilno uporabo matematičnih in računalniških simbolov ter zapisov, • praktično znanje in veščine pri razvoju programske in strojne opreme ter informacijskih tehnologij, ki so potrebne za uspešno delo na strokovnem področju računalništva in informatike (programiranje, računalniška arhitektura, omrežja itd.),	*The learning unit mainly contributes to the development of the following general and specific competences:* • knowledge and understanding of processes in the technical-technological and business environment, as well as the ability for their analysis, synthesis and prediction of the solutions and their consequences, • the ability to define, understand and creatively solve professional challenges in the fields of computer science and informatics, • the ability to acquire new and deepen the acquired professional knowledge of computer science and informatics, • being qualified to analyze and design systems, • the ability to describe the given situation with a proper use of mathematical and computer symbols and records, • practical knowledge and skills in the development of software and hardware

• usposobljenost za analizo in razvoj strojne in programske opreme, • poznavanje zmožnosti in omejitev informacijskih tehnologij.	and information technologies necessary for successful work in the field of computer science and informatics (programming, computer architecture, networks, etc.), • being qualified for the analysis and development of hardware and software, • knowing the capabilities and limitations of information technologies.

Predvideni študijski rezultati:	**Intended learning outcomes:**
Študent/študentka: • pozna in razume osnovne značilnosti programskih jezikov, še zlasti jezikov 3. generacije, • pozna postopke priprave in izvajanja programa, • pozna in razume prednosti objektno usmerjenega programiranja in ta pristop uporablja pri programiranju, • pozna in uporablja programski jezik Java za reševanje enostavnejših programskih problemov, • pozna in uporablja osnovne algoritme za reševanje nekaterih tipičnih programskih problemov (recimo dvojiško iskanje elementov v tabeli ali urejanje nizov), • pozna in v svojih rešitvah uporablja obstoječe (splošne) razrede in njihove metode.	***Students:*** • know and understand the basic characteristics of programming languages, especially the 3rd generation languages, • know the process of preparation and implementation of the program, • know and understand the advantages of objectively oriented programming and use this approach in programming, • know and use the Java programming languages for solving simpler programming problems, • know and use the basic algorithms for solving certain typical programming problems (e.g. binary search for elements in a table or arrangement of strings), • know and use the existing (general) classes and their methods in solutions.

Metode poučevanja in učenja:	**Learning and teaching methods:**
• *predavanja* z aktivno udeležbo študentov (razlaga, diskusija, vprašanja, primeri, reševanje problemov), • *laboratorijske vaje:* refleksija izkušenj, praktično reševanje več tipičnih problemov na računalniku, predstavitev in zagovor programskih rešitev, diskusija, sporočanje povratne informacije.	• *lectures* with active student participation (explanation, discussion, questions, examples, problem solving), • *laboratory work:* reflection on experience, practical solving of several typical problems on a computer, presentation and defence of programming solutions, discussion, feedback.

Načini ocenjevanja:	Delež (v %) Weight (in %)	**Assessment:**
Načini: • izpit Ocenjevalna lestvica: ECTS.	100 %	Types: • exam Grading scheme: ECTS.