

	<b>UČNI NAČRT PREDMETA/COURSE SYLLABUS</b>
<b>Predmet</b>	<b>Programiranje II</b>
<b>Course title</b>	<b>Programming II</b>

<b>Študijski program in stopnja</b> <b>Study programme and level</b>	<b>Študijska smer</b> <b>Study field</b>	<b>Letnik</b> <b>Academic year</b>	<b>Semester</b> <b>Semester</b>
Poslovna informatika / I. stopnja	Poslovna informatika	I. letnik	2.
Business Informatics / I <sup>st</sup> Cycle	Business Informatics	I <sup>st</sup> year	2 <sup>nd</sup>

**Vrsta predmeta/Course type** obvezni/obligatory

**Univerzitetna koda predmeta/University course code** I\_PI\_I\_UN6

<b>Predavanja</b>	<b>Seminar</b>	<b>Sem.</b>	<b>Lab. vaje</b>	<b>Teren.</b>	<b>Samost.</b>	<b>ECTS</b>
<b>Lectures</b>	<b>Seminar</b>	<b>vaje</b>	<b>Laboratory work</b>	<b>vaje</b>	<b>delo</b>	
		<b>Tutorial</b>		<b>Field work</b>	<b>Individ. work</b>	
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:**

<ul style="list-style-type: none"> <li>• Vpis v prvi letnik študijskega programa.</li> <li>• Študent mora pred pristopom k izpitu izdelati vse programe, ki so predpisani v okviru laboratorijskih vaj.</li> </ul>	<ul style="list-style-type: none"> <li>• The prerequisite for inclusion is enrolment in the first year of study.</li> <li>• Students have to successfully develop all programs necessary in the scope of laboratory work before the examination.</li> </ul>
--	---

**Vsebina:**

**Content (Syllabus outline):**

<ul style="list-style-type: none"> <li>• <i>Uvod:</i> Ponovitev osnovnega poznavanja programskega jezika Java – osnovno o objektno orientiranem programiranju, temeljni elementi Jave (spremenljivke, izrazi, stavčni blok, ključne besede, dobesedne vrednosti), podatkovni tipi, nadzor izvajanja programa, objekti, paketi, dedovanje, obravnavanje izjem.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Introduction:</i> Revision of the basic knowledge of the Java programming language - general information about objectively oriented programming, fundamental elements of Java (variables, expressions, statement block, key words, literal values), data types, supervision of program implementation,</li> </ul>
--	---

<ul style="list-style-type: none"> <li>• <i>Dokumentiranje javanskih programov</i>: namen in cilji, orodje Java Doc.</li> <li>• <i>Refleksija</i>: Koncept refleksije, razred Class, dostop do elementov razreda preko uporabe razreda Class, uporaba koncepta.</li> <li>• <i>Optimizacija kode - nitno programiranje</i>: Uporaba glavne niti, izdelava in uporaba lastnih niti, nadzor nad večnitnim delovanjem, prioritete in sinhronizacija izvajanja niti.</li> <li>• <i>Datoteke in tokovi</i>: Razredi za delo z datotekami, delo z datotekami, tokovi števil in znakov, delo s tokovi.</li> <li>• <i>Serializacija in deserializacija</i>: Splošno o konceptu, namen in delo s konceptom.</li> <li>• <i>Grafični uporabniški vmesnik (GUI)</i>: Splošno, sestava.</li> <li>• <i>Abstract Windowing Toolkit (AWT)</i>: Splošno o AWT, paket awt., dogodkovni model, sestava, gradniki, uporaba v programu.</li> <li>• <i>SWING</i>: Splošno o SWING-u, sorodstvo z AWT, paket javax., sestava, gradniki, uporaba v programu, izdelava GUI z uporabo grafičnega vmesnika IRO (integriranega razvojnega okolja).</li> <li>• <i>JAVA FX</i>: Splošno o Java FX, sorodstvo z AWT in SWING, paket javax., gradniki, uporaba v programu, izdelava GUI z uporabo grafičnega vmesnika IRO,</li> <li>• <i>JDBC (Java Database Connectivity)</i>: Splošno o relacijskih podatkovnih bazah, vmesnik JDBC, vzpostavitev povezave, izdelava poizvedb, izvedba transakcij.</li> </ul>	<p>objects, batches, inheritance, dealing with exceptions.</p> <ul style="list-style-type: none"> <li>• <i>Documenting Java programs</i>: Purpose and goals, the Java Doc tool.</li> <li>• <i>Reflection</i>: Concept of reflection, Class, access to class elements using Class, use of concept.</li> <li>• <i>Optimisation of code - threaded programming</i>: Use of the main thread, preparation and use of own threads, supervision of multi-thread functioning and synchronisation of using threads.</li> <li>• <i>Files and streams</i>: Classes for working with files, working with files, streams of numbers and characters, working with streams.</li> <li>• <i>Serialisation and deserialisation</i>: General information about the concept, purpose and working with the concept.</li> <li>• <i>Graphical user interface (GUI)</i>: General information, structure.</li> <li>• <i>Abstract Windowing Toolkit (AWT)</i>: General information about AWT, AWT package, event model, structure, building blocks, use in the program.</li> <li>• <i>SWING</i>: General information about SWING, similarities with AWT, the javax package, structure, building blocks, use in the program, preparation of GUI using as graphical user interface.</li> <li>• <i>Java FX</i>: General information about Java FX, similarities with AWT and SWING, the javax package, building blocks, use in the program, preparation of GUI using a graphical interface,</li> <li>• <i>JDBC (Java Database Connectivity)</i>: General information about relational databases, JDBC interface, connection creation, query making, transaction execution.</li> </ul>
--	--

### Temeljna literatura in viri/Readings:

#### Temeljna literatura/Basic literature

- Mesojedec, U. in Fabjan, B. (2004). *Java 2: Temelji programiranja*. Pasadena.
- Schildt, H. (2018) *Java: The Complete Reference, Eleventh Edition*. McGraw-Hill Education.

#### Priporočljiva literatura/Recommended literature

- Schildt, H. (2018). *Java: A Beginner's Guide, Eighth Edition*. McGraw-Hill Education.
- Farrell, J. (2014). *Java programming*. Course Technology, Cengage Learning.

**Cilji in kompetence:**

*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 poglobljanje 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.),
- usposobljenost za analizo in razvoj strojne in programske opreme,
- poznavanje zmožnosti in omejitev informacijskih tehnologij.

**Objectives and competences:**

*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 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:****Študent/študentka:**

- dobi pregled nad razvojnim okoljem za programski jezik java,
- dobi pregled in obvlada uporabo javanskega programskega vmesnika (API-ja),
- se spozna s postopki za pisanje učinkovitih javanskih programov,
- obvlada dokumentiranje javanskih programov,

**Intended learning outcomes:****Students:**

- review the development environment for the Java programming language,
- review and master the use of the Java application programming interface (API),
- are familiarised with procedures for writing effective Java programs,
- master the documentation of Java programs,
- master the coding of simple network applications,

<ul style="list-style-type: none"> <li>• obvlada pisanje preprostih mrežnih aplikacij,</li> <li>• obvlada pisanje preprostih grafičnih programov in grafičnih vmesnikov,</li> <li>• pozna osnove vključevanja javanskih programov v operacijski sistem in aplikacijske programe,</li> <li>• pozna prednosti in omejitve, ki jih prinaša prenosljivost javanskih programov,</li> <li>• pozna in uporablja knjižnico JDBC za dostop do relacijskih podatkovnih baz.</li> </ul>	<ul style="list-style-type: none"> <li>• master the coding of simple graphical programs and graphical interfaces,</li> <li>• know the basics of including Java programs in the operating system and application programs,</li> <li>• know the advantages and limitations of the portability of Java programs,</li> <li>• know and use the JDBC library to access relational databases.</li> </ul>
--	---

**Metode poučevanja in učenja:**

**Learning and teaching methods:**

<ul style="list-style-type: none"> <li>• <i>predavanja</i> z aktivno udeležbo študentov (razlaga, diskusija, vprašanja, primeri, reševanje problemov),</li> <li>• <i>laboratorijske vaje</i>: refleksija izkušenj, praktično reševanje več tipičnih problemov na računalniku, predstavitev in zagovor programskih rešitev, diskusija, sporočanje povratne informacije.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>lectures</i> with active student participation (explanation, discussion, questions, examples, problem solving),</li> <li>• <i>laboratory work</i>: reflection on experience, practical solving of several typical problems on a computer, presentation and defence of programming solutions, discussion, feedback.</li> </ul>
---	---

**Načini ocenjevanja:**

Delež (v %)

Weight (in %)

**Assessment:**

<p>Načini:</p> <ul style="list-style-type: none"> <li>• izpit</li> <li>• izdelava, predstavitev in zagovor seminarske naloge</li> </ul> <p>Ocenjevalna lestvica: ECTS.</p>	<p>60 %</p> <p>40 %</p>	<p>Types:</p> <ul style="list-style-type: none"> <li>• exam</li> <li>• preparation, presentation and defence of the seminar paper</li> </ul> <p>Grading scheme: ECTS.</p>
--	-------------------------	---