Univerza v Mariboru

Fakulteta za naravoslovje in matematiko

UČNI NAČRT PREDMETA / SUBJECT SPECIFICATION	
Predmet:	Principi programskih jezikov
Subject Title:	Principles of Programming Languages

Študijski program Study programme	Študijska smer Study option	Letnik Year	Semester Semester
Enovit magistrski študijski program druge stopnje Predmetni učitelj		5.	9
Five-year master's degree program Subject Teacher		5.	9

Univerzitetna koda predmeta / University subject code:

Predavanja Lectures	Seminar Seminar	Seminarske vaje Tutorial	Laborat. vaje Lab work	Terenske vaje Field work	Samostojno delo Individual work	ECTS
30		2	28		120	**6**

Nosilec predmeta / Lecturer: Marjan Mernik

Jeziki / Languages:	**Predavanja / Lecture:**	slovenski / Slovene
	Vaje / Tutorial:	slovenski / Slovene

Pogoji za vključitev v delo oz. za opravljanje študijskih obveznosti:	**Prerequisites:**
Ni pogojev.	None.

Vsebina:	**Content (Syllabus outline):**
• Uvod v programske jezike: neformalna definicija programskih jezikov, delitve programskih jezikov, kratka zgodovina programskih jezikov. • Vrednosti in tipi: delitev tipov, statično in dinamično preverjanje tipov, ekvivalenca tipov, vrste izrazov. • Pomnilnik: spremenljivka, shranljive vrednosti, življenjska doba spremenljivk, vrste ukazov, izrazi s stranskimi učinki. • Povezovanje: povezljive vrednosti, statični in dinamični doseg, vrste deklaracij, bločni ukazi in bločni izrazi, kvalifikacijski princip.	• Introduction to programming languages: informal definition of programming languages, classification of programming languages, brief history of programming languages. • Values and types: type classification, static and dynamic type checking, type equivalence, kind of expressions. • Storage: variable, storable values, variable lifetime, kind of commands, expressions with side effects. • Binding: bindable values, static and dynamic scope, kind of declarations, block commands and block expressions, the

• Abstrakcije: princip abstrakcije, vrste abstrakcij, mehanizmi prenosa parametrov, korespondenčni princip, dosledni in normalni izračun. • Ograjevanje: paketi, abstraktni tipi, objekti in razredi, generični moduli. • Sistemi tipov: monomorfni in polimorfni sistem tipov, vrste polimorfizma. Generiki v programskem jeziku Kotlin. • Objektno usmerjeno programiranje: objekt, razred, meta-razred. Razredni objektno usmerjeni jeziki in prototipni objektno usmerjeni jeziki. • Vrste dedovanja: enkratno/večkratno, razredno/objektno, striktno/nestriktno, urejeno/neurejeno, dinamično/selektivno, običajno/mixin. Vgnezdeni razredi v programskem jeziku java. • Funkcijsko programiranje: značilnosti funkcijskih jezikov, uvod v programski jezik lisp/haskell. • Uvod v programski jezik Kotlin.	qualification principle. • Abstractions: abstraction principle, kind of abstractions, parameter passing mechanisms, the correspondence principle, eager and normal evaluation. • Encapsulation: packages, abstract types, objects and classes, generic modules. • Type systems: monomorphic and polymorphic type systems, kind of polymorphisms. Kotlin generics. • Object-Oriented programming: object, class, meta-class. Class-based object-oriented languages and prototype-based object-oriented languages. • Kinds of inheritance: single/multiple, class-based/object-based, strict/non-strict, ordered/un-ordered, dynamic/selective, ordinary/mixin. Nested Java classes. • Functional programming: characteristics of functional languages, introduction to Lisp/Haskell programming language. • Introduction to programming language Kotlin.

Temeljni študijski viri / Textbooks:

- K.C. Louden: Programming Languages: Principles & Practices, Third Edition, Cenage Learning, 2013.
- P. Sestoft: *Programming Language Concepts,* Springer, Berlin, 2012.
- D. A. Watt: *Programming Language Design Concepts*, John Wiley, Chichester, 2004.

Cilji:	**Objectives:**
Cilj tega predmeta je razumevanje osnovnih konceptov programskih jezikov in razumeti bistvene razlike med funkcijskim, proceduralnim in objektno usmerjenim programiranjem.	The objective of this course is to understand the basic concepts of programming languages and to understand essential differences between functional, procedural and object-oriented programming.

Predvideni študijski rezultati:	**Intended learning outcomes:**
Znanje in razumevanje: Po zaključku tega predmeta bo študent sposoben • identificirati slabosti in prednosti posameznega programskega vzorca, • izbrati primeren programski jezik za rešitev dane naloge, • razumeti koncepte programskih jezikov, s pomočjo katerih se bo hitreje naučil novega programskega jezika, • razumeti razlike med statičnim in dinamičnim tipiziranjem, • razumeti različne oblike dodeljevanja pomnilnika, • razumeti različne tehnike prenosa parametrov, • razumeti različne oblike polimorfizma. Prenosljive/ključne spretnosti in drugi atributi: • *Spretnosti komuniciranja:* ustni zagovor laboratorijskih vaj, pisno izražanje pri pisnem izpitu. • *Uporaba informacijske tehnologije:* uporaba različnih prevajalnikov in interpreterjev. • *Reševanje problemov:* načrtovanje in implementacija programov z uporabo različnih programskih vzorcev.	Knowledge and understanding: On completion of this course the student will be able to • identify shortcomings and advantages of particular programming language, • select suitable programming language to solve particular problem, • understand concepts of programming languages with the aim to quicker learn new programming language, • understand differences between static and dynamic typing, • understand different ways of memory allocation, • understand different parameter passing techniques, • understand different forms of polymorphisms. Transferable/Key skills and other attributes: • *Communication skills:* oral lab work defence, manner of expression at written examination. • *Use of information technology:* use of different compilers and interpreters. • *Problem solving:* program design and implementation using different programming paradigms.

Metode poučevanja in učenja:	**Teaching and learning methods:**
• predavanja, • laboratorijske vaje, •	• lectures, • lab work, •

Načini ocenjevanja:	Delež (v %) / Weight (in %)	**Assessment methods:**
Sprotni način • laboratorijske vaje, • 1. kolokvij,	50 25	**Constant assessment methods** • lab work, • 1st midterm examination,

• 2. kolokvij.	25	• 2nd midterm examination.

Opomba:
Kolokvija se lahko nadomestita s pisnim izpitom v deležu 50 %..
Note:
The midterm examinations may be replaced by written exam in the weight of 50%..

Reference nosilca / Lecturer's references:

1. KOSAR, Tomaž, ZHENLI, Lu, MERNIK, Marjan, HORVAT, Marjan, ČREPINŠEK, Matej. A case study on the design and implementation of a platform for hand rehabilitation. Applied sciences, ISSN 2076-3417, 2021, vol. 11, iss. 1, str. 1-21, ilustr., doi: 10.3390/app11010389. [COBISS.SI-ID 45500163]
2. KOS, Tomaž, MERNIK, Marjan, KOSAR, Tomaž. A tool support for model-driven development: an industrial case study from a measurement domain. Applied sciences, ISSN 2076-3417, 2019, vol. 9, iss. 21, str. 1-20, ilustr., doi: 10.3390/app9214553. [COBISS.SI-ID 22712342]
3. KOSAR, Tomaž, GABERC, Sašo, CARVER, Jeffrey C., MERNIK, Marjan. Program comprehension of domain-specific and general-purpose languages: replication of a family of experiments using integrated development environments. Empirical software engineering, ISSN 1382-3256, 2018, vol. 23, iss. 5, str. 2734-2763, doi: 10.1007/s10664-017-9593-2. [COBISS.SI-ID 21123606]