NOVO

Univerza v Mariboru

Fakulteta za naravoslovje in matematiko

FERI Fakulteta za elektrotehniko, računalništvo in informatiko

UČNI NAČRT PREDMETA / COURSE SYLLABUS	
Predmet:	Programski vzorci
Course title:	Programming Paradigms

Študijski program in stopnja **Study programme and level**	**Študijska smer** **Study field**	**Letnik** **Academic year**	**Semester** **Semester**
Izobraževalno računalništvo 1. stopnja UN		3.	poletni
Educational Computer Science 1st cycle Academic undergraduate			Spring

Vrsta predmeta / Course type	Izbirni predmet
Univerzitetna koda predmeta / University course code:	

Predavanja **Lectures**	**Seminar** **Seminar**	**Vaje** **Tutorial**	**Klinične vaje** **work**	**Druge oblike** **študija**	**Samost. delo** **Individ. work**	**ECTS**
30		**30**			**120**	**6**

Nosilec predmeta / Lecturer:	Janez Brest

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

Pogoji za vključitev v delo oz. za opravljanje študijskih obveznosti:	**Prerequisits:**
Znanje, vključeno v predmete Programiranje I, Programiranje II in Diskretne strukture.	Knowledge included in the courses Programming 1, Programming 2, and Discrete Mathematics.

Vsebina:	**Content (Syllabus outline):**
• Uvod: neformalna definicija programskih jezikov, delitve in zgodovina programskih jezikov. • Vrednosti in tipi: delitev tipov, preverjanje tipov, ekvivalenca tipov, vrste izrazov, vrednosti prvega in drugega razreda, princip polnosti tipa. • Pomnilnik: spremenljivka, model pomnilnika, selektivno ali popolno ažuriranje, shranljive vrednosti, življenjska doba,	• Introduction: definition of programming languages, classification and history of programming languages. • Values and types: type classification, type checking, type equivalence, types of expressions, first-class values, second-class values, type completeness principle. • Storage: variable, storage, total updating or selective updating, storable value, lifetime, types of commands, and

vrste ukazov, izrazi s stranskimi učinki.

- Povezovanje: povezovanje, okolje in blok, povezljive vrednosti, vrste deklaracij, statični in dinamični doseg, bločni izrazi in bločni ukazi, kvalifikacijski princip.
- Abstrakcije: princip abstrakcije, funkcijske abstrakcije proceduralne abstrakcije, izbirne abstrakcije in generične abstrakcije, kopirni mehanizmi parametrov, korespondenčni princip, dosledni in normalni izračun.
- Pojem "kapsuliranja", koncepti, ki podpirajo modularnost.
- Sistemi tipov: monomorfni in polimorfni sistemi tipov, vrste polimorfizma, pojem "sekvencer", vrste sekvencerjev.
- Osnovni koncepti objektno usmerjenega programiranja, sistemi prvega, drugega, tretjega in petega nivoja, hierarhija objektnih jezikov, vrste dedovanja.
- Funkcijsko programiranje, programski jezik Lisp/Haskell.
- Logično programiranje, programski jezik prolog.
- Paralelno programiranje, konstrukti za podporo paralelnosti, programski jeziki za paralelno programiranje.

expressions with side effects.

- Binding: binding, environment and block, bindable values, types of declarations, static scope, dynamic scope, block expression, qualification principle.
- Abstraction: abstraction principle, functional abstraction, procedural abstraction, selector abstraction, generic abstraction, copy parameter mechanism, correspondence principle, applicative-order evaluation, normal-order evaluation.
- Term "encapsulate", concepts of modularity.
- Type systems: monomorphic and polymorphic type systems, types of polymorphism, term "sequencer", types of sequencers.
- Basic concepts of object oriented programming, first, second, third, and fifth level systems, hierarchy of object languages, types of inheritance.
- Functional programming, programming language Lisp/Haskell.
- Logic programming, programming language prolog.
- Parallel programming, parallelism constructs, programming languages for parallel programming.

Temeljni literatura in viri / Readings:

- V. Žumer, M. Mernik, *Principi programskih jezikov*, Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko, Maribor 2003.
- R. Sethi: *Programming Languages: Concepts and Constructions*, Second Edition, Addison-Wesley, Reading, 1996.
- D. A. Watt, *Programming Language Concepts and Paradigms*. Prentice-Hall International, New York, 1990.
- D. A. Watt, Programming Language Design Concepts, John Wiley, Chichester, 2004.
- M. Gabbrielli, S. Martini, Programming Languages: Principles and Paradigms, Springer, 2010.

Cilji in kompetence:

Cilj predmeta je seznaniti študente s koncepti programskih jezikov in spoznati programske vzorce funkcijskega, logičnega in objektno usmerjeno programiranja.

Objectives and competences:

The objective of this course is to acquaint students with concepts of programming languages and programming paradigms of functional, logical, imperative and object oriented programming.

Predvideni študijski rezultati:

Znanje in razumevanje:

Po zaključku tega predmeta bo študent sposoben

- identificirati prednosti in slabosti posameznega vzorca,
- izbrati primeren programski jezik za rešitev dane naloge,
- razumeti koncepte programskih jezikov s pomočjo katerih se bodo hitreje naučili novega programskega jezika,
- razumeti razlike med statičnim in dinamičnim tipiziranjem,
- razumeti različne oblike dodeljevanja pomnilnika, dosega,
- razumeti različne oblike prenosa parametrov,
- razumeti pomen abstrakcij,
- razumeti različne oblike polimorfizma.

Intended learning outcomes:

Knowledge and understanding:

On completion of this course the student will be able to

- identify advantages and disadvantages of some paradigms,
- to choose suitable program language for solving defined problem,
- understand concepts of programming languages for quickly learning a new programming language,
- understand differences between static and dynamic typing
- understand different storage and scope model,
- understand parameter mechanisms,
- understand term abstraction,
- understand different types of polymorphism.

Prenosljive/ključne spretnosti in drugi atributi:

- Spretnosti komuniciranja: ustni zagovor laboratorijskih vaj, pisno izražanje pri pisnem izpitu.
- Uporaba informacijske tehnologije: uporaba objektno usmerjenih, funkcijskih in logičnih programskih jezikov.
- Reševanje nalog: uporaba programskih vzorcev pri načrtovanju in implementaciji programov.

Transferable/Key skills and other attributes:

- Communication skills: oral lab work defence, manner of expression at written examination.
- Use of information technology: use object oriented, functional and logic programming languages.
- Problem solving: using programming paradigms for program design and implementation.

Metode poučevanja in učenja:

- predavanja,
- seminarske vaje,
- laboratorijske vaje,
- kvizi.

Learning and teaching methods:

- lectures,
- tutorials,
- lab work,
- quizzes.

Načini ocenjevanja:	Delež (v %) / Weight (in %)	Assessment:
• kvizi,	15 %	• quizzes,
• laboratorijske vaje,	35 %	• lab work,
• 1. vmesni pisni izpit,	16 %	• 1st midterm written exam,
• 2. vmesni pisni izpit,	17 %	• 2nd midterm written exam,
• 3. vmesni pisni izpit.	17 %	• 3rd midterm written exam.

Reference nosilca / Lecturer's references:

- ZAMUDA, Aleš, BREST, Janez. Vectorized procedural models for animated trees reconstruction using differential evolution. Information sciences, ISSN 0020-0255. 2014, vol. 278, str. 1-21, [COBISS.SI-ID 17793558].
- BREST, Janez, SEPESY MAUČEC, Mirjam. Self-adaptive differential evolution algorithm using population size reduction and three strategies. Soft computing, ISSN 1432-7643. 2011, vol. 15, no. 11, str. 2157-2174, [COBISS.SI-ID 14398230], [JCR, SNIP, WoS do 5. 8. 2014: št. citatov (TC): 19, Scopus : št. citatov (TC): 32]
- FISTER, Iztok, FISTER, Iztok, YANG, Xin-She, BREST, Janez. A comprehensive review of firefly algorithms. Swarm and evolutionary computation, ISSN 2210-6502, Dec. 2013, vol. 13, str. 34-46, doi: 10.1016/j.swevo.2013.06.001. [COBISS.SI-ID 17010454], [SNIP, Scopus do 8. 10. 2014: št. citatov (TC): 29]
- ZAMUDA, Aleš, BREST, Janez. Environmental framework to visualize emergent artificial forest ecosystems. Information sciences, ISSN 0020-0255. 2013, vol. 220, str. 522-540, doi: 10.1016/j.ins.2012.07.031. [COBISS.SI-ID 16157206], [JCR, SNIP, WoS do 6. 8. 2014: št. citatov (TC): 1, čistih citatov (CI): Scopus do 13. 8. 2014: št. citatov (TC): 3]
- BREST, Janez, GREINER, Sašo, BOŠKOVIĆ, Borko, MERNIK, Marjan, ŽUMER, Viljem. Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. IEEE transactions on evolutionary computation, ISSN 1089-778X. 2006, vol. 10, no. 6, str. 646-657. [COBISS.SI-ID 10376982], [JCR, SNIP, WoS do 2. 11. 2014: št. citatov (TC): 352, Scopus: št. citatov (TC): 857]