NAPREDNA RAČUNALNIŠKA ORODJA

UČNI NAČRT PREDMETA/COURSE SYLLABUS

Predmet:	Napredna računalniška orodja
Course title:	High-end computing tools
Članica nosilka/UL Member:	UL FS

Študijski programi in stopnja	Študijska smer	Letnik	Semestri	Izbirnost
Strojništvo - razvojno raziskovalni program, prva stopnja, univerzitetni	Ni členitve (študijski program)	3. letnik	1. semester	izbirni

Univerzitetna koda predmeta/University course code:	0562777
Koda učne enote na članici/UL Member course code:	2036-U

Predavanja /Lectures	Seminar /Seminar	Vaje /Tutorials	Klinične vaje /Clinical tutorials	Druge oblike študija /Other forms of study	Samostojno delo /Individual student work	ECTS
30		30			40	4

Nosilec predmeta/Lecturer:	Janez Povh, Leon Kos

Izvajalci predavanj:	
Izvajalci seminarjev:	
Izvajalci vaj:	
Izvajalci kliničnih vaj:	
Izvajalci drugih oblik:	
Izvajalci praktičnega usposabljanja:	

Vrsta predmeta/Course	Splošni izbirni predmet /Elective general course

type:	

Jeziki/Languages:	Predavanja/Lectures:	Slovenščina
	Vaje/Tutorial:	Slovenščina

Pogoji za vključitev v delo oz. za opravljanje študijskih obveznosti:	**Prerequisites:**
Izpolnjevanje pogojev za vpis v Univerzitetni študijski program I. stopnje Strojništvo - Razvojno raziskovalni program.	Meeting the enrollment conditions for the Academic study programme of Mechanical Engineering - Research and Development program.

Vsebina:	**Content (Syllabus outline):**
Uvod v Mathematico: osnove dela z Mathematico, izvajanje temeljnih računskih operacij z Mathematico, simbolno računanje z Mathematico. 2. Mathematica: programiranje v jeziku Wolfram, vizualizacija podatkov in matematičnih funkcij, Mathematica v oblaku. 3. Uvod v Matlab: spoznavanje Matlab sintakse, izvajanje vhodno izhodnih operacij, izvajanje temeljnih računskih operacij v Matlabu. Matlab v oblaku. 4. Matlab: izvajanje zahtevnejših numeričnih izračunov (reševanje enačb, faktorizacija matrik, numerično integriranje in odvajanje). 5. Uvod v LaTeX: temeljne značilnosti LaTeX-a, predstavitev ključnih distribucij, urejevalniki teksta, prilagojeni za LaTeX, pisanje kompleksnih dokumentov v Latex, izdelava drsnic v okolju Beamer, LaTeX. sodelovalna okolja v oblaku (ShareLaTeX, OverLeaf). 6. Uvod v Github: Razvoj z Git po vejah in zlivanje datotek z zahtevki, sledenje in pregled kode, integracija tekstov in dokumentiranje z restuktuiranim tekstom. 7. Uvod v C++: osnove C++ sintakse, razvojna okolja za C++ (urejevalniki kode, prevajalniki, razhroščevalniki), kazalci, funkcije. 8. Osnove objektnega programiranja s	Introduction to Mathematica: environment fundamentals, basic numerical computing with Mathematica, symbolic computing with Mathematica. 2. Mathematica: programming in Wolfram language, data and functions visualization , Mathematica in the cloud. 3. Introduction to Matlab: Matlab syntax fundamentals, Input/Output operations, basic computing operations in Matlab, Matlab in the cloud. 4. Matlab: performing advanced numerical calculations (equation solving, matrix factorizations, numerical integration and derivation). 5. Introduction to LaTeX: LaTeX fundamentals, popular LaTeX distributions, text editors for LaTeX, preparing complex document with LaTeX, slides with Beamer, LaTeX collaborative cloud environments (ShareLaTeX, OverLeaf). 6. Introduction to Github: Development with Git by branches and merging with pull requests, tracking and code review, integrating texsts and documenting with restructured text. 7. Introduction to C++: C++ syntax fundamentals, C++ development environments (code editors, compilers, debuggers), pointers, functions. 8. Basics of object programming with C++: classes and objects, constructor, destructor, name mangling, inheritance,

C++: razredi in objekti, konstruktor, destruktor, prikrivanje, dedovanje, polimorfizmi. 9. Uvod v Linux: opis najbolj razširjenih Linux operacijskih sistemov, shell, terminal in konzola, osnovne operacije nad datotekami, navigiranje med direktoriji. 10. Linux: napredni ukazi, vhodno/izhodno preusmerjanje tokov, CentOS. 11. Uvod v visokozmogljivo računalništvo (HPC) : • Razumevanje HPC in njegovih možnih aplikacij. • Pregled razpoložljivih zmogljivosti, zgodovina, trendi. • Heterogeni računalniški sistemi - GPU vs. CPU. • HPC na FS – kako se prijaviti, kako oddati nalogo, kako spremljati potek dela. • Knjižnice za učinkovito paralelno računanje: Blas, LaPack, Eigen,… 12. Sočasnost, vzporednost in porazdeljeni sistemi • Sočasnost in vzporednost;. • Paralelizacija računskih nalog v Matlabu, Mathematici in Pythonu. • Pisanje C++ kode za HPC z deljenim spominom z uporabo OpenMP in za porazdeljene sisteme z uporabo vmesnika za pošiljanje sporočil (MPI). 13. Multifizika • Pregled modulov multifizike; priprava modelov CAD za simulacije. • Izdelava sheme v okolju Workbench; vezava vstopnih podatkov med moduli (geometrija, materialni parametri, vključevanje rezultatov prve analize kot vstopne podatke druge). • Predstavitev glavnih oblik konstrukcijskih modelov za simulacije (linijski nosilci, lupine, telesa). • Parametri porazdelitve problemov na HPC, vzporedni datotečni sistemi, Uporaba oddaljenega računanja (RSM). • Vizualizacija in VirtualGL • Ansys v oblaku. 14. Programska orodja za simulacije CFD	polymorphisms. 9. Introduction to Linux: description of the most common Linux distributions, shell, terminal and console, basic file operations, directory navigation. 10. Advanced Linux Commands: Input/Output redirections, CentOS 11. Introduction to High Performance Computing (HPC): • Understanding HPC and possible applications. • Overview of available capacities, history, trends. • Heterogeneous Computer Systems - GPU vs. CPU. • HPC on UL FME - how to apply, how to submit a job, how to monitor workflow. • Libraries for efficient parallel computing: Blas, LaPack, Eigen,… 12. Concurrency, parallelism and distributed systems • Concurrency and parallelism. • Parallelization of computational tasks in Matlab, Mathematica and Python. • Writing C++ code for shared memory HPC using OpenMP and for multi-node systems using message passing interface (MPI). 13. Multiphysics: • An overview of multiphysics modules; preparation of CAD models for simulations. • Creating a workflow in Workbench environment; binding input data between modules (geometry, material parameters, integration of the results of the first analysis as input data of the second). • Presentation of the main shapes of construction models for simulations (line beams, shells, bodies). • Parameters of problem parallelism in HPC, parallel file systems, using remote computing (RSM). • Visualization and VirtualGL • Ansys in the cloud. 14. Software tools for CFD simulations • Overview of codes, operations and manipulations of open fields.

• Pregled kode, operacije in manipulacije odprtega polja. • Predprocesiranje, izdelava in pretvorba mrež, domenska dekompozicija za HPC. • Solverji za CFD, zgorevanje, elektromagnetiko, mehaniko trdin. • Orodja za rekonstrukcijo, postprocesiranje in vizualizacijo. 15. Simulacije z odprtokodnimi okolji • Pregled modulov SALOME. • Uporaba zunanjih solverjev z modulom za paketno obdelavo na HPC. • Integracija in programiranje vmesnikov za solverje na HPC. • Programiranje CAD geometrije in mreženje. • Multifizika z Elmer in povezava Medcoupling. • ParaView in povezave z IOR.	• Preprocessing, generation and conversion of meshes, domain decomposition for HPC. • Solvers for CFD, combustion, electromagnetics and solids mechanics. • Tools for reconstruction, post-processing and visualization. 15. Simulations with open-source environments • Overview of SALOME modules. • Using external solvers with batch processing module for HPC. • Integration and programming of interfaces for solvers for HPC. • CAD geometry programming and meshing. • Elmer Multiphysics and Medcoupling • ParaView and IOR connectectivity.

Temeljna literatura in viri/Readings:

1. Hager, G. in Gerhard W. Introduction to high performance computing for scientists and engineers. CRC Press, 2010.
2. Eijkhout, V. Introduction to High-Performance Scientific Computing, http://pages.tacc.utexas.edu/~eijkhout/istc/istc.html
3. Latex for beginners, http://www.docs.is.ed.ac.uk/skills/documents/3722/3722-2014.pdf
4. Moore, H. MATLAB for Engineers. Pearson, 2017.
5. Moukaled-Mangani-Darwish, The Finite Volume Method in Computational Fluid Dynamics: An Advanced Introduction with OpenFOAM and MATLAB, Springer, 2016
6. Choudary, R.B. Introduction to ANSYS 10, 2/E I K International Publishing House; Second Edition, 2016
7. Magrab, E. B.., An Engineer's Guide to Mathematica, John Wiley & Sons, 2014
8. https://guides.github.com/introduction/git-handbook/
9. https://docs.salome-platform.org/latest/gui/GEOM/index.html
10. https://www.wolfram.com/language/fast-introduction-for-math-students/en/

Cilji in kompetence:	**Objectives and competences:**
Cilji: 1. Naučiti študente osnov Linuxa in kako uporabljati računalniško orodja za kreiranje kompleksnih dokumentov ter za sočasno projektno delo. 2. Naučiti študente osnov Matlaba, Mathematice in C++.	Objectives: 1. To teach students the basics of Linux and how to use computer tools to create complex documents and usage for collaborative tasks. 2. Introduce students to fundamentals of Matlab, Mathematica and C++. 3. Prepare students to be able using

3. Naučiti študente, da uporabljajo superračunalnike kot orodja za reševanje zahtevnih računskih problemov.
4. Naučiti študente, da znajo prenesti težke naloge iz svojih prenosnih ali namiznih računalnikov na superračunalnik in jih tam rešiti.
5. Naučiti študente, kako izvajati s sodobnimi računalniškimi orodji priprave, simulacije, analize in prikaz rezultatov različnih fizikalnih modelov.

Kompetence:

1. S1-RRP: Sposobnost za opredelitev, razumevanje in ustvarjalno reševanje strokovnih izzivov z uporabo naprednih računalniških orodij.
2. S2-RRP: Razvijanje sposobnosti kritičnega, analitičnega in sintetičnega mišljenja s poudarkom na reševanju različnih inženirskih problemov z naprednimi računalniškimi orodji in na kritični presoji dobljenih rešitev.
3. S5-RRP: Sposobnost uporabe IKT, še posebej superračunalniške tehnologije, za reševanje težkih računskih problemov s področja strojništva.
4. S4-RRP: Sposobnost strokovnega sporazumevanja in pisnega izražanja, vključno z uporabo tujega strokovnega jezika in naprednih računalniških orodij za pisanje kompleksnih dokumentov in za obvladovanje kompleksne dokumentacije, ki nastaja v projektnih timih.
5. P2-RRP: Študent se zna prijaviti na enega od slovenskih superračunalnikov in tam izvajati osnovne naloge v okolju Linux.
6. P2-RRP: Študent zna kreirati kompleksne dokumente v okolju Latex.
7. P2-RRP: Študent zna prenesti podatke o svojem problemu na superračunalnik in tam zagnati sodobna programska orodja za reševanje teh problemov.

supercomputers as a tools to solve complex computational problems;
4. To teach students how to model and transfer computationally demanding tasks from their laptops or desktop computers to supercomputers and solve them remotely.
5. To teach students how to prepare, simulate, analyze and visualise the results of various physical models using advanved computer tools;

Competences:

1. S1-RRP: The ability to define, understand and creatively solve professional challenges using advanced computing tools.
2. S2-RRP: Development of creative, analytical and synthetic thinking with an emphasis on (i) solving various engineering problems using advanced computer tools and (ii) on critical evaluations of the solutions.
3. S5-RRP: The ability to use ICT, especially supercomputer technology, to solve difficult computational problems related to mechanical engineering.
4. S4-RRP: Professional communication and writing communication skills, including the use of foreign technical language and advanced computer tools to write complex documents and to master complex documentation generated by project teams.
5. P2-RRP: Student can login to one of the Slovenian supercomputers and perform basic tasks in Linux environment.
6. P2-RRP: Student is able to create complex documents in the Latex environment.
7. P2-RRP: Student is able to transfer data of a computational problem to a supercomputer and run advanced software tools to solve these problems.
8. P2-RRP, P4-RRP: Student can write basic programs in C++, MATLAB, Mathematica, and utilizing advantage

8. P2-RRP, P4-RRP: Študent zna napisati osnovne programe v okoljih C++, MATLAB, Mathematica, ki izkoriščajo možnost paralelizacije na superračunalnikih. 9. P6-RRP: Študent zna uporabljati okolje Git za obvladovanje sprememb pri skupinskem razvoju programske kode ali druge dokumentacije. 10. P4-RRP: Študent zna uporabiti napredna računalniška orodja, kot so Ansys., OpenFOAM in Salome, za reševanje kompleksnih inženirskih problemov.	of parallelization on supercomputers. 9. P6-RRP: Student knows how to use the Git environment to control code or documentation changes within collaborative development environment. 10. P4-RRP: Student can use advanced computing tools, such as Ansys, OpenFOAM or Salome, to solve complex engineering problems.

Predvideni študijski rezultati:	**Intended learning outcomes:**
Znanja: Z1: Poglobljeno strokovno teoretično in praktično znanje na določenem področju, podprto s širšo teoretično in metodološko osnovo: razumevanje in sposobnost uporabe naprednih računalniških orodij za reševanje inženirskih problemov ter za obvladovanje kompleksne dokumentacije; Spretnosti: S1.1: Sposobnost uporabe nekaterih standardnih računalniških programov, ki jih študentje srečajo tekom študija, na superračunalniku in z njimi učinkovito reševati težke inženirske probleme. S1.2: Sposobnost samostojno izdelati kompleksne dokumente, vključno z drsnicami za predstavite, z orodjem LaTeX in Beamer in kako obvladovati kompleksno dokumentacijo z orodji Git. S1.4: Študentje se naučijo osnov programskega jezika C++ ter uporabe knjižnic za komunikacijo v okoljih z deljenim in porazdeljenim spominom. S1.4: Študentje se naučijo napisati preproste računalniške programev okoljih Mathematica, Matlab, C++, ki vključujejo paralelizacijo izvajanja nad sodobnimi superračunalniškimi arhitekturami.	Knowledge: Z1: In-depth professional theoretical and practical knowledge in a specific field, supported by a broader theoretical and methodological basis: understanding and ability to use advanced computer tools to solve engineering problems and to master complex documentation; Skills: S1.1: Ability to use some standard computer programs encountered by students during their studies, on a supercomputer to efficiently solve hard engineering problems. S1.2: Ability to produce complex documents on your own, including presentation sliders, with LaTeX, and to handle complex documentation using Git tools. S1.4: Students learn the basics of the C++ programming language and how to link libraries for processor communication in shared and distributed memory systems. S1.4: Students learn to write simple computer programs in Mathematica, Matlab and C++ languages, which include parallelization over modern supercomputing architectures.

Metode poučevanja in učenja:	Learning and teaching methods:
P1: Avditorna predavanja z reševanjem izbranih - za področje značilnih - teoretičnih in praktično uporabnih primerov. P2: Obravnava snovi po urejeni in vnaprej razloženi sistematiki. P3 Avditorne vaje, kjer se teoretično znanje s predavanj podkrepi z računskimi primeri. P4 Laboratorijske vaje z namenskimi didaktičnimi pripomočki: • namizni računalnik s primerno programsko oporemo, • programi za oddaljeni dostop, • okolje za pisanje programske kode v C++, • okolje za izvajanje analiz velepodatkov, • oddaljeni superračunalnik. P5 Uporaba študijskega gradiva v obliki: • Zapiskov in drsnic, • Tiskanih in e-knjig, • e-verzije predavanj (video predavanja). P6 Interaktivna predavanja. P8 Izdelava in predstavitev aplikativnih seminarskih nalog. P12 Individualizirane domače naloge v spletni učilnici. P13 Individualizirani kolokviji in izpiti s samodejnim popravljanjem. P15 Uporaba video vsebin kot priprava na predavanja in vaje.	P1 Auditorial lectures with solving selected domain-specific theoretical and applied use cases. P2 Presenting the content according to the explained systematics. P3 Auditorial tutorials, where the theoretical content from the lectures is supplemented with practical examples. P4 Laboratory exercises with special-purpose didactic devices: • Desktop computer with appropriate software, • Clients for remote access, • C++ programming environment • Big Data analysis tools , • supercomputer via remote access. P5 Application of different study materials: • lecture notes and slides, • printed and eBooks, • e-versions of lectures (video lectures). P6 Interactive Lectures. P8 Preparation and presentation of application oriented seminars. P12 Individualized homeworks in e-classrooms. P13 Individualized mid-term exams and self-evaluating exams. P15 Video contents for preparing students for lectures and tutorials.

Načini ocenjevanja:	Delež/ Weight	Assessment:
Teoretični izpit.	40,00 %	Examination.
Projekt.	20,00 %	Project.
Domače naloge.	20,00 %	Coursework.

Reference nosilca/Lecturer's references:

Janez Povh:

1. CRNKIĆ, Aladin, **POVH, Janez**, JAĆIMOVIĆ, Vladimir, LEVNAJIĆ, Zoran. Collective dynamics of phase-repulsive oscillatorssolves graph coloring problem. Chaos, ISSN 1054-1500, 2020, vol. 30, str. 033128-1-033128-10
2. DICKINSON, Peter J. C., **POVH, Janez**. A new approximation hierarchy for polynomial conic optimization. *Computational optimization and applications*, ISSN 0926-6003. [Print ed.], Jan. 2019, str. [1-31], ilustr. https://link.springer.com/article/10.1007%2Fs10589-019-00066-0, doi: 10.1007/s10589-019-00066-0. [COBISS.SI-ID 16466459];
3. MALOD-DOGNIN, Noël, PETSCHNIGG, Julia, WINDELS, Sam F. L., **POVH, Janez**, HEMMINGWAY, Harry, KETTELER, Robin, PRŽULJ, Nataša. Towards a data-integrated cell. *Nature communications*, ISSN 2041-1723, Feb. 2019, [Vol.] 10, f. 1-13, ilustr. https://www.nature.com/articles/s41467-019-08797-8, doi: 10.1038/s41467-019-08797-8. [COBISS.SI-ID 16484379];
4. LUŽAR, Borut, LEVNAJIĆ, Zoran, **POVH, Janez**, PERC, Matjaž. Community structure and the evolution of interdisciplinarity in Slovenia's scientific collaboration network. *PloS one*, ISSN 1932-6203, 2014, vol. 9, iss. 4, str. e94429-1-e94429-5, doi: 10.1371/journal.pone.0094429. [COBISS.SI-ID 20503816];
5. BURGDORF, Sabine, KLEP, Igor, **POVH, Janez**. *Optimization of polynomials in non-commuting variables*, (SpringerBriefs in mathematics (Print)). [S. l.]: Springer, cop. 2016. X, 111 str., graf. prikazi, tabele. ISBN 978-3-319-33336-6. ISBN 978-3-319-33338-0. [COBISS.SI-ID 2048381715];

Leon Kos:

1. **KOS, Leon**, PITTS, R. A., SIMIČ, G., BRANK, Matic, ANAND, H., ARTER, W. SMITER : a field-line tracing environment for ITER. *Fusion engineering and design*, ISSN 0920-3796. [Print ed.], Sep. 2019, vol. 146, pt. B, str. 1796-1800, ilustr. https://www.sciencedirect.com/science/article/pii/S092037961930359X?via%3Dihub, doi: 10.1016/j.fusengdes.2019.03.037.
2. ANAND, H., PITTS, R. A., VRIES, P. C. de, SNIPES, J. A., NESPOLI, F., LABIT, B., GALPERTI, C., CODA, S., BRANK, Matic, **KOS, Leon**. Experimental implementation of a real-time power flux estimator for the ITER first wall on the TCV tokamak. *Fusion engineering and design*, ISSN 0920-3796. [Print ed.], Oct. 2019, vol. 147, str. 1-7, ilustr. https://www.sciencedirect.com/science/article/pii/S0920379619307203#!, doi: 10.1016/j.fusengdes.2019.111242.
3. **KOS, Leon**, JELIĆ, Nikola, GYERGYEK, Tomaž, KUHN, S., TSKHAKAYA, David. Modeling and simulations of plasma and sheath edges in warm-ion collision-free discharges. *AIP advances*, ISSN 2158-3226, Oct. 2018, vol. 8, no 10, str. 1-23, ilustr. https://aip.scitation.org/doi/pdf/10.1063/1.5044664?class=pdf, doi: 10.1063/1.5044664
4. FALCHETTO, G. L., COSTER, D., COELHO, Roland J., **KOS, Leon**, KULOVEC, Simon, LENGAR, Igor, SNOJ, Luka, et al., ITM-TF Contributors ; ASDEX Upgrade Team ; JET-EFDA Contributors. The European Integrated Tokamak Modelling (ITM) effort : achievements and first physics results. *Nuclear fusion*, ISSN 0029-5515, 2014, iss. 4, vol. 54, 043018, doi: 10.1088/0029-5515/54/4/043018