



FIȘA DISCIPLINEI
ANUL UNIVERSITAR 2020 - 2021

1. Date despre program

1.1. Instituția de învățământ superior	UNIVERSITATEA DIN CRAIOVA
1.2. Facultatea	FACULTATEA DE AUTOMATICĂ, CALCULATOARE ȘI ELECTRONICĂ
1.3. Departamentul	AUTOMATICĂ ȘI ELECTRONICĂ (D28)
1.4. Domeniul de studii	INGINERIA SISTEMELOR
1.5. Ciclul de studii ¹⁾	LICENȚĂ
1.6. Specializarea/ Programul de studii	AUTOMATICĂ ȘI INFORMATICĂ APLICATĂ (cod L20601022010)
1.7. Forma de învățământ	CU FRECVENȚĂ

2. Date despre disciplină

2.1. Denumirea disciplinei		Practică de domeniu						
2.2. Titularul activităților de curs								
2.3. Titularul activităților de seminar/ laborator/ proiect		Ș.l. dr. ing. Lavinia Aurelian BĂDULESCU						
2.4. Anul de studiu	2	2.5. Semestrul	4	2.6. Tipul de evaluare	V	2.7. Regimul disciplinei	Continut ²	DD
							Obligativitate ³	DI

3. Timpul total estimat (ore pe semestru al activităților didactice)

3.1. Număr de ore pe săptămână	30	din care: 3.2. curs	0	3.3. seminar/ laborator/ proiect	30
3.4. Total ore din planul de învățământ	120	din care: 3.5. curs	0	3.6. seminar/laborator	90
Distribuția fondului de timp					ore
3.4.1. Studiul după manual, suport de curs, bibliografie și notițe					5
3.4.2. Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					4
3.4.3. Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					-
3.4.4. Tutoriala					-
3.4.5. Examinări					1
3.4.6. Alte activități					-
3.7. Total ore studiu individual	10				
3.8. Total ore pe semestru	100				
3.9. Numărul de credite ⁴	4				

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Studentul trebuie să posede cunoștințe de specialitate dobândite la disciplinele de programare prevăzute de planul de învățământ până la nivelul anului II. Dintre acestea pot fi menționate: Programarea calculatoarelor și limbaje de programare, Ingineria sistemelor de programe, Programare orientată pe obiecte, Informatică aplicată I și II, Baze de date, Grafică asistată de calculator, Metode numerice,.
4.2. de competențe	Studentul trebuie să aibă competențe referitoare la programare.

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Nu sunt prevăzute activități de predare
5.2. de desfășurare a seminarului/ laboratorului/ proiectului	Activitatea practică se desfășoară în domeniul studierii limbajelor de programare. Procesul de instruire are următoarea structură: <ul style="list-style-type: none"> • 80% instruire practică, studii de caz, studiul documentațiilor, teme de casă etc. • 20% activitate interactivă (discuții cu studenții) În cazul predării online se folosesc facilitățile aplicației Google Classroom.



ROMÂNIA
MINISTERUL EDUCAȚIEI ȘI CERCETĂRII
UNIVERSITATEA DIN CRAIOVA

FACULTATEA DE AUTOMATICĂ, CALCULATOARE ȘI ELECTRONICĂ
Blvd. Decebal nr.107, Craiova, RO-200440, Tel./Fax +(4)-0251-438.198, http://ace.ucv.ro



6. Competențe specifice acumulate

Competențe profesionale	<p>Prin cunoștințele dobândite, prin exemplele prezentate și prin aplicațiile practice efectuate, disciplina „Practică de domeniu” contribuie la formarea competențelor profesionale:</p> <p>C6: Aplicarea de cunoștințe de legislație, economie, marketing, afaceri și asigurare a calității, în contexte economice și manageriale.</p> <p>C6.1 Identificarea metodelor și tehnicilor de analiză și evaluare a produselor, a elementelor de design, precum și a principiilor de management, marketing și de inginerie a calității, aplicabile în activități ingineresti.</p> <p>C6.2 Interpretarea documentației specifice organizării procesului de execuție și implementare a proiectelor de sisteme automate și a aplicațiilor de informatică.</p> <p>C6.3 Elaborarea de documentație tehnică (proiecte) corect fundamentată din punct de vedere managerial și legislativ pentru probleme bine-definite din ingineria sistemelor.</p> <p>C6.4 Aprecierea măsurii și modului în care diferitele activități și documentații au fundamentare legislativă, economică, managerială și de asigurare a calității.</p> <p>C6.5 Organizarea și conducerea de activități specifice domeniului sistemelor automate și informaticii aplicate, incluzând execuția proiectelor, în condiții de respectare a cerințelor legale și manageriale.</p>
Competențe transversale	<p>CT2</p> <p>Identificarea rolurilor și responsabilităților într-o echipă plurispecializată, luarea deciziilor și atribuirea de sarcini, cu aplicarea de tehnici de relaționare și muncă eficientă în cadrul echipei.</p>

7. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

7.1. Obiectivul general al disciplinei	Contribuie la formarea viitorilor ingineri automatiști, specialiști în conducerea proceselor și informatică aplicată, asigurându-le competențe profesionale de operare cu concepte fundamentale din știința calculatoarelor, tehnologia informației și comunicațiilor.
7.2. Obiectivele specifice	<ul style="list-style-type: none"> ▪ Utilizarea argumentată a conceptelor din informatică și tehnologia calculatoarelor în rezolvarea de probleme bine definite din ingineria sistemelor și în aplicații ce impun utilizarea de hardware și software în sisteme industriale sau în sisteme informatice ▪ Rezolvarea de probleme uzuale din domeniul ingineriei sistemelor folosind concepte ale științei calculatoarelor și tehnologiei informației referitoare la utilizarea de software dedicat și de mijloace de proiectare asistată de calculator și la adaptarea și extinderea acestora. ▪ Folosirea proiectării hardware – software integrate (co-design) și a ingineriei programării ca metodologii de dezvoltare, inclusiv în vederea unei modelări la nivel de sistem.

8. Conținuturi

8.1 CURS (unități de conținut)	Nr. ore	Metode de predare
Bibliografie		
8.2 ACTIVITAȚI APLICATIVE (subiecte/teme)	Nr. ore	Metode de predare
<p>1. C/C++: aplicații embedded (pe plăci sau cu Proteus)</p> <p>1.1. Fundamentele sistemelor embedded</p> <p>1.2. Configurarea mediului</p> <p>1.3. Lucrul cu diferite arhitecturi</p> <p>1.4. Manipularea întreruperilor</p> <p>1.5. Depanare, jurnalizare și profilare</p> <p>1.6. Pointerii și gestionarea memoriei</p> <p>1.7. Intrări și ieșiri discrete</p> <p>1.8. Întreruperi</p> <p>1.9. Multitasking, multifer și sincronizare</p> <p>1.10. Comunicare și serializare</p> <p>1.11. Programarea microcontrollerelor</p>		<p>Desfășurarea de activități practice la nivelul postului de lucru.</p> <p>În cazul predării online se folosesc facilitățile aplicației Google Classroom.</p> <p>Structura activităților cuprinde:</p> <ul style="list-style-type: none"> • 80% îndeplinirea de sarcini conform



<p>2. Java: aplicații Android</p> <p>2.1. Platforma Android SDK</p> <p>2.2. Java SE SDK Android 10</p> <p>2.3. Java for Android Development, 4th Edition</p> <p>2.4. Colecții Java, concurență, I / O, rețele, persistență, programare funcțională și acces la date în aplicațiile Android</p> <p>2.5. Parsarea, crearea și transformarea documentelor XML și JSON</p> <p>2.6. Android Jetpack (Android Studio Integrated Development Environment (IDE), Android Architecture Components, Modern App Architecture Guidelines)</p> <p>2.7. Topici specifice Google Play: implementarea hărților folosind Google Maps Android API, trimiterea aplicațiilor către Google Play Developer Console</p> <p>2.8. Caracteristici cheie ale Android Studio 4.0 și Android 10: Layout Editor, clasele ConstraintLayout și ConstraintSet, animația MotionLayout, legările vizualizărilor, lanțurile de constrângeri, barierele și notificările de răspuns direct, App Links, Dynamic Delivery, Android Studio Profiler și Gradle build configuration.</p> <p>2.9. Interfața de utilizator Android: activități și machete, gestionarea evenimentelor, imagini, meniuri și bara de acțiuni</p> <p>2.10. Integrarea datelor cu persistența datelor</p> <p>2.11. Accesarea cloud-ului</p> <p>2.12. Jocuri Android: Gestionarea diferitelor dimensiuni și rezoluții ale ecranului; Clasa Canvas; Sistemul de coordonate Android; Metodele: onTouchEvent, takeShot, boom; Motorul jocului, fire și bucla jocului; Clasele Ball și Bat; Implementarea interfeței; UIController; Singleton Pattern, Java HashMap.</p>		portofoliului de practică
<p>3. Java: aplicații iOS</p> <p>3.1. JavaFxMobile Plugin Git Repo: plugin gradle pentru construirea de aplicații mobile în Java</p> <p>3.2. JavaFxPorts</p> <p>3.3. Gluon Plugin</p> <p>3.4. Multi-OS Engine</p>		• 20% interpretarea rezultatelor și discuții cu studenții
<p>4. Python</p> <p>4.1. Crearea de pagini Web cu Django (mediul virtual și structura proiectului pentru construirea de aplicații web Python: construcția modelelor, vizualizărilor, formelor și șabloanelor, integrarea JavaScript în aplicațiile Django; site-uri responsive multilingvistice; încărcarea și procesarea imaginilor, randarea datelor în HTML5, PDF și Excel; API; bune practici de securitate și tehnici de caching; Django cu Docker și Ansible)</p> <p>4.2. Crearea de pagini Web cu Masonite, cadrul Python centrat pe dezvoltator (personalizarea cererii și ciclului de răspuns cu middleware; grupuri de rute și antete; securitate; crearea și controlul caracteristicilor de întreținere ale aplicației de la un terminal; gestiunea bazei de date astfel încât colegii de echipă și mediile să fie întotdeauna sincronizate; citirea și scrierea din sistemul de fișiere; procesarea în fundal a seturilor mari de date și a operațiilor lente; caracteristici în timp real și sarcini recurente)</p> <p>4.3. Crearea de pagini Web cu Flask (configurare; modelare cu Jinja2; modelarea datelor; vederi; WTForms; autentificare; RESTful API Building; interfața administratorului pentru aplicații Flask; internaționalizare și localizare)</p> <p>4.4. Crearea de GUI (controale, managerii geometriei, gestionarea evenimentelor, tkinter, PyQt5)</p>		



ROMÂNIA
MINISTERUL EDUCAȚIEI ȘI CERCETĂRII
UNIVERSITATEA DIN CRAIOVA

FACULTATEA DE AUTOMATICĂ, CALCULATOARE ȘI ELECTRONICĂ
Blvd. Decebal nr.107, Craiova, RO-200440, Tel./Fax +(4)-0251-438.198, <http://ace.ucv.ro>



<p>4.5. Analiza și reprezentarea grafică a datelor (biblioteci Python pentru analiza datelor; NumPy, Pandas, Matplotlib, Seaborn; statistică, știința datelor; probabilitate; teorema Bayes; permutări, combinații, testarea ipotezelor: ANOVA, testul Chi-pătrat, testul z și testul t; biblioteca Scipy); descoperirea de tipare și tendințe noi; analize și modelări complexe de date; manipularea datelor; curățarea datelor; vizualizarea datelor; analiza seriilor temporale și procesarea semnalului utilizând modele ARMA; algoritmi de învățare automată: regresia, clasificarea, analiza componentelor principale (PCA), gruparea; analiza datelor text și a imaginilor; calcul paralel folosind Dask; prognozarea valorilor din date)</p> <p>4.6. Simularea dinamicii sistemelor: SimuPy, cadru pentru simularea modelelor de sistem dinamice interconectate</p> <p>4.7. Modele de simulare (conceptul de aleatoriu și numere aleatorii; probabilități și procese de generare a datelor; explorarea simulărilor Monte Carlo; procesul de decizie Markov bazat pe simulare; metode de eșantionare; utilizarea simulărilor pentru îmbunătățirea și optimizarea sistemelor; utilizarea modelelor de simulare pentru ingineria financiară; simularea fenomenelor fizice folosind rețelele neuronale; modelare și simulare pentru managementul proiectelor)</p> <p>4.8. Simularea computerizată: ciclul de viață al unei simulări, modelare și implementare; avantajele și limitele simulării; modele conceptuale; simularea stocastică; grafice de evenimente și proiectarea experimentelor; compararea rezultatelor obținute din simulare cu cele obținute folosind metode analitice; trecerea de la modele la programe; simularea cu/fără o listă de evenimente; single-queue single-server queuing system; metoda Monte Carlo și tehnici de reducere a varianței; graf de evenimente; simulare condusă de timp și condusă de evenimente.</p>		
TOTAL	90	
Bibliografie		
<ol style="list-style-type: none">1. Brian Kernighan, Dennis Ritchie, <i>The C Programming Language</i>, 2nd ed., Prentice Hall, 1988, ISBN 978-0131103627.2. Stroustrup, B., <i>The C++ Programming Language</i>, 4th ed., Addison-Wesley, 2013, ISBN 978-0321563842.3. Viarheichyk, I., <i>Embedded Programming with C++ Cookbook: Practical recipes to help you build robust and secure embedded applications using C++</i>, Packt Publishing, 2020, ISBN: 183882104X,9781838821043.4. Barr, M., <i>Embedded C Coding Standard</i>, Barr Group, 2018, ISBN: 1721127984,9781721127986.5. Siegesmund. M., <i>Embedded C programming: techniques and applications of c and pic mcus</i>, Newnes, Elsevier, 2015, ISBN: 978-0-12-801314-4,0128013141.6. Miasnikov, A., <i>C++ for embedded systems</i>, 2015.7. Thakur, M., <i>Arduino Projects Vol-I: With Proteus Simulation Files. Don't just read it, Try it...</i>, 2016.8. Bao, J., <i>Android App-Hook and Plug-In Technology (Java)</i>, CRC, 2020, ISBN: 9780367207007.9. Späth, P., Friesen, J., <i>Learn Java for Android Development - Migrating Java SE Programming Skills to Mobile Development</i>, Apress, 2020, ISBN: 9781484259429,9781484259436.10. Smyth, N., <i>Android Studio 4.0 Development Essentials - Java Edition: Developing Android Apps Using Android Studio 4.0, Java and Android Jetpack</i>, Payload Media, Inc., 2020, ISBN: 9781951442217.11. Hagos, T., <i>Learn Android Studio 4: Efficient Java-Based Android Apps Development</i>, Apress, 2020, ISBN: 148425936X,9781484259368.12. Horton, J., <i>Learning Java by Building Android Games</i>, Packt Publishing, 2018, ISBN: 1788839153, 978-1788839150.13. Neuburg, M., <i>Programming iOS 14</i>, O'Reilly Media, Inc., 2021, ISBN: 9781492092179.14. https://github.com/javafxports/javafxmobile-plugin15. https://docs.gluonhq.com/javafxports/#_how_it_works16. https://docs.gluonhq.com/17. https://multi-os-engine.org/18. https://github.com/multi-os-engine/multi-os-engine19. https://marketplace.eclipse.org/category/free-tagging/ios-app-development20. Bendoraitis, A., Kronika, J., <i>Django 3 Web Development Cookbook: Actionable solutions to common problems in Python web development</i>, Packt Publishing, 2020, ISBN: 1838987428,9781838987428.21. Pitt, C., Mancuso, J., <i>The Definitive Guide to Masonite: Building Web Applications with Python</i>, Apress, 2020, ISBN: 1484256018,9781484256015.		



ROMÂNIA
MINISTERUL EDUCAȚIEI ȘI CERCETĂRII
UNIVERSITATEA DIN CRAIOVA

FACULTATEA DE AUTOMATICĂ, CALCULATOARE ȘI ELECTRONICĂ
Blvd. Decebal nr.107, Craiova, RO-200440, Tel./Fax +(4)-0251-438.198, <http://ace.ucv.ro>



22. Ashley, D., *Foundation Dynamic Web Pages with Python: Create Dynamic Web Pages with Django and Flask*, 2020, ISBN: 9781484263389,9781484263396.
23. Aggarwal, S., *Flask Framework Cookbook: Over 80 proven recipes and techniques for Python web development with Flask*, Packt Publishing, 2019, ISBN: 1789951291, 978-1789951295.
24. Meier, B., *Python GUI Programming Cookbook: Develop functional and responsive user interfaces with tkinter and PyQt5*, Packt Publishing, 2019.
25. Rajagopalan, G., *A Python Data Analyst's Toolkit: Learn Python And Python-based Libraries With Applications In Data Analysis And Statistics*, Apress, 2021, ISBN: 1484263987, 9781484263983, 9781484263990.
26. Navlani, A., Fandango, A., Idris, I., *Python Data Analysis*, Packt, 2021, ISBN: 1789955246,9781789955248.
27. <https://github.com/simupy/simupy>
28. Margolis, B.W.L., *SimuPy Documentation*, 2020, <https://readthedocs.org/projects/simupy/downloads/pdf/latest/>
29. Ciaburro, G., *Hands-On Simulation Modeling with Python: Develop simulation models to get accurate results and enhance decision-making processes*, Packt Publishing, 2020, ISBN: 1838985093,9781838985097.
30. Osais, Y., E., *Computer Simulation: A Foundational Approach Using Python*, Chapman and Hall / CRC; Taylor & Francis Group, 2018, ISBN: 1498726828,978-1-498-72682-5.
31. Bădulescu, L., A., *Limbajul Python. Un curs practic*, Ed. Sitech, Craiova, 2020, ISBN 978-606-11-7447-8.
32. Bădulescu, L., A., *Proiectarea algoritmilor în limbajul Python*, Ed. Sitech, Craiova, 2020, ISBN: 978-606-11-7563-5.
33. Bădulescu, L., A., *Python. Aplicații și teste*, Ed. Sitech, Craiova, 2020, ISBN 978-606-11-7448-5.

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunităților epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

- S.C. C-S România S.A. Craiova
- S.C. NetRom Software S.R.L. Craiova
- HELLA Craiova

10. Evaluare

Tip activitate	10.1. Criterii de evaluare	10.2. Metode de evaluare	10.3. Pondere din nota finală
10.4. Curs			
10.5. Activități aplicative	<ul style="list-style-type: none"> • Capacitatea de a realiza conexiuni între conceptele cu care operează • Capacitatea de analiză și sinteză, de identificare a soluției optime și rezolvare a aplicațiilor propuse • Interpretarea rezultatelor • Aplicarea corectă a principiilor programării imperative • Soluțiile aplicațiilor se prezintă și se discută în cadrul grupei și cu tutorele 	Verificare pe parcurs și finală.	100%
10.6. Standard minim de performanță			
<ul style="list-style-type: none"> • Utilizarea conceptelor și instrumentelor din știința calculatoarelor și tehnologia informației și comunicațiilor pentru rezolvarea de probleme specifice ingineriei sistemelor. • Mijloc de validare: Teme de casă și proiecte cu sarcini individuale bine stabilite • Obținerea a minim 50% din punctajul verificărilor pe parcurs și finale. • Calculul notei finale se face prin rotunjirea la notă întregă a punctajului final. 			



Titular curs

Data completării
25.09.2020

Ș.I. dr. ing. Laviniu Aurelian
BĂDULESCU

Data avizării în
departament
30.09.2020

Director de departament
Prof. Dr. ing. Cosmin IONETE