

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Facultatea de Autovehicule Rutiere, Mecatronică și Mecanică
1.3 Departamentul	Mecatronica și Dinamica Masinilor
1.4 Domeniul de studii	Mecatronica și Robotica
1.5 Ciclul de studii	Licență
1.6 Programul de studii / Calificarea	Mecatronica (lic)
1.7 Forma de învățământ	IF – învățământ cu frecvență
1.8 Codul disciplinei	6.00

2. Date despre disciplină

2.1 Denumirea disciplinei	Programarea calculatoarelor și limbaje de programare I						
2.2 Aria de conținut	(se completează din grila 2: arii de conținut)						
2.3 Responsabil de curs	Prof. Univ. Dr. Ing. Iulian Lupea iulian.lupea@mep.utcluj.ro						
2.4 Titularul activităților de seminar / laborator / proiect	Prof. Univ. Dr. Ing. Iulian Lupea iulian.lupea@mep.utcluj.ro						
2.5 Anul de studiu	1	2.6 Semestrul	1	2.7 Tipul de evaluare	C	2.8 Regimul disciplinei	DF DI

3. Timpul total estimat

3.1 Număr de ore pe săptămână	3	din care: 3.2 curs	1	3.3 seminar / laborator	2
3.4 Total ore din planul de învățământ	42	din care: 3.5 curs	14	3.6 seminar / laborator	28
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					10
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					4
Pregătire seminarii / laboratoare, teme, referate, portofolii și eseuri					12
Tutoriat					3
Examinări					4
Alte activități.....					
3.7 Total ore studiu individual	33				
3.8 Total ore pe semestru	75				
3.9 Numărul de credite	3				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Discipline cu profil Informatic din liceu
4.2 de competențe	Matematică (liceu și semestrul I facultate)

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Prezența la curs este obligatorie 80%
5.2. de desfășurare a seminarului / laboratorului / proiectului	Prezența la laborator este obligatorie 100%. Se programează la terminal teme impuse și teme individuale colectate în dosar.

6. Competențele specifice acumulate

Competențe profesionale	<p>După parcurgerea disciplinei studenții vor fi capabili:</p> <ul style="list-style-type: none"> • Să fie familiarizați cu conceptele de programare grafică. • Să fie edificați cu conceptul de instrumentație virtuală, • Să cunoască și adapteze instrucțiunile de decizie, selecție, ciclare, rolul și locul lor în programare în general și programarea grafică în special, • Să poată modulariza programele prin conceperea și apelul de funcții, • Să reprezinte grafic funcții 2D și 3D și alte reprezentări grafice inginerești, • Să înțeleagă mediul de dezvoltare al aplicațiilor pentru achiziție de date de la senzori și control, • Să fie inițiați în prelucrare numerică a datelor provenite de la senzori cu accente pe statistică, prelucrarea în domeniul frecvență a datelor, procesarea șirurilor și matricelor.
Competențe transversale	<p>Formare continuă și utilizarea eficientă a surselor informaționale și a resurselor de comunicare (portaluri Internet, aplicații software de specialitate, baze de date, cursuri on-line etc.).</p> <ul style="list-style-type: none"> • preluarea și explicarea unei aplicații mai complexe din baza de exemple disponibile în mediul Labview (face parte din dosarul de laborator) • combinarea cunoștințelor de programare cu cele ale disciplinelor matematice, • să poată aplica cunoștințele de la disciplina <i>Programarea calculatoarelor și limbaje de programare I în cadrul disciplinelor viitoare în sensul rezolvării numerice a calculelor necesare la acele discipline</i>

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

7.1 Obiectivul general al disciplinei	Însușirea principiilor limbajelor de programare a calculatoarelor în vederea aplicării în cadrul disciplinelor de profil și în inginerie
7.2 Obiectivele specifice	<p>După parcurgerea disciplinei studenții vor fi capabili:</p> <ul style="list-style-type: none"> • Să facă calcule numerice inginerești cu ușurință, folosind produse program; • Să realizeze aplicații cu instrumente virtuale; • Să poată scrie relativ rapid aplicații pentru implementarea unor algoritmi specifici disciplinelor de profil; <p>Să poată aplica programarea la probleme de măsurare a mărimilor mecanice, control, procesare numerică a datelor și interacțiunea cu mediul în luarea deciziilor prin intermediul senzorilor.</p>

8. Conținuturi

8.1 Curs	Metode de predare	Observații
1. Generalități despre structura calculatorului, sist. de operare, clasificare limbaje. Introducere în programarea grafică; domenii de aplicare, specificitate. Instrumentul virtual; panoul frontal, diagrama. Generare șiruri numerice (pattern, wave) cu scopul de a fi prelucrate și trasate grafic. Funcții predefinite vizualizare grafică 2D. Instrucțiuni de ciclare I.	În procesul de predare se folosesc metode noi ce utilizează aparatură media combinate cu metode clasice	Noțiunile teoretice vor fi însoțite de scurte prezentări practice, simulări și /sau exemplificări din experiența inginerească 2 ore fiecare curs Total 7 cursuri
2. Tipuri de date (simple și structurate) inclusiv specifice măsurătorilor inginerești (formă de undă), operatori (implementare Labview), expresii numerice, logice și relaționale. Structuri pentru controlul execuției: Select, Case. Instrucțiuni de ciclare II: exemple pentru prelucrarea matricelor numerice. Aplicații șiruri: generare de șiruri aleatoare și calcule statistice. Calcul histograma.		
3. Includerea de cod limbajul C în Labview (formula node). Prezentarea sintaxei și a instrucțiunilor limbajului C. Prelucrarea șirurilor II (ordonări), prelucrare matrice II. Funcții predefinite pentru sisteme de ecuații liniare și neliniare. Prelucrare numere complexe I. Programare transformata Fourier (Real FFT). Funcții grafice I (vizualizare coeficienți spectrali). Procesare șiruri III: integrări și derivari numerice.		

4. Prelucrare șiruri și matrice de numere complexe II. Aplicații: putere spectrală, interspectrală, funcție de răspuns în frecvență. Reprezentări grafice complexe.		
5. Tipul de dată structură, tabloul de structuri. Forma de undă (waveform). Variabila locală versus registrul de transfer. Salvare date numerice în fișier. Aplicații cu șiruri: programare ferestre de ponderare și vizualizare grafică. Dezvoltarea de instrumente virtuale simple și modificarea unora mai complexe. Programare pe evenimente – Event-Driven programming - cu Labview: evenimente mouse, key, modificare ferestre, panouri, modificare valori controale, filtrare evenimente, evenimente dinamice.		
6. Aplicații cu placi de sunet programate în Labview. Funcții pentru achiziție sunet prin microfon. Canale achiziții. Aplicație pentru vizualizare spectru de putere. Funcții generare sunet la difuzor. Baleere cu semnal sinus. Achiziție și generare sunet simultan. Simulare taste telefon.		
7. Programare placi Arduino, Interfața Labview-Arduino, citire senzori (accelerație 3axe, magnetic, umiditate, lumina, proximitate), comanda motor cc.; Prezentare aplicații de achiziție și procesări complexe folosind senzori.		
<p>Bibliografie</p> <ol style="list-style-type: none"> 1. www.viaclab.utcluj.ro, Lupea, I., Cursul propriu și laboratoare actualizate anual, oferite în format electronic. <i>In biblioteca UTC-N + internet format .pdf</i>, 2021 2. Anghel, T., Programarea plăcii Arduino, 2020 3. Bishop, R., Learning with Labview 6i, Prentice Hall, 2001 4. Blokdyk, G., LabVIEW A Complete Guide - 2021 Edition Kindle Edition, 2021 5. Cottet, F., Ciobanu, O., Bazele programării în Labview, Editura Matrix Rom, Buc., 1998 6. Fadhil, A., Lecture Notes in LabVIEW and Data Acquisition, Kindle Edition, 2021 7. Hedeșiu H., Munteanu R. Jr., Intr. în programare grafică instrumentală, Ed. Mediamira, Cluj-N., 2003 8. Isen, F., DSP for MATLAB and LabVIEW II: Discrete Frequency Transforms (Synthesis Lectures on Signal Processing), 2008. 9. Johnson, G., Labview Power Programming, McGraw-Hill, NY, 1998. 10. Lupea I., Roboți și vibrații, Editura Dacia, Cluj-Napoca, 1996. 11. Lupea I., Lupea, M., Limbajul C, teorie și aplicații, Editura Casa Cărții de Știință,, Cluj-Napoca, 1998. 12. Lupea I., Măsurători de vibrații și zgomote prin programare cu Labview, Casa Cărții de Știință, Cluj-Napoca, 2005. 13. Lupea I., <u>Labview - Programare Grafică</u>, Editura Risoprint, Cluj-Napoca, 2008. 14. Morris, A., Langari, R., Measurement and instrumentation : theory and application, 2012 15. Larsen, R., <u>LabView for engineers</u>, 2011 16. Paton, B., Sensors, transducers, and Labview, Prentice Hall, 1999 17. Ricardo A., LabVIEW: A Flexible Environment for Modeling and Daily Laboratory Use, 2021 18. Travis, J., Labview for everyone: graphical programming made easy and fun, Prentice Hall, 2007 19. Schwartz, M., Manickum, O., Programing Arduino with Labview, Packt Publishing, 2015. 20. www.ni.com 21. ** Getting Started with LabVIEW, www.ni.com, 2018 22. ** Labview User manual, www.ni.com, 2020 23. ** Labview Analysis concepts, www.ni.com, 2017 24. ** LabVIEW Fundamentals, www.ni.com, 2018 25. ** www.arduino.cc 		
8.2 Seminar / laborator / proiect	Metode de predare	Observații
1. Mediul de programare Labview; controale și indicatoare numerice, indicatoare grafice. Aplicații cu ciclul While: oprire ciclu pe diverse condiții, reprezentare grafică a unei funcții	La laborator studentul vine cu conspectul facut / materialul descărcat de pe site:	Activitatea de laborator se desfășoară pe semigrupe cu durata 2 ore
2. Aplicații cu ciclul For (sumă și medii elemente șir numeric, suma și medii elemente din matrice). Verific. relații matriceale (direct).		

3. Op. Select: sumă elemente pozitive din șir. Cicluri For imbricate (zone din matrice). Regiștrii shift multipli (Sir Fibonacci). Probleme propuse.	www.viaclab.utcluj.ro La unele laboratoare se lansează teme de casă care se adună în dosarul final.	săptămânal media 2 ore un laborator, Total 14 laboratoare ** întârzierile la laborator se recuperează
4. Inst. Case - selecție multiplă; selector Boolean, numeric, enum. Aplicații rezolvate și propuse: suma poz. cu Case; element maxim din matrice și poziția lui etc.		
5. Reprezentare grafică f. armonică, media, RMS, deviația standard. Semnale aleatoare (Uwn, Gwn, Prn), histograma. Aplic. rezolvate și propuse.		
5' Aplicații statistice pentru studenți avansați.		
6. Formula Node (Labview), utilizare cod limbajul C (I). Date tip structură.		
7. Calcule cu numere complexe I (real FFT, power spectrum etc.)		
8. Calcule cu numere complexe II (matrice numerice, FRF, medieri).		
9. Cod C în Labview (II). Aplicații cu ferestre de ponderare. Aplic. rezolvate și propuse.		
10. Programare pe evenimente: aplicații diverse.		
11. Forme de unde, tablouri de forme de unde și procesări de semnale achiziționate.		
12. Placa de sunet; funcții de achiziție sunet; canale de achiziție. Funcții pentru generare de sunete. Prelucrarea sunetelor.		
13. Programare placă Arduino, diverși senzori (temp, magnetic, umiditate, lumina, accelerometru 3axial; comanda motor cc. sens și turație.		
14. Test practic la terminal.		
Bibliografie www.viaclab.utcluj.ro , Lupea, I., Site Laborator. Laboratoare actualizate anual, oferite în format electronic de pe site www.ni.com https://www.ni.com/ro-ro/innovations/academic-research.html + bibliografia de la curs.		

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

Se realizează prin discuții periodice programate de facultate cu reprezentanți ai angajatorilor * observarea nevoilor în domeniile deservite, contractele cu industria, dialog cu responsabilii domeniilor de studiu

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Examen scris constând din subiecte de teorie + întrebări de sinteză + interpretare/scriere segmente scurte de program.	Verificarea cunoștințelor (teorie și aplicații) în scris pe durata a 2 ore; minim 5 pentru promovare	50%
10.5 Seminar/Laborator	Test practic la terminal/ calculator 3 probleme + dosar (3 probleme, una fiind la alegere) + teste de scurta durata 10 minute la laboratoare.	Se apreciază cu notă cuprinsă între 1 și 10; minim 5 pentru promovare și condiționează intrarea la scris	50%
10.6 Standard minim de performanță			
<ul style="list-style-type: none"> Abordarea la nivel mediu a unor programe de instrumentație virtuală. Rezolvarea satisfăcătoare a unor programe de complexitate medie incluzând instrucțiuni condiționale și instrucțiuni de ciclare atât pentru promovarea componentei practice de la laborator cât și la scris. Explicarea unor aplicații de complexitate medie funcționale. 			

Data completării:	Titulari	Titlu Prenume NUME	Semnătura
23.05.2024	Curs	Prof.dr.ing. Iulian LUPEA	
	Aplicații	Prof.dr.ing. Iulian LUPEA	

Data avizării în Consiliul Departamentului MDM,
31.05.2024

Director Departament MDM,
Prof. dr. ing. Mircea Bara

Data aprobării în Consiliul Facultății ARMM,

Decan ARMM,
Prof. dr. ing. Nicolae Filip