

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Autovehicule Rutiere, Mecatronica și Mecanica
1.3 Departamentul	Mecatronica și Dinamica Masinilor + Inginerie Mecanica
1.4 Domeniul de studii	Mecatronica Robotica, Inginerie Mecanică
1.5 Ciclul de studii	Licență
1.6 Programul de studii / Calificarea	MTR, IM.
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 seminar 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 înțeleagă conceptele de programare în general și programare grafică în particular. • 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 in mediul Labview (face parte din dosarul de laborator) • combinarea cunoștințelor de programare cu cele ale disciplinelor matematice, • să poată aplica cunostintele 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 predef. vizualizare grafică 2D. Instrucțiuni de ciclare I. Variabila locală versus registrul de transfer.	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 limb. 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. Programare pe evenimente.		
5. Tipul de dată structură, tabloul de structuri. Forma de undă (waveform). Salvare în fișier. Aplicații cu șiruri: programare ferestre de ponderare și vizualizare grafică. Dezvoltarea de instrumente virtuale simple și modificarea unora mai complexe. Programarea unor plăci Arduino și o plajă de senzori + motor cc. prin Labview.		
6. Aplicații cu plăci de sunet. Funcții pentru achiziție sunet microfon. Canale achiziției. Aplicație vizualizare spectru. Funcții generare sunet la difuzor. Baleere sinus. Achiziție și generare sunet simultan. Simulare taste telefon.		
7. Prezentare aplicații de achiziție și procesări complexe folosind senzori.		
<p>Bibliografie</p> <p>1. www.viaclab.utcluj.ro, Lupea, I., Cursul propriu și laboratoare actualizate anual (2005-2020), oferite în format electronic.</p> <p>In biblioteca UTC-N + internet format .pdf</p> <p>2. Bishop, R., Learning with Labview 6i, Prentice Hall, 2001</p> <p>3. Cottet, F., Ciobanu, O., Bazele programării în Labview, Editura Matrix Rom, Buc., 1998</p> <p>4. Hedeșiu H., Munteanu R. Jr., Intr. în programare grafică instrumentală, Editura Mediamira, Cluj-Napoca, 2003</p> <p>5. Johnson, G., Labview Power Programming, McGraw-Hill, NY, 1998.</p> <p>6. Lupea I., Roboți și vibrații, Editura Dacia, Cluj-Napoca, 1996.</p> <p>7. Lupea I., Lupea, M., Limbajul C, teorie și aplicații, Casa Cărții de Știință, Cluj-Napoca, 1998.</p> <p>8. Lupea I., Măsurători de vibrații și zgomote prin programare cu Labview, Casa Cărții de Știință, Cluj-Napoca, 2005.</p> <p>9. Lupea I., <u>Labview - Programare Grafică</u>, Editura Risoprint, Cluj-Napoca, 2008.</p> <p>10. Morris, A., Langari, R., Measurement and instrumentation : theory and application, 2012</p> <p>11. Larsen, R., <u>LabView for engineers</u>, 2011</p> <p>12. Paton, B., Sensors, transducers, and Labview, Prentice Hall, 1999</p> <p>13. Travis, J., Labview for everyone: graphical programming made easy and fun, Prentice Hall, 2007</p> <p>14. www.ni.com</p>		
8.2 Seminar / laborator / proiect	Metode de predare	Observații
L1. Mediul de programare Lv. Aplicație demonstrativă. Ciclul While. Reprezentarea grafică a unei funcții. Modificarea aplicațiilor rezolvate. Programe model și probleme propuse.		
L2. Ciclul For (sumă elemente șir numeric și matrice). Verific. relații matriceale (direct). Registrul de transfer. Op. Select: sumă elemente pozitive din șir. Cicluri For imbricate (zone din matrice). Regiștrii shift multipli (Șir Fibonacci). Probleme propuse.		
L3. I. Case- selecție multiplă; selector Boolean, numeric, enum, rescriere programe cu Case în locul Select. Variabila locală. Rescriere aplicații cu variabila locală în locul reg. Shift. Element maxim din matrice și poziția lui. Aplicații rezolvate și probleme propuse.		
L4. Date de tip Structura. Bundle, unbundle +by name; Aplicație cu prelucrare tablouri de structuri (studenți, discipline, note). Structura de control Sequence (plat, stivuit). Aplicații diverse. Probleme propuse.		
L5. Organizare pe pagini în Panoul Frontal cu Tab Control. Reprezentare grafică f. armonică, media, RMS, deviația standard. Semnale aleatoare (uwn, gwn, prn), histograma. Aplic. rezolvate și probleme propuse.		
L5' Aplicații statistice pentru studenți avansați.		
L6. Formula Node (Labview), instrucțiuni și utilizare cod limbajul C (I). Apel C/C++ DLL din Labview.		
	La laborator studentul vine cu conspectul făcut / materialul descărcat de pe site: www.viaclab.utcluj.ro La unele laboratoare se lansează teme de casă care se adună în dosarul final.	Activitatea de laborator se desfășoară pe semigrupe cu durata 2 ore săptămânal media 2 ore un laborator, Total 14 laboratoare ** întâzierile la laborator se recuperează ** recuperările urmăresc

L7. Calcule cu numere complexe I (programare calcul real FFT, power spectrum etc.)		
L8. Calcule cu numere complexe II (matrice numerice, FRF, IF, COH).		
L9. Cod C în Labview (II). Aplic. ferestre de ponderare. Aplic. rezolvate și probleme propuse.		
L10. Programare pe evenimente (Event structure). Ciclul While (polling) versus Event Structure. Eveniment static/dinamic. Aplicații diverse și probleme propuse.		
L11. Programare Arduino (Uno, Mega2560) prin Labview; senzori LED, buzzer, lumina, umiditate, piezo-acceleratie, motor cc (comanda turatie, sens). Aplicații diverse.		
L12. Placa de sunet; funcții de achiziție sunet; canale de achiziție. Tablouri de forme de unde și procesări de semnale achiziționate. Funcții pentru generare de sunete. Prelucrarea sunetelor.		
L13. Procesare sunete achiziționate și generare sunete (II). Utilizare instrumente virtuale: analizor spectral, sonometru etc.		
L14. Test practic la terminal.		
<p>Bibliografie</p> <p>www.viacolab.utcluj.ro, Lupea, I., Site Laborator. Laboratoare actualizate anual, oferite în format electronic de pe site și depuse în MS Teams</p> <p>Schwartz, M., Programming Arduino with Labview, 2015</p> <p>www.arduino.cc</p> <p>https://duino4projects.com/interfacing-labview-with-arduino/</p> <p>https://microcontrollerslab.com/structures-labview-programming/</p> <p>https://microcontrollerslab.com/list-labview-tutorials-projects/</p> <p>https://www.eqr.msu.edu/~hashsham/courses/ene806/docs/LabView%20Introduction.pdf</p> <p>https://www.vyssotski.ch/BasicsOfInstrumentation/LabVIEW%20Fundamentals.pdf</p> <p>http://www.ni.com/pdf/manuals/373427j.pdf</p> <p>www.ni.com</p> <p>+ bibliografia de la curs.</p>		

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

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

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs/colocviu	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	Notare student pe parcurs. Test practic la terminal/calculator + dosar (3 probleme, una fiind la alegere) + teste de scurta durata 10 minute pe parcurs la laboratoare.	Se apreciază cu notă cuprinsă între 1 și 10; minim 5 pentru promovare și condiționează intrarea la scris/colocviu.	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 functionale. 			

Data completării:	Titulari	Titlu Prenume NUME	Semnătura
10.10.2020	Curs	Prof.dr.ing. Iulian LUPEA	
	Aplicații	Prof.dr.ing. Iulian LUPEA	

Data avizării în Consiliul Departamentului	Director Departament

Data aprobării în Consiliul Facultății ARMM	Decan Prof.dr.ing. Nicolae FILIP
