

FIȘA DISCIPLINEI¹

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Politehnică Timișoara
1.2 Facultatea ² / Departamentul ³	Facultatea de Inginerie Hunedoara / Inginerie Electrică și Informatică Industrială
1.3 Catedra	—
1.4 Domeniul de studii (denumire/cod ⁴)	ȘTIINȚE INGINEREȘTI APLICATE / 270
1.5 Ciclul de studii	Licenta
1.6 Programul de studii (denumire/cod/calificarea)	INFORMATICĂ INDUSTRIALĂ / 50 / Inginer

2. Date despre disciplină

2.1 Denumirea disciplinei/Categoria formativă ⁵	Programare aplicațiilor de timp real / DD						
2.2 Titularul activităților de curs	Ș. L. dr. ing. Abrudean Cristian						
2.3 Titularul activităților aplicative ⁶	Ș. L. dr. ing. Abrudean Cristian						
2.4 Anul de studii ⁷	IV	2.5 Semestrul	II	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei ⁸	DO

3. Timp total estimat - ore pe semestru: activități didactice directe (asistate integral sau asistate parțial) și activități de pregătire individuală (neasistate)⁹

3.1 Număr de ore asistate integral/săptămână	3,5 , format din:	3.2 ore curs	2	3.3 ore seminar/laborator/proiect	1,5
3.1* Număr total de ore asistate integral/sem.	49 , format din:	3.2* ore curs	28	3.3* ore seminar/laborator/proiect	21
3.4 Număr de ore asistate parțial/săptămână	, format din:	3.5 ore practică		3.6 ore elaborare proiect de diplomă	
3.4* Număr total de ore asistate parțial/semestru	, format din:	3.5* ore practică		3.6* ore elaborare proiect de diplomă	
3.7 Număr de ore activități neasistate/săptămână	3,92 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			0,92
		ore studiu individual după manual, suport de curs, bibliografie și notițe			1,5
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			1,5
3.7* Număr total de ore activități neasistate/semestru	55 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			13
		ore studiu individual după manual, suport de curs, bibliografie și notițe			21
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			21
3.8 Total ore/săptămână ¹⁰	7,42				
3.8* Total ore/semestru	104				
3.9 Număr de credite	4				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	<ul style="list-style-type: none"> Programarea calculatoarelor, Sisteme de operare
4.2 de competențe	<ul style="list-style-type: none">

¹ Formularul corespunde Fișei Disciplinei promovată prin OMECTS 5703/18.12.2011 și cerințelor Standardelor specifice ARACIS valabile începând cu 01.10.2017.

² Se înscrie numele facultății care gestionează programul de studiu căruia îi aparține disciplina.

³ Se înscrie numele departamentului căruia i-a fost încredințată susținerea disciplinei și de care aparține titularul cursului.

⁴ Se înscrie codul prevăzut în HG nr.140/16.03.2017 sau în HG similare actualizate anual.

⁵ Disciplina se încadrează potrivit planului de învățământ în una dintre următoarele categorii formative: disciplină fundamentală (DF), disciplină de domeniu (DD), disciplină de specialitate (DS) sau disciplina complementară (DC).

⁶ Prin activități aplicative se înțeleg activitățile de: seminar (S) / laborator (L) / proiect (P) / practică (Pr).

⁷ Anul de studii în care este prevăzută disciplina în planul de învățământ.

⁸ Disciplina poate avea unul din următoarele regimuri: disciplină impusă (DI), disciplină opțională (DO) sau disciplină facultativă (Df).

⁹ Numărul de ore de la rubricile 3.1*, 3.2*,...,3.8* se obțin prin înmulțirea cu 14 (săptămâni) a numărului de ore din rubricile 3.1, 3.2,..., 3.8. Informațiile din rubricile 3.1, 3.4 și 3.7 sunt chei de verificare folosite de ARACIS sub forma: (3.1)+(3.4) ≥ 28 ore/săpt. și (3.8) ≤ 40 ore/săpt.

¹⁰ Numărul total de ore / săptămână se obține prin însumarea numărului de ore de la punctele 3.1, 3.4 și 3.7.

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	<ul style="list-style-type: none"> • Sală de curs echipată cu videoproiector și conexiune la Internet. • Studenții nu se vor prezenta la prelegeri cu telefoanele mobile deschise. • Nu se acceptă părăsirea sălii de curs fără aprobarea cadrului didactic.
5.2 de desfășurare a activităților practice	<ul style="list-style-type: none"> • Sală de laborator echipată cu computere • Studenții nu se vor prezenta la activitățile practice cu telefoanele mobile deschise. • Nu se acceptă părăsirea sălii de desfășurare a activității practice fără aprobarea cadrului didactic.

6. Competențe la formarea cărora contribuie disciplina

Competențe specifice	<p>C 2.</p> <p>C 2.1. Descrierea structurii și a modului de funcționare a sistemelor informatice în general; C 2.2. Explicarea rolului, funcționalității și utilității sistemelor informatice în general și a sistemelor de prelucrare și gestiune a datelor în domeniul specializării; C 2.3. Utilizarea componentelor software ale sistemelor informatice, folosind algoritmi, protocoale, limbaje, structuri de date; C 2.4. Aprecierea caracteristicilor și calității sistemelor informatice; C 2.5. Prelucrarea și gestionarea datelor utilizând sisteme informatice dedicate.</p> <p>C 4.</p> <p>C 4.1. Descrierea arhitecturilor de bază pentru sistemele informatice aplicate în conducerea sistemelor energetice sau industriale. C 4.2. Explicarea și interpretarea funcționării elementelor sistemelor informatice aferente conducerii proceselor energetice sau industriale; C 4.3. Alegerea elementelor unui sistem informatic destinat conducerii, comenzii, reglajului sau supravegherii unui proces energetic sau industrial; C 4.4. Utilizarea criteriilor și metodelor de evaluare a performanțelor tehnice și informatice ale unui sistem informatic de proces; C 4.5. Implementarea unei structuri de sistem informatic de conducere a proceselor din sistemele energetice sau industriale.</p> <p>C 5.</p> <p>C 5.1. Descrierea structurilor de conducere automată bazate pe microprocesoare și microcontrolere; C 5.2. Explicarea utilizării microprocesoarelor și microcontrolerelor și cunoașterea softului aferent acestora; C 5.3. Modelarea, simularea și testarea sistemelor de conducere automată a proceselor industriale; C 5.4. Evaluarea performanțelor de regim staționar și dinamic ale sistemelor de conducere automată; C 5.5. Realizarea unui sistem de comandă și reglare automată a unui proces industrial specific domeniului specializării.</p> <p>C 6.</p> <p>C 6.1. Descrierea principiilor de bază privind achiziția și transmisia de date din proces; C 6.2. Explicarea rolului componentelor sistemelor de achiziție de date aferente unui sistem informatic destinat conducerii automate a proceselor industriale; C 6.3. Configurarea sistemelor de achiziție și transmisie de date aferente proceselor industriale; C 6.4. Utilizarea adecvată a metodelor de evaluare a performanțelor sistemelor informatice și de validare a datelor achiziționate din proces; C 6.5. Implementarea componentelor sistemelor informatice de achiziție de date.</p>
Competențele profesionale în care se înscriu competențele specifice	<ul style="list-style-type: none"> • C 2. Utilizarea sistemelor informatice de prelucrare și gestiune a datelor. • C 4. Realizarea și implementarea sistemelor informatice de conducere, comandă, reglaj și supraveghere a proceselor energetice sau industriale • C 5. Analiza și sinteza sistemelor de conducere a proceselor industriale bazate pe microprocesoare și microcontrolere. • C 6. Configurarea, implementarea și folosirea sistemelor de achiziție de date.

Competențele transversale în care se înscriu competențele specifice	•
---	---

7. Obiectivele disciplinei (asociate competențelor de la punctul 6)

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> Obiectivele cursului constau în însușirea de către studenți a cunoștințelor teoretice și aplicative a disciplinei de programare în timp real, cunoștințe necesare pentru proiectarea și implementarea de aplicații în timp real în sistemele de operare timp real. Însușirea acestei discipline are ca rezultat o pregătire de specialitate a studenților punându-le la dispoziție cunoștințe din domeniul programării nesecventiale, programării threadurilor și proceselor concurente în sistemele de operare de timp real, în special sistemul de operare QNX, cu ajutorul cărora să se poată alinia la progresul științei, să-și dezvolte abilități de programare cu constrângeri de timp real sub diverse familii de sisteme de operare real time (QNX, Real Time Linux); să devină competenți pentru programarea de aplicații real time.
7.2 Obiectivele specifice	<ul style="list-style-type: none"> Scopul formativ al cursului este ca studentul să își formeze o viziune de ansamblu asupra sistemelor integrate de timp-real. Se abordează în acest curs problematica sistemelor integrate de timp-real, din perspectiva dezvoltării de aplicații software de timp-real. Sunt exemplificate practic modalitățile de gestiune a timpului și de concurență în dezvoltarea software-ului de timp-real. Dezvoltare de aplicații în C/POSIX (QNX), Java și Real-Time Java (Windows / QNX / Linux). La finele cursului, studenții trebuie să aibă cunoștințe teoretice și abilități de cercetare, strict necesare viitorilor specialiști, dovedind competențe în selectarea, utilizarea corectă și combinarea adecvată a tehnologiilor de programare real/time în sistemele de operare de timp real.

8. Conținuturi¹¹

8.1 Curs	Număr de ore	Metode de predare ¹²
1. Noțiuni introductive despre sistemele integrate timp real 1.1. Sistem integrat de timp-real (embedded real-time system). Sisteme de operare de timp real. Arhitecturi software pentru proiectarea aplicațiilor de timp-real (executiv de tip ciclic, sisteme event-based, etc). 1.2. Limbaje de programare pentru aplicații de timp-real (secventiale, concurențiale). Programare concurență. Definiții (proces, fir de execuție, task, proces, preemțiunea taskurilor). Probleme de bază ale proiectării aplicațiilor de timp-real (sincronizare, excludere mutuală, deadlock, starvation).	4	Expunerea informațiilor esențiale pe videoprojector. Detalierea informațiilor expuse. Conversația
2. Aspecte concurențiale ale aplicațiilor de timp-real 2.1. Sincronizarea taskurilor. Sistem de activități. Mecanisme pentru realizarea sincronizării. Semafoare binare. 2.2. Semafoare generalizate. Excludere mutuală. Transmitere de mesaje. Cutii postale. 2.3. Sincronizarea cu ajutorul cutiilor postale. Excluderea mutuală cu ajutorul cutiilor postale. 2.4. Buffere dinamice (pipes). Monitoare. Sincronizarea taskurilor cu ajutorul monitoroarelor.	8	
3. Aspecte temporale ale aplicațiilor de timp-real 3.1. Noțiuni referitoare la "timp". Aspecte de timp-real (precizia măsurătorilor, consistența, curba funcției de utilitate). Utilizarea noțiunii de "timp" într-un sistem de timp-real. Scopuri temporale. 3.2. Caracteristici de limbaj de programare - timeout-uri, timere, alarme, semnale. Deadline-uri. Specificarea și analiza deadline-urilor. Priorități. Algoritmi de evitare a inversiunii de prioritate. 3.3. Algoritmi de planificare (rate monotonic scheduling, deadline	6	

¹¹ Se detaliază toate activitățile didactice prevăzute prin planul de învățământ (tematicile prelegerilor și ale seminariilor, lista lucrărilor de laborator, conținuturile etapelor de elaborare a proiectelor, tematica fiecărui stagiu de practică). Titlurile lucrărilor de laborator care se efectuează pe standuri vor fi însoțite de notația „(*)”.

¹² Prezentarea metodelor de predare va include și folosirea noilor tehnologii (e-mail, pagină personalizată de web, resurse în format electronic etc.).

monotonic scheduling, arbitrary-fixed priority scheduling, earliest-deadline-first scheduling).		
4. Tehnici de specificare si proiectare a aplicatiilor de timp-reall 4.1. Organigrame, diagrame structurate, masini de stare finita, retele Petri, etc. 4.2. Metode de proiectare structurata (HRT-HOOD, UML pentru real-time). Ciclul de viata al software-ului. Standarde.	4	
5. Caracteristicile sistemelor de timp-real 5.1. Determinism, siguranta in functionare, etc. Planificarea taskurilor in sisteme de timp-real. Arhitecturi software pentru proiectarea aplicatiilor de timp-real. 5.2. Limbaje de programare pentru aplicatii de timp-real. Deadlock, starvation, cozi ciclice - algoritmi de detectie, evitare si prevenire. Interactiunea cu dispozitivele de I/O - mecanisme hardware I/O. Caracteristici de limbaj pentru programare low-level.	4	
6. Modele de stare si programe Java 6.1. Modelarea proceselor cu ajutorul masinilor de stare finita. 6.2. Procese si fire de executie.	2	
Bibliografie ¹³ 1. Dragoicea, M., Programarea aplicatiilor in timp real. Editura Universitara, Bucuresti, 2009. 2. BOIAN F.M., FERDEAN C. M., BOIAN R.F. DRAGOS R.C. Programare concurenta pe platforme Unix, Windows, Editura Albastra - grupul Microinformatica, Cluj, 2002. 3. Letia, T., Sisteme de timp-real, Ed. Albastra, 2000 4. Wellings, A., Concurrent and Real-Time Programming in Java, John Wiley, 2004 5. Robu Nicolae ,Programare concurentă : Mecanisme suport orientate timp real, Ed. Politehnica, 2002.		
8.2 Activități aplicative¹⁴	Număr de ore	Metode de predare
1. Prezentarea sistemului de operare de timp-real QNX. Dezvoltarea aplicațiilor în QNX.	4	Verificare cunostinte din tematica laboratorului
2. Standardul POSIX pentru timp real. Mecanisme de sincronizare si comunicare sub QNX/POSIX. Semafoare. Sincronizarea firelor de executie. Sincronizarea proceselor..	2	Elaborare de aplicatii interactive si testarea lor
3. Mecanisme de sincronizare si comunicare sub QNX / POSIX. Mutex-uri. Zone partajate de memorie. Variabile conditionale.	2	
4. Mecanisme de sincronizare si comunicare sub QNX / POSIX. Buffere dinamice (pipes). Mesaje. Semnale.	2	
5. Planificarea taskurilor pe conditie de timp sub QNX / POSIX. Alarmer. Timere.	2	
6. Prezentarea pachetului Real-Time Java. Metode proiectare a aplicatiilor de timp-real. Masini de stare finita si programe Java.	2	
Bibliografie ¹⁵ 1. Dragoicea, M., Programarea aplicatiilor in timp real. Editura Universitara, Bucuresti, 2009. 2. BOIAN F.M., FERDEAN C. M., BOIAN R.F. DRAGOS R.C. Programare concurenta pe platforme Unix, Windows, Editura Albastra - grupul Microinformatica, Cluj, 2002. 3. Wellings, A., Concurrent and Real-Time Programming in Java, John Wiley, 2004.		

¹³ Cel puțin un un titlu trebuie să aparțină colectivului disciplinei iar cel puțin un titlu trebuie să se refere la o lucrare de referință pentru disciplină, de circulație națională și internațională, existentă în biblioteca UPT.

¹⁴ Tipurile de activități aplicative sunt cele precizate în nota de subsol 5. Dacă disciplina conține mai multe tipuri de activități aplicative atunci ele se trec consecutiv în liniile tabelului de mai jos. Tipul activității se va înscrie într-o linie distinctă sub forma: „Seminar:”, „Laborator:”, „Proiect:” și/sau „Practică:”.

¹⁵ Cel puțin un titlu trebuie să aparțină colectivului disciplinei.

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

- Disciplina este din categoria disciplinelor fundamentale, și se regăsește în planul de învățământ (eventual sub denumiri apropiate) al tuturor programelor de studii din domeniul Inginerie Electrică și din alte domenii de studii din România. De asemenea și în planul de învățământ al programelor de studii din străinătate se regăsește aceasta disciplină:
- http://www.howard.edu/ceacs/departments/electrical/Program_BSEE.htm
- <http://www.ee.ucl.ac.uk/syllabus/syllabusELEC1011-4.pdf>
- <http://cegt201.bradley.edu/coursework/bsee.shtml>

10. Evaluare

Tip activitate	10.1 Criterii de evaluare ¹⁶	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	4 subiecte cu caracter aplicativ care demonstrează însușirea elementelor prezentate la curs	Examen pe calculator	0,66
10.5 Activități aplicative	S:		
	L: Lucrări de control, teme de casă și răspunsurile la întrebările puse la laborator	Lucrări de control pe calculator, teme pe suport electronic, conversație	0,34
	P¹⁷:		
	Pr:		
10.6 Standard minim de performanță (se prezintă cunoștințele minim necesare pentru promovarea disciplinei și modul în care se verifică stăpânirea lor¹⁸)			
Studentul va promova disciplina dacă reușește să implementeze două dintre aplicațiile propuse.			
<ul style="list-style-type: none"> • Programele realizate trebuie să fie macar compilate corect. 			

Data completării

04.09.2017

**Titular de curs
(semnătura)**

.....

**Titular activități aplicative
(semnătura)**

.....

**Director de departament
(semnătura)**

.....

Data avizării în Consiliul Facultății¹⁹

06.09.2017

**Decan
(semnătura)**

.....

¹⁶ Fișele disciplinelor trebuie să conțină procedura de evaluare a disciplinei cu precizarea criteriilor, a metodelor și a formelor de evaluare, precum și cu precizarea ponderilor atribuite acestora în nota finală. Criteriile de evaluare se formulează în mod distinct pentru fiecare activitate prevăzută în planul de învățământ (curs, seminar, laborator, proiect). Ele se vor referi și la formele de verificare pe parcurs (teme de casă, referate ș.a.)

¹⁷ În cazul când proiectul nu este o disciplină distinctă, în această rubrică se va preciza și modul în care rezultatul evaluării proiectului condiționează admiterea studentului la evaluarea finală din cadrul disciplinei.

¹⁸ Nu se va explica cum se acorda nota de promovare.

¹⁹ Avizarea este precedată de discutarea punctului de vedere al board-ului de care aparține programul de studii cu privire la fișa disciplinei.