

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	Limbaje de asamblare / DS						
2.2 Titularul activităților de curs	Prof. dr. Ing. Pănoiu Manuela						
2.3 Titularul activităților aplicative ⁶	Șef lucr. dr. ing. Abrudean Cristian						
2.4 Anul de studii ⁷	III	2.5 Semestrul	II	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei ⁸	DI

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	4 , format din:	3.2 ore curs	2	3.3 ore seminar/laborator/proiect	2
3.1* Număr total de ore asistate integral/sem.	56 , format din:	3.2* ore curs	28	3.3* ore seminar/laborator/proiect	28
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	3,14 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			1
		ore studiu individual după manual, suport de curs, bibliografie și notițe			1
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			1,14
3.7* Număr total de ore activități neasistate/semestru	44 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			14
		ore studiu individual după manual, suport de curs, bibliografie și notițe			14
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			16
3.8 Total ore/săptămână ¹⁰	7,14				
3.8* Total ore/semestru	100				
3.9 Număr de credite	4				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Cunoștințe de algoritmică și programarea calculatoarelor, sisteme bazate pe microprocesoare
4.2 de competențe	• Programare C

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 disciplină 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	• Sală de curs, Materiale suport: laptop, proiector, tablă.
5.2 de desfășurare a activităților practice	• Laborator cu 12-14 calculatoare, tablă.

6. Competențe la formarea cărora contribuie disciplina

Competențe specifice	C4 C4.1 Descrierea arhitecturilor de bază pentru sistemele informatice aplicate în conducerea sistemelor energetice sau industriale. C4.2 Explicarea și interpretarea funcționării elementelor sistemelor informatice aferente conducerii proceselor energetice sau industriale; C4.3 Alegerea elementelor unui sistem informatic destinat conducerii, comenzii, reglajului sau supravegherii unui proces energetic sau industrial; C4.4 Utilizarea criteriilor și metodelor de evaluare a performanțelor tehnice și informatice ale unui sistem informatic de proces; C4.5 Implementarea unei structuri de sistem informatic de conducere a proceselor din sistemele energetice sau industriale.. C5 C5.1 Descrierea structurilor de conducere automată bazate pe microprocesoare și microcontrolere; C5.2 Explicarea utilizării microprocesoarelor și microcontrolerelor și cunoașterea softului aferent acestora; C5.3 Modelarea, simularea și testarea sistemelor de conducere automată a proceselor industriale; C5.4 Evaluarea performanțelor de regim staționar și dinamic ale sistemelor de conducere automată; C5.5 Realizarea unui sistem de comandă și reglare automată a unui proces industrial specific domeniului specializării. C6 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. •
Competențele profesionale în care se înscriu competențele specifice	C4 Realizarea și implementarea sistemelor informatice de conducere, comandă, reglaj și supraveghere a proceselor energetice sau industrial C5 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	• Însușirea și cunoașterea principalelor caracteristici ale limbajelor de asamblare
7.2 Obiectivele specifice	Dezvoltarea și depanarea programelor în limbajul de asamblare pentru Intel 8x86 si 8x87, folosind programarea modulara bazata pe utilizarea subprogramelor, functiilor sistem si macroinstruciunilor definite. Abilitatea de a apela în programe scrise in limbaje de nivel înalt a procedurilor scrise în

	limbaj de asamblare. Dezvoltarea abilității de a scrie programe eficiente și fiabile. Structurarea registrelor și modul de adresare a memoriei la familia x86 Cunoașterea principalelor grupe de instrucțiuni I80x86; Scrierea procedurilor, moduri de transmitere a parametrilor, definirea și utilizarea macroinstrucțiunilor; Crearea și utilizarea fișierelor pentru stocarea și regasirea informațiilor pe disc. Structura coprocesorului de virgula flotantă 80x87 și scrierea programelor ce utilizează numere reale. Capacitatea de a implementa programe mixte scrise în limbaj de nivel înalt și în limbaj de asamblare
--	--

8. Conținuturi¹¹

8.1 Curs	Număr de ore	Metode de predare ¹²
1. ARHITECTURA MICROPROCESORULUI INTEL x86, SI MODURI DE ADRESARE 1.1. Structura microprocesorului I 8x86. 1.2. Sistemul de operare MSDOS, comenzi și structură. 1.3. Editarea, asamblarea, link-editarea și rularea programelor. 1.4. Depanarea programelor cu AFD(Advanced Full Screen Debugger) și CodView	4	Expunerea informațiilor esențiale pe videoproiector. Detalierea informațiilor expuse Demonstrarea utilizând un mediu de programare Conversația
2. INSTRUCȚIUNII8086/I8x86 2.1 Macroasamblorul MASM86, structura programe, definire date. 2.2. Moduri de adresare. 2.3. Instrucțiuni de mutare. 2.4. Instrucțiuni aritmetice și logice. 2.5. Calcul în zecimal pe lungime variabilă. 2.6. Programe ramificate și ciclice, instrucțiuni de salt, tablouri. 2.7. Prelucrare texte, instrucțiuni pe șiruri de caractere. 2.8. Structuri de date. 2.9. Asamblări condiționate.	8	
3. SUBPROGRAME 3.1 Clasificare subprograme. 3.2. Transmitere parametrii prin registre și prin tabela de adrese. 3.3. Subprograme de conversie binar-zecimal și zecimal-bină. 3.4. Transmiterea param. printr-o zonă de memorie cu structura dată. 3.5. Subprograme reentrante. Transmiterea parametrilor prin stivă, variabile locale. Subprograme recursive. 3.6. Definirea macroinstrucțiuni pentru chemarea subprogramelor. 3.7. Comunicarea între programele scrise în limbaj de nivel înalt și limbaj de asamblare(C)	8	
4. UTILIZARE FUNCTII MS-DOS 4.1 Funcții MSDOS standard de intrare-iesire. 4.2. Funcții MSDOS pentru prelucrare fișiere prin FCB. 4.3. Funcții MSDOS pentru fișiere cu indicator logic. 4.4. Funcții de tratare a devierilor - subprograme rezidente. 4.5. Alte funcții MSDOS	4	
5. UTILIZAREA COPROCESOR MATEMATIC 80x87 5.1 Structura coprocesor matematic și mod de lucru. 5.2. Structura informațiilor în virgula flotantă. 5.3. Instrucțiuni de transfer și conversie. Instrucțiuni aritmetice. 5.4. Conversie din binar virgula flotantă și invers.	4	

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.).

Bibliografie ¹³ 1. Pănoiu M., Limbaje de asamblare, note de curs , https://cv.upt.ro/course/view.php?id=2713/ 2. Vasile Lungu, PROCESOARE INTEL, PROGRAMARE IN LIMBAJ DE ASAMBLARE, EDITIA A II-A, Editura Teora, 2005 3. Muscă Gh., Programarea în limbaj de asamblare, Ed. Teora, București 1996		
8.2 Activități aplicative¹⁴	Număr de ore	Metode de predare
1. Structura microprocesor I8086, utilizare Debugger AFD 2. Moduri de adresare, instrucțiuni de transfer.	4	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare
3. Utilizare MASM, LINK, CodeView, structura programe, definire date, segmentare, instrucțiuni aritmetice și logice. 4. Programe ramificate și ciclice, prelucrare tablouri.	4	
5. Prelucrare siruri de caractere. (2 ore) 6. Subprograme. Transmitere param. prin registre.	4	
7. Conversie hexa-ASCII, zec-binar, binar- zec.	2	
8. Transmiterea parametrilor printr-o zona de memorie, def. structuri de date	2	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare
9. Transmiterea parametrilor prin stivă, alocare variabile locale 10. Subprograme reentrante și recursive	4	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare
11. Funcții MSDOS standard (de I/E). Definire macroinstrucțiuni 12. Funcții MSDOS pentru fisiere cu FCB și/sau indicator logic	4	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare
13. Comunicarea între programe scrise în limbaje de nivel înalt și limbaj de asamblare	2	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare
14. Utilizare instrucțiuni de virgulă flotantă	2	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare
Bibliografie ¹⁵ 1. Abrudean C., Limbaje de asamblare, s , https://cv.upt.ro/course/view.php?id=2713 2. Vasile Lungu, PROCESOARE INTEL, PROGRAMARE IN LIMBAJ DE ASAMBLARE, EDITIA A II-A, Editura Teora, 2005 3. Muscă Gh., Programarea în limbaj de asamblare, Ed. Teora, București 1996.		

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

- Disciplina este din categoria disciplinelor de specialitate fiind o disciplină solicitată de toate companiile cu profil IT deoarece asigură dobândirea de abilități necesare programării pe sisteme integrate (embedded systems)
--

10. Evaluare

Tip activitate	10.1 Criterii de evaluare¹⁶	10.2 Metode de evaluare	10.3 Pondere din nota finală
----------------	---	--------------------------------	-------------------------------------

¹³ Cel puțin 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.

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.)

10.4 Curs	Abilitatea de a programa în limbajul de asamblare al familiei de microprocesoare Intel x86	Examen pe campusul virtual sau examen scris. Subiectele constau în întrebări/aplicații . Studenții vor avea acces la bibliografie	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	34%
	P¹⁷:		
	Pr:		
10.6 Standard minim de performanță (se prezintă cunoștințele minim necesare pentru promovarea disciplinei și modul în care se verifică stăpânirea lor ¹⁸)			
Studentul va promova disciplina dacă reușește să identifice și să aplice corect metoda de rezolvare adecvată pentru jumătate dintre aplicațiile propuse ca subiecte de examen			

Data completării

05.10.2023

**Director de departament
(semnătura)**

.....

**Titular de curs
(semnătura)**

.....

Data avizării în Consiliul Facultății¹⁹

16.10.2023

**Titular activități aplicative
(semnătura)**

.....

**Decan
(semnătura)**

.....



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.