FIȘA DISCIPLINEI

1. Date despre program

Facultatea	Inginerie Electrică şi Ştiinţa Calculatoarelor
Departamentul	Calculatoare, Electronică şi Automatică
Domeniul de studii	Automatică, informatică aplicată şi sisteme inteligente
Ciclul de studii	Licenţă, dual
Programul de studii	Automatică şi informatică aplicată

2. Date despre disciplină

Denumirea disciplinei	LIMBAJE DE ANSAMBLARE				
Anul de studiu	II	Semestrul	4	Tipul de evaluare	E
Regimul disciplinei	Categoria formativă a disciplinei DF - fundamentală, DS - de specializare, DC – complementară				DF
	Categoria de opționalitate a disciplinei: DOB – obligatorie, DOP – opțională, DFA - facultativă				DOB

3. Timpul total estimat (ore alocate activităților didactice)

I a) Număr de ore pe săptămână	Total general	3	Curs	1	Seminar		Laborator IIS	2	Proiect IIS		Practică IIS	
							Laborator IM		Proiect IM		Practică IM	
I b) Totalul de ore pe semestru din planul de învățământ		42	Curs	14	Seminar		Laborator	28	Proiect		Practică	

(*IIS – instituție de învățământ superior; IM – învățare prin muncă*)

Distribuţia fondului de timp pe semestru:	Ore IIS	Ore IM
II. Studiul după manual, suport de curs, bibliografie şi notiţe	5	
III. Examinări	3	
IV. Alte activităţi (precizaţi):		

Total ore studiu individual (II.a+II.b+III)	Ore IIS	8	Ore IM	
Total ore pe semestru (I.b+II.a+II.b+III+IV)	Ore IIS	50	Ore IM	
Numărul de credite	Credite IIS	2	Credite IM	

4. Competențe specifice acumulate

Competențe transversale	CT6. Îşi asumă responsabilitatea
Competențe profesionale	CP2. Proiectează componente de automatizare CP5. Proiectează o rețea de calculatoare

5. Rezultatele învățării

Cunoştinţe	Aptitudini	Responsabilitate şi autonomie
Studentul descrie, identifică, sumarizează concepte şi metode elementare privitoare la limbaje de programare la nivel maşina (programare în limbaj de asamblare), medii de programare (DOS Debug sau Turbo-Assembler (TASM) şi Macro-Assembler (MASM)), tehnici de programare,inginerie software şi modul lor de aplicare în probleme concrete.	Studentul analizează, elaborează, dezvoltă şi testează programe în limbaj de asamblare pentru diferite arhitecturi de calcul: I8086 şi ARM.	Studentul descrie, identifică, sumarizează concepte şi metode elementare privitoare la limbaje de programare la nivel maşina (programare în limbaj de asamblare), medii de programare (DOS Debug sau Turbo-Assembler (TASM) şi Macro-Assembler (MASM)), tehnici de programare,inginerie software şi modul lor de aplicare în probleme concrete.

6. Obiectivele disciplinei (reieşind din grila competenţelor specifice acumulate)

Obiectivul general al disciplinei	Însuşirea aspectelor fundamentale privind dezvoltarea de aplicaţii utilizând limbajul de asamblare, pentru diferite arhitecturi de calcul RISC/CISC cu creşterea interesului faţă de programarea low level.

7. Conținutul predării și învățării

Curs	Nr. ore	Metode de predare	Observaţii
1. Introducere. 1.1 Arhitectura procesorului I8086. 1.2 Unitatea de executie. 1.3 Unitatea de interfata cu magistrala. 1.4 Registre de uz general. 1.5 Moduri de adresare.	2h	expunerea, prelegerea-dezbatere, demonstraţia	
2. Setul de instructiuni 8086. 2.1 Instructiuni de transfer. 2.2 Instructiuni aritmetice si logice. 2.3 Instructiuni pentru operatii cu siruri de caractere/cuvinte. 2.4 Instructiuni de apel de procedura si de salt. 2.5 Instructiuni pentru controlul procesorului. 2.6 Dezvoltarea programelor in limbaj de asamblare.	2h	expunerea, prelegerea-dezbatere, demonstraţia	
3. Structura programelor. Definirea si initializarea datelor. Operatori. 3.1 Segmentare. Directive pentru definirea segmentelor. 3.2 Directive pentru legarea modulelor. 3.3 Definirea si initializarea datlor. 3.4 Definirea etichetelor. 3.5 Definirea structurilor. 3.6 Operatori in limbajul de asamblare. 3.7 Directive de asamblare conditionata.	2h	expunerea, prelegerea-dezbatere, demonstraţia	
4. Macroinstructiuni. 4.1 Scopul macroinstructiunilor. Definire si expandare. 4.2 Macroinstructiuni cu parametri. 4.3 Macroinstructiuni repetitive. 4.4 Invocarea recursiva. 4.5 Tehnici avansate de utilizare a macroinstructiunilor.	2h	expunerea, prelegerea-dezbatere, demonstraţia	
5. Tehnici de programare in limbaj de asamblare. 5.1 Decizia simpla si decizia compusa. 5.2 Cicluri cu test la partea superioara si inferioara. 5.3 Selectia. Tabele de salt sau de apel de proceduri. 5.4 Transferul parametrilor catre proceduri. 5.5 Intoarcerea datelor da catre proceduri. 5.6 Proceduri cu zone de date proprii. 5.7 Proceduri recursive. 5.8 Proceduri cu număr variabil de parametri. 5.9 Tehnici avansate de programare.	2h	expunerea, prelegerea-dezbatere, demonstraţia	
6. Programarea in limbaj de asamblare ARM. 6.1 Manipularea datelor. 6.2 Incarcarea constantelor in registri. 6.3 Incarcarea unei date de memorie intr-un registru. 6.4 Memorarea datelor din regitri in memorie. 6.5 Conversia unei instructiuni simple C in limbaj de asamblare ARM. 6.6 Calcularea adreselor de memorie. 6.7 Exemple pentru adresarea memoriei. 6.8 Instructiuni pentru lucrul cu stiva. 6.9 Instructiuni de procesare a datelor.	2h	expunerea, prelegerea-dezbatere, demonstraţia	
7. Structurile de control in asamblarea ARM. 7.1 Secventierea instructiunilor. 7.2 Implementarea deciziilor. 7.3 Implementarea buclelor. 7.4 Implementarea functiilor.	2h	expunerea, prelegerea-dezbatere, demonstraţia	

Bibliografie minimală recomandată

1. Bazele Microprocesoarelor – Liviu Kreindler – Matrix Rom Bucuresti, 1997
2. Familia de Microcontrolere MCS51 - Vasile Gaitan, 1997
3. Athanasiu I, Panoiu Al - Microprocesoarele 8086,286,386 Ed. TEORA 1992
4. Procesoare Intel, Programare în Limbaj de asamblare, ediţia II a, - Vasile Lungu, 2004
5. Programare in limbaj de asamblare, - Gheorghe Musca. Editura Teora. 1998
6. Fundamentals of Embedded Software with the ARM Cortex-M3, Second Edition, Daniel W. Lewis, 2013
7. Modern Arm Assembly Language Programming, Covers Armv8-A 32-bit, 64-bit and SIMD, Daniel Kusswurm, 2020

Aplicaţii (Seminar/**laborator**/lucrări practice/proiect)	Nr. ore	Metode de predare	Observaţii
Laborator 1. Norme de protecţia muncii. Noţiuni introductive ale Programării în Limbaj de Asamblare	2h	lucrări practice, experimentul	

2. Reprezentarea datelor. Setul de instrucțiuni 8086. Aritmetica pe 16 biți	2h		
3. Aritmetica pe 32 biți. Conversii de lungime a datelor. Instrucțiuni condiționale	2h		
4. Tablouri. Operații pe biți	2h		
5. Stiva. Subrutine. Macroinstrucțiuni	2h		
6. Moduri de transmitere a parametrilor	2h		
7. Prelucrarea sirurilor de parametri	2h		
8. Structuri. Întreruperi. Evaluarea cunoștințelor	2h		
9. Setul de instrucțiuni ARM.	2h		
10. Aritmetica pe 16 si 32 de biți	2h		
11. Instrucțiuni condiționale ARM	2h		
12. Instrucțiuni repetitive ARM	2h		
13.Translatare cod C in cod ARM	2h		
14. Evaluarea cunoștințelor. Recuperări	2h		

Bibliografie minimală recomandată

1. Bazele Microprocesoarelor – Liviu Kreindler – Matrix Rom Bucuresti, 1997
2. Familia de Microcontrolere MCS51 - Vasile Gaitan, 1997
3. Athanasiu I, Panoiu Al - Microprocesoarele 8086,286,386 Ed. TEORA 1992
4. Procesoare Intel, Programare în Limbaj de asamblare, ediția II a, - Vasile Lungu, 2004
5. Programare in limbaj de asamblare, - Gheorghe Musca. Editura Teora. 1998
6. Fundamentals of Embedded Software with the ARM Cortex-M3, Second Edition, Daniel W. Lewis, 2013
7. Modern Arm Assembly Language Programming, Covers Armv8-A 32-bit, 64-bit and SIMD, Daniel Kusswurm, 2020

8. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Evaluare finală	Participarea activă în timpul cursurilor.	Evaluare continuă	**10 %**
	Insusirea notiunilor de baza si utilizarea corecta a limbajului de specialitate	Evaluare prin probă finală de tip test practic (oral) și scris (Moodle) din problemele furnizate la curs.	**40%**
Evaluare pe parcurs	Finalizarea lucrările practice propuse	evaluare continuă (prin metode orale şi probe practice)	**30%**
	Rezolvarea problemelor propuse in temele de acasă	evaluare sumativă (prin prezentarea orală publică, utilizând slide-uri, a rezolvării temei de casă).	**20%**

Fișa disciplinei include, dacă este cazul, elemente adaptate persoanelor cu dizabilități, în funcție de tipul și gradul acestora.

Data completării	Grad didactic, nume, prenume, semnătura titularului de curs	Grad didactic, nume, prenume, semnătura titularului de aplicație
22.09.2026	As. univ. dr. ing. Simona-Anda TCACIUC	As. univ. dr. ing. Simona-Anda TCACIUC

Data avizării	Grad didactic, nume, prenume, semnătura responsabilului de program
23.09.2026	Ş.l.dr.ing. Corneliu BUZDUGA

Data avizării în departament	Grad didactic, nume, prenume, semnătura directorului de departament
24.09.2026	Conf.dr.ing. Eugen COCA

Data aprobării în consiliul facultății	Grad didactic, nume, prenume, semnătura decanului
25.09.2026	Prof.dr.ing. Laurenţiu-Dan MILICI