

FIȘA DISCIPLINEI

1. Date despre program

Facultatea	Inginerie Electrică și Știința Calculatoarelor
Departamentul	Calculatoare, Electronică și Automatică
Domeniul de studii	Ingineria sistemelor
Ciclul de studii	Licență, dual
Programul de studii	Automatică și informatică aplicată

2. Date despre disciplină

Denumirea disciplinei	LIMBAJE DE ANSAMBLARE				
Anul de studiu	II	Semestrul	4	Tipul de evaluare	E
Regimul disciplinei	Categorica formativă a disciplinei DF - fundamentală, DS - de specializare, DC – complementară				DF
	Categorica 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	4	Curs	2	Seminar	Laborator IIS	2	Proiect IIS	Practică IIS
I b) Totalul de ore pe semestru din planul de învățământ		56	Curs	28	Seminar	Laborator IM	28	Proiect IM	Practică IM

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

II Distribuția fondului de timp pe semestru:	Ore IIS	Ore IM
II a) Studiul după manual, suport de curs, bibliografie și notițe	19	22
II b) Tutoriat (pentru ID)		
III Examinări	3	
IV Alte activități (precizați):		

Total ore studiu individual (II.a+II.b+III)	Ore IIS	22	Ore IM	22
Total ore pe semestru (I.b+II.a+II.b+III+IV)	Ore IIS	50	Ore IM	50
Numărul de credite	Credite IIS	2	Credite IM	2

4. Competențe specifice acumulate

Competențe profesionale/generale	CP2. Operarea cu concepte fundamentale din știința calculatoarelor, tehnologia informației și comunicațiilor C5. Dezvoltarea de aplicații și implementarea algoritmilor și structurilor de conducere automată, utilizând principii de management de proiect, medii de programare și tehnologii bazate pe microcontrolere, procesoare de semnal, automate programabile, sisteme încorporate
Competențe transversale	

5. Rezultatele învățării

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

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

Obiectivul general al disciplinei	Înșușirea aspectelor fundamentale privind dezvoltarea de aplicații utilizând limbajul de asamblare, pentru diferite arhitecturi de calcul RISC/CISC cu creșterea interesului față de programarea low level.
-----------------------------------	---

7. Conținutul predării și învățării

Curs	Nr. ore	Metode de predare	Observații
1. Introducere. 1.1 Arhitectura procesorului I8086. 1.2 Unitatea de execuție. 1.3 Unitatea de interfata cu magistrala. 1.4 Registre de uz general. 1.5 Moduri de adresare.	2h	expunerea, prelegerea-dezbateri, demonstrația	
2. Setul de instrucțiuni 8086. 2.1 Instrucțiuni de transfer. 2.2 Instrucțiuni aritmetice și logice. 2.3 Instrucțiuni pentru operații cu siruri de caractere/cuvinte. 2.4 Instrucțiuni de apel de procedură și de salt. 2.5 Instrucțiuni pentru controlul procesorului. 2.6 Dezvoltarea programelor în limbaj de asamblare.	4h	expunerea, prelegerea-dezbateri, demonstrația	
3. Structura programelor. Definirea și initializarea datelor. Operatori. 3.1 Segmentare. Directive pentru definirea segmentelor. 3.2 Directive pentru legarea modulelor. 3.3 Definirea și initializarea datelor. 3.4 Definirea etichetelor. 3.5 Definirea structurilor. 3.6 Operatori în limbajul de asamblare. 3.7 Directive de asamblare condiționată.	2h	expunerea, prelegerea-dezbateri, demonstrația	
4. Macroinstrucțiuni. 4.1 Scopul macroinstrucțiunilor. Definire și expansiune. 4.2 Macroinstrucțiuni cu parametri. 4.3 Macroinstrucțiuni repetitive. 4.4 Invocarea recursivă. 4.5 Tehnici avansate de utilizare a macroinstrucțiunilor.	2h	expunerea, prelegerea-dezbateri, demonstrația	
5. Tehnici de programare în limbaj de asamblare. 5.1 Decizia simplă și decizia compusă. 5.2 Cicli cu test la partea superioară și inferioară. 5.3 Selecția. Tabele de salt sau de apel de proceduri. 5.4 Transferul parametrilor către proceduri. 5.5 Întoarcerea datelor de la 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.	6h	expunerea, prelegerea-dezbateri, demonstrația	
6. Programarea în limbaj de asamblare ARM. 6.1 Manipularea datelor. 6.2 Încărcarea constantelor în registre. 6.3 Încărcarea unei date de memorie într-un registru. 6.4 Memorarea datelor din registre în memorie. 6.5 Conversia unei instrucțiuni simple C în limbaj de asamblare ARM. 6.6 Calcularea adreselor de memorie. 6.7 Exemple pentru adresarea memoriei. 6.8 Instrucțiuni pentru lucrul cu stivă. 6.9 Instrucțiuni de procesare a datelor.	6h	expunerea, prelegerea-dezbateri, demonstrația	
7. Structurile de control în asamblarea ARM. 7.1 Secvențierea instrucțiunilor. 7.2 Implementarea deciziilor. 7.3 Implementarea buclelor. 7.4 Implementarea funcțiilor.	6h	expunerea, prelegerea-dezbateri, demonstrația	

Bibliografie minimală recomandată

1. Bazele Microprocesoarelor – Liviu Kreindler – Matrix Rom București, 1997
2. Familia de Microcontrolere MCS51 - Vasile Gaitan, 1997
3. Athanasii I, Panoiu AI - Microprocesoarele 8086,286,386 Ed. TEORA 1992
4. Procesoare Intel, Programare în Limbaj de asamblare, ediția II a, - Vasile Lungu, 2004
5. Programare în 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			
Aplicații (Seminar/ laborator /lucrări practice/proiect)	Nr. ore	Metode de predare	Observații
Laborator			
1. Norme de protecția muncii. Noțiuni 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 și 32 de biți	2h		
11. Instrucțiuni condiționale ARM	2h		
12. Instrucțiuni repetitive ARM	2h		
13. Traducere cod C în 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 AI - Microprocesoarele 8086,286,386 Ed. TEORA 1992 4. Procesoare Intel, Programare în Limbaj de asamblare, ediția II a, - Vasile Lungu, 2004 5. Programare în 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
Curs	Participarea activă în timpul cursurilor.	Evaluare continuă	10 %
	Însușirea noțiunilor de bază și utilizarea corectă a limbajului de specialitate	Evaluare prin probă finală de tip test practic (oral) și scris (Moodle) din problemele furnizate la curs.	40%
Seminar			
Laborator IIS			
Laborator IM	Finalizarea lucrărilor practice propuse	evaluare continuă (prin metode orale și probe practice)	30%
	Rezolvarea problemelor propuse în temele de acasă	evaluare sumativă (prin prezentarea orală publică, utilizând slide-uri, a rezolvării temei de casă).	20%
Proiect IIS			
Proiect IM			

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
24.09.2025	As. univ. dr. ing. Simona-Anda TCACIUC	As. univ. dr. ing. Simona-Anda TCACIUC

Data avizării	Grad didactic, nume, prenume, semnătura responsabilului de program
25.09.2025	Ș.I.dr.ing. Corneliu BUZDUGA

Data avizării în departament	Grad didactic, nume, prenume, semnătura directorului de departament
25.09.2025	Conf.dr.ing. Eugen COCA

Data aprobării în consiliul facultății	Grad didactic, nume, prenume, semnătura decanului
26.09.2025	Prof.dr.ing. Laurențiu-Dan MILICI