

FIȘA DISCIPLINEI

1. Date despre program

Instituția de învățământ superior	Universitatea „Ștefan cel Mare” Suceava
Facultatea	Facultatea de Inginerie Electrică și Știința Calculatoarelor
Departamentul	Departamentul de Calculatoare
Domeniul de studii	Calculatoare și tehnologia informației
Ciclul de studii	Licență
Programul de studii/calificarea	Calculatoare / Inginer

2. Date despre disciplină

Denumirea disciplinei	PROGRAMAREA CALCULATOARELOR SI LIMBAJE DE PROGRAMARE III				
Titularul activităților de curs	Conf. dr.ing Cristian Andy TANASE				
Titularul activităților de aplicative	As. Ing. Dr. Anda TCACIUC				
Anul de studiu	II	Semestrul	IV	Tipul de evaluare	Examen
Regimul disciplinei	Categoría formativă a disciplinei DF - fundamentală, DD - în domeniu, DS - de specialitate, DC - complementară				DF
	Categoría de opționalitate a disciplinei: DO - obligatorie (impusă), DA - opțională (la alegere), DL - facultativă (liber aleasă)				DO

3. Timpul total estimat (ore alocate activităților didactice)

I a) Număr de ore pe săptămână	4	Curs	2	Seminar	Laborator/lucrări practice	2	Proiect	
I b) Totalul de ore pe semestru din planul de învățământ	56	Curs	28	Seminar	Laborator/lucrări practice	28	Proiect	

II Distribuția fondului de timp pe semestru:	ore
II a) Studiul după manual, suport de curs, bibliografie și notițe	17
II b) Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	10
II c) Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	10
II d) Tutoriat	4
III Examinări	3
IV Alte activități:	0

Total ore studiu individual II (a+b+c+d)	41
Total ore pe semestru (I+II+III+IV)	100
Numărul de credite	4

4. Precondiții (acolo unde este cazul)

Curriculum	<ul style="list-style-type: none"> • Programarea calculatoarelor și limbajelor de programare
Competențe	C1. Operarea cu fundamente științifice, ingineresti ale informaticii C2. Proiectarea componentelor hardware, software si de comunicatii C3. Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor

5. Condiții (acolo unde este cazul)

Desfășurare a cursului	<ul style="list-style-type: none"> • PC, videoproiector 	
Desfășurare aplicații	Seminar	<ul style="list-style-type: none"> •
	Laborator/lucrări practice	laborator dotat cu minim 12 calculatoare PC cu sistem de operare Windows si TASM GuiTurboAssembler instalat.

6. Competențe specifice acumulate

Competențe profesionale	CP1. Efectueaza cercetare stiintifica CP2. Analizeaza grupuri masive de date CP3. Administreaza sisteme de colectare a datelor
-------------------------	--

Competențe transversale	CT1 Lucreaza în echipa CT2 Isi asuma responsabilitatea
-------------------------	---

7. **Obiectivele disciplinei** (reieșind din grila competențelor specifice acumulate)

Obiectivul general al disciplinei	<ul style="list-style-type: none"> Dezvoltarea de aplicații utilizând limbajul de asamblare. Manifestarea interesului față de programarea în limbaj de asamblare.
Obiective specifice	<ul style="list-style-type: none"> Prezentarea limbajului de asamblare aferente familiei de procesoare I8086. În cadrul orelor de laborator studenții sunt familiarizați cu diverse metode de programare, utilizând resursele și particularitățile microprocesoarelor studiate. Se urmărește însușirea principalelor modalități de realizare practică a unor programe editate direct în Assembler sub diferite aplicații cum ar fi sub comanda DOS Debug sau Turbo-Assembler (TASM) și Macro-Assembler (MASM).

8. **Conținuturi**

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. Definierea și inițializarea datelor. Operatori. 3.1 Segmentare. Directive pentru definierea segmentelor. 3.2 Directive pentru legarea modulelor. 3.3 Definierea și inițializarea datelor. 3.4 Definierea etichetelor. 3.5 Definierea 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. Definiție și extindere. 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.	6h	expunerea, prelegerea-	

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.		dezbatere, demonstrația	
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.	6h	expunerea, prelegerea- dezbatere, demonstrația	
7. Structurile de control in asamblarea ARM. 7.1 Secventierea instructiunilor. 7.2 Implementarea deciziilor. 7.3 Implementarea buclelor. 7.4 Implementarea functiilor.	6h	expunerea, prelegerea- dezbatere, demonstrația	

Bibliografie

1. Practica dezvoltarii software in limbaje de asamblare – Ion Ivan, Paul Pocatilu, Doru Cazan Editura Economica, 2002
2. Totul despre Z80 – M. PATRUBANY – Editura Tehnica, 1989
3. Bazele Microprocesoarelor – Liviu Kreindler – Matrix Rom Bucuresti, 1997
4. Familia de Microcontrolere MCS51 - Vasile Gaitan, 1997
5. Athanasiu I, Panoiu AI - Microprocesoarele 8086,286,386 Ed. TEORA 1992
6. Procesoare Intel, Programare în Limbaj de asamblare, ediția II a, - Vasile Lungu, 2004
7. Programare in limbaj de asamblare, - Gheorghe Musca. Editura Teora. 1998
8. Fundamentals of Embedded Software with the ARM Cortex-M3, Second Edition, Daniel W. Lewis, 2013
9. Modern Arm Assembly Language Programming, Covers Armv8-A 32-bit, 64-bit and SIMD, Daniel Kusswurm, 2020

Bibliografie minimală

1. Practica dezvoltarii software in limbaje de asamblare – Ion Ivan, Paul Pocatilu, Doru Cazan Editura Economica, 2002
2. Programare in limbaj de asamblare, - Gheorghe Musca. Editura Teora. 1998
3. Fundamentals of Embedded Software with the ARM Cortex-M3, Second Edition, Daniel W. Lewis, 2013
4. Modern Arm Assembly Language Programming, Covers Armv8-A 32-bit, 64-bit and SIMD, Daniel Kusswurm, 2020

Aplicații (laborator)	Nr. ore	Metode de predare	Observații
<ul style="list-style-type: none"> • 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 si 32 de biți	2h		
11. Instrucțiuni condiționale ARM	2h		
12. Instrucțiuni repetitive ARM	2h		
13. Traducere cod C in cod ARM	2h		

14. Evaluarea cunoștințelor. Recuperări	2h		
Bibliografie			
1. Practica dezvoltării software în limbaje de asamblare – Ion Ivan, Paul Pocatilu, Doru Cazan Editura Economica, 2002 2. Totul despre Z80 – M. PATRUBANY – Editura Tehnica, 1989 3. Bazele Microprocesoarelor – Liviu Kreindler – Matrix Rom Bucuresti, 1997 4. Familia de Microcontrolere MCS51 - Vasile Gaitan, 1997 5. Athanasiu I, Panou AI - Microprocesoarele 8086,286,386 Ed. TEORA 1992 6. Procesoare Intel, Programare în Limbaj de asamblare, ediția II a, - Vasile Lungu, 2004 7. Programare în limbaj de asamblare, - Gheorghe Musca. Editura Teora. 1998 8. Fundamentals of Embedded Software with the ARM Cortex-M3, Second Edition, Daniel W. Lewis, 2013 9. Modern Arm Assembly Language Programming, Covers Armv8-A 32-bit, 64-bit and SIMD, Daniel Kusswurm, 2020			
Bibliografie minimală			
1. Practica dezvoltării software în limbaje de asamblare – Ion Ivan, Paul Pocatilu, Doru Cazan Editura Economica, 2002 2. Programare în limbaj de asamblare, - Gheorghe Musca. Editura Teora. 1998 3. Fundamentals of Embedded Software with the ARM Cortex-M3, Second Edition, Daniel W. Lewis, 2013 4. Modern Arm Assembly Language Programming, Covers Armv8-A 32-bit, 64-bit and SIMD, Daniel Kusswurm, 2020			

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

<ul style="list-style-type: none"> Conținutul cursului, al laboratorului și proiectului este în concordanță cu conținutul disciplinelor similare de la programele de studiu Calculatoare de la alte universități din țară și străinătate în proporție de aproximativ 70%. http://courses.engr.illinois.edu/ece390/books/artofasm/artofasm.html http://facstaff.uww.edu/nguyenh/uww_courses/compsci271/271syllabus.html http://www.zota.ase.ro/bti/IntroASM.pdf http://www.science.upm.ro/~traian/web_curs/Asm/start.html# http://ac.upg-ploiesti.ro/plan_inv/auto/la_f.pdf
--

10. Evaluare

10.1. Standard minim de performanță evaluare la curs

Standardul minim de performanță evaluarea la curs constă în realizarea în mod corect a temelor date la curs pe parcursul semestrului și de răspunderea corectă la minim patru întrebări din testul grila Moodle.

10.2. Standard minim de performanță evaluare la activitatea aplicativă

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	participarea activă în timpul cursurilor.	Evaluare continuă	10 %
	Însușirea notiunilor 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/lucrări practice	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%

Standard minim de performanță

- cunoașterea arhitecturii I8086
 - cunoașterea reprezentării informației
 - cunoașterea modurilor de adresare
 - cunoașterea indicatorilor de condiție
 - cunoașterea aritmeticilor binare și zecimale
- Standarde minime pentru nota 10:
- descrierea întreprinderilor
 - descrierea modurilor de adresare prin exemple
 - aplicații în limbaj de asamblare

Programa analitică / Fișa disciplinei

Data completării	Semnătura titularului de curs	Semnătura cadrului didactic coordonator
23.09.2024		

Data avizării	Semnătura responsabilului de program
24.09.2024	

Data avizării în departament	Semnătura directorului de departament
25.09.2024	

Data aprobării în consiliul facultății	Semnătura decanului
27.09.2024	