

FIȘA DISCIPLINEI

(licență)

1. Date despre program

Instituția de învățământ superior	Universitatea „Ștefan cel Mare” din Suceava
Facultatea	Facultatea de Inginerie Electrică și Știința Calculatoarelor
Departamentul	Departamentul de Calculatoare, Electronică și Automatică
Domeniul de studii	Inginerie electronică, telecomunicații și tehnologii informaționale
Ciclul de studii	Licență
Programul de studii	Rețele și software de telecomunicații / Electronică aplicată

2. Date despre disciplină

Denumirea disciplinei	PROGRAMAREA CALCULATOARELOR ȘI LIMBAJE DE PROGRAMARE II				
Titularul activităților de curs	As. dr. ing. Anda TCACIUC				
Titularul activităților aplicative	As. dr. ing. Anda TCACIUC				
Anul de studiu	II	Semestrul	4	Tipul de evaluare	E
Regimul disciplinei	Categorია formativă a disciplinei DF - fundamentală, DD - în domeniu, DS - de specialitate, DC - complementară				DF
	Categorია de opționalitate a disciplinei: DI - impusă, DO - opțională, DF				DI

3. Timpul total estimat (ore alocate activităților didactice)

I a) Număr de ore pe săptămână	5	Curs	2	Seminar		Laborator/lucrări practice	2	Proiect	1
I b) Totalul de ore pe semestru din planul de învățământ	70	Curs	28	Seminar		Laborator lucrări practice	28	Proiect	14

II Distribuția fondului de timp pe semestru:	ore
II a) Studiul după manual, suport de curs, bibliografie și notițe	27
II b) Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	17
II c) Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	8
II d) Tutoriat	
III Examinări	3
IV Alte activități:	

Total ore studiu individual II (a+b+c+d)	52
Total ore pe semestru (I+II+III+IV)	125
Numărul de credite	5

4. Precondiții (acolo unde este cazul)

Curriculum	
Competențe	

5. Condiții (acolo unde este cazul)

Desfășurare a cursului	• PC, videoproiector	
Desfășurare aplicații	Seminar	•
	Laborator / lucrări practice	laborator dotat cu minim 12 calculatoare PC cu sistem de operare Windows si TASM GuiTurboAssembler instalat.
	Proiect	

6. Competențe specifice acumulate

Competențe profesionale	C3. Aplicarea cunoștințelor, conceptelor și metodelor de bază privitoare la arhitectura sistemelor de calcul, microprocesoare, microcontrolere, limbaje și tehnici de programare C4. Elaborarea specificațiilor tehnice, achiziția, instalarea și exploatarea echipamentelor de comunicații, fixe și mobile, precum și planificarea, configurarea și integrarea serviciilor de telecomunicații și elemente de securitatea informației
Competențe transversale	

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.
-----------------------------------	---

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 interfață cu magistrala. 1.4 Registre de uz general. 1.5 Moduri de adresare.	2h	expunerea, prelegerea-dezbatere, 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-dezbatere, 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-dezbatere, 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-dezbatere, 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-dezbatere, 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.	6h	expunerea, prelegerea-dezbatere, demonstrația	

6.7 Exemple pentru adresarea memoriei. 6.8 Instrucțiuni pentru lucrul cu stiva. 6.9 Instrucțiuni de procesare a datelor.			
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-dezbatere, demonstrația	

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, 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 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

Aplicații (laborator / lucrări practice)	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 șirurilor 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		
<ul style="list-style-type: none"> • Proiect 			
Temele de proiect sunt individuale și constau în realizarea unor programe în limbaj de asamblare x86, cum ar fi, de exemplu, implementarea unui calculator matematic asemănător celui existent în cadrul sistemului de operare Windows, implementarea unor algoritmi de sortare a șirurilor de caractere, implementarea unor jocuri gen X și 0, operații cu matrici, calculul celui mai mic multiplu comun și a celui mai mare divizor comun a două numere.	14h		

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, 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

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

- 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 performanta evaluarea la curs consta in realizarea in mod corect a temelor date la curs pe parcursul semestrului si de raspunderea corecta 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	Nota acordată pentru participarea activă în timpul cursurilor.	Evaluare continuă	10 %
	Nota acordată la examinarea finală.	Evaluare prin probă finală de tip test practic (oral) și scris (Moodle) din problemele furnizate la curs	40%
Laborator/lucrări practice	Media notelor acordate la lucrările practice	Evaluare continuă (prin metode orale și probe practice)	20%
	Evaluarea temelor de acasă	Evaluare sumativă (prin prezentarea orală publică, utilizând slide-uri, a rezolvării temei de casă)	20%
Proiect	Evaluarea temelor primite la proiect	Evaluare sumativă (prin prezentarea orală și practică a îndeplinirii etapelor temelor de proiect).	10%

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 întreruperilor
- descrierea modurilor de adresare prin exemple
- aplicații în limbaj de asamblare

Data completării	Semnătura titularului de curs	Semnătura titularului de aplicație
19.09.2024		

Data avizării	Semnătura responsabilului de program
20.09.2024	

Data avizării în departament	Semnătura directorului de departament
23.09.2024	

Data aprobării în consiliul facultății	Semnătura decanului
27.09.2024	

