

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

CENTRUL UNIVERSITAR NORD DIN BAIA MARE

**FIȘA DISCIPLINEI****1. Date despre program**

1.1 Instituția de învățământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Științe
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii	Matematică
1.5 Ciclu de studii	<b>Licență</b>
1.6 Programul de studii / Calificarea	<b>Matematică informatică</b>
1.7 Forma de învățământ	IF – învățământ cu frecvență
1.8 Codul disciplinei	SMAIL205

**2. Date despre disciplină**

2.1 Denumirea disciplinei	Arhitectura sistemelor de calcul						
2.2 Aria de conținut							
2.3 Responsabil de curs	Conf.univ.dr. Ioana ZELINA ioanazelina@yahoo.com						
2.4 Titularul activităților de seminar / laborator / proiect	Asist. univ. drd. Mara HAJDU-MĂCELARU						
2.5 Anul de studiu	I	2.6 Semestrul	2	2.7 Tipul de evaluare	C	2.8 Regimul disciplinei	DOB

**3. Timpul total estimat**

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar / laborator	2
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar / laborator	28
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					14
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					14
Pregătire seminar / laboratoare, teme, referate, portofolii și eseuri					28
Tutoriat					14
Examinări					4
Alte activități.....					
3.7 Total ore studiu individual	<b>74</b>				
3.8 Total ore pe semestru	<b>130</b>				
3.9 Numărul de credite	<b>5</b>				

**4. Precondiții (acolo unde este cazul)**

4.1 de curriculum	• Logica computațională
4.2 de competențe	• Utilizarea sistemelor de calcul

**5. Condiții (acolo unde este cazul)**

5.1. de desfășurare a cursului	• Sala de curs dotată cu tablă și retroproiector
5.2. de desfășurare a seminarului / laboratorului / proiectului	• Sala de laborator dotată cu sisteme de calcul corespunzătoare



## 6. Competențele specifice acumulate

Competențe profesionale	CUNOȘTINȚE: C1.1 Descrierea adecvată a paradigmelor de programare și a mecanismelor de limbaj specifice, precum și identificarea diferenței dintre aspectele de ordin semantic și sintactic.
	ABILITĂȚI: C1.3 Elaborarea codurilor sursă adecvate și testarea unitară a unor componente într-un limbaj de programare cunoscut, pe baza unor specificații de proiectare date. C1.4 Testarea unor aplicații pe baza unor planuri de test.
Competențe transversale	<ul style="list-style-type: none"> <li>• Dezvoltarea capacității de a se integra și de a lucra în cadrul unei echipe;</li> <li>• Dezvoltarea capacității organizatorice și a autonomiei în derularea de activități;</li> <li>• Selectarea resurselor informaționale, utilizarea eficientă a surselor de formare profesională, dezvoltarea capacității de corelare a activității profesionale la cerințele unei societăți dinamice.</li> </ul>

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

7.1 Obiectivul general al disciplinei	Cunoașterea arhitecturii calculatoarelor compatibile IBM și a seturilor de instrucțiuni și date de bază.
7.2 Obiectivele specifice	Cunoașterea limbajului de asamblare.

## 8. Conținuturi

8.1 Curs	Metode de predare	Observații
Structura generală a unui sistem de calcul. Arhitectura microprocesorului 8086: regiștri, calcul de adrese, reprezentarea instrucțiunilor mașină, moduri de adresare. Arhitectura microprocesorului 80386.	Expunere	4 ore
Elementele limbajului de asamblare: formatul unei linii sursă, expresii, directive		4 ore
Instrucțiuni ale limbajului de asamblare: manipularea datelor, operații, ramificări, salturi, cicluri, instrucțiuni pe șiruri.		8 ore
Înteruperi: generalități, sistemul de întreruperi 80286, instrucțiuni specifice lucrului cu întreruperi, utilizarea întreruperilor specifice 80286, formatele com și exe.		6 ore
Programarea multimodul: directive de segment simplificate, legarea modulelor asamblare cu module scrise în limbaje de nivel înalt.		2 ore
Programarea low-level în Pascal și C: inserarea de cod mașină în texte sursă Pascal și C, asamblare inline, proceduri și funcții imbricate, accesarea regiștrilor și apelarea de întreruperi, rutine de tratare a întreruperilor în Pascal și C.		4 ore
Bibliografie:		
<ol style="list-style-type: none"> <li>1. Boian F.M., Sisteme de programe pentru minicalculatoare, Lito. Univ. Cluj_Napoca, 1988</li> <li>2. Ceaparu M., Programmation el langage d'assemblage, Ed. Matrix Rom, București, 1998</li> <li>3. Knuth D.E., Tratat de programare a calculatoarelor, vol. I, II, III, Ed. Tehnică, București, 1976</li> <li>4. Lungu V., Procesoare Intel: programare în limbaj de asamblare, Ed. Teora, București, 2004</li> <li>5. Ivan L., Programarea în limbaj de asamblare: culegere de probleme, Ed. Infocrec, București, 1997</li> <li>6. Muscă Gh., Programarea în limbaj de asamblare, Ed. Teora, București, 1998</li> <li>7. Patrubby M., Totul despre microprocesorul Z80, Ed. Tehnica, București, 1989</li> </ol>		

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**CENTRUL UNIVERSITAR NORD DIN BAI A MARE**

8.3. Laborator	Metode de predare	Observații
Reprezentarea numerelor în memoria calculatorului : sistemele binar, octal, zecimal, hexazecimal ; conversii între reprezentări și operații aritmetice	Exemple, implementare aplicații	2 ore
Introducere în limbaj de asamblare : utilizarea programelor tasm, tlink, td ; structura minimală a unui program asm.		2 ore
Expresii aritmetice în limbaj de asamblare		2 ore
Aplicații cu operații pe biți		2 ore
Instrucțiuni condiționale în limbaj de asamblare. Aplicații.		2 ore
Operații asupra tablourilor. Aplicații.		2 ore
Operații asupra șirurilor de caractere. Aplicații.		4 ore
Utilizarea întreruperilor în programe asm.		2 ore
Implementarea operațiilor de intrare/ieșire în limbaj de asamblare.		4 ore
Programe în Pascal și C cu module scrise în asm.		4 ore
Verificare		2 ore
<b>Bibliografie:</b> 1. Boian F.M., Sisteme de programe pentru minicalculatoare, Lito. Univ. Cluj_Napoca, 1988 2. Ceaparu M., Programmation el langage d'assemblage, Ed. Matrix Rom, București, 1998 3. Knuth D.E., Tratat de programare a calculatoarelor, vol. I, II, III, Ed. Tehnică, București, 1976 4. Lungu V., Procesoare Intel : programare în limbaj de asamblare, Ed. Teora, București, 2004 5. Ivan L., Programarea în limbaj de asamblare : culegere de probleme, Ed. Infocrec, București, 1997 6. Muscă Gh., Programarea în limbaj de asamblare, Ed. Teora, București, 1998 7. Patrübany M., Totul despre microprocesorul Z80, Ed. Tehnica, București, 1989		

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

- Tematica acestui curs este în concordanță cu ceea ce este prevăzut în programul de studii la nivel licență al celor mai importante universități din țară și străinătate.

## 10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Cunoașterea principalelor aspecte teoretice prezentate la curs. Rezolvarea unor probleme similare celor explicate la curs și la orele de laborator.	Examen scris	60%
10.6 Laborator	Prezentarea tuturor aplicațiilor propuse la laborator, implementate în limbaj de asamblare.	Verificare	40%
10.8 Standard minim de performanță:			
<ul style="list-style-type: none"> <li>• Cunoașterea structurii unui sistem de calcul și a instrucțiunilor ASM</li> <li>• implementarea și documentarea unităților de program în limbaj de asamblare propuse la laborator</li> </ul>			

Data completării  
20.09.2016

Titular de curs  
Conf.univ.dr. Ioana ZELINA

Titular de laborator  
Asist.drd. Mara HAJDU MĂCELARU

Data avizării în Departament  
29.09.2016

Director Departament  
Prof.univ.dr. Vasile BERINDE