

FIȘA DISCIPLINEI

1. Date despre program

Instituția de învățământ superior	Universitatea Lucian Blaga din Sibiu
Facultatea	Facultatea de Inginerie
Departament	Departamentul de Calculatoare și Inginerie Electrică
Domeniul de studiu	Calculatoare si tehnologia informatiei
Ciclul de studii	Studii de licență
Specializarea	Tehnologia informatiei

2. Date despre disciplină

Denumirea disciplinei	Sisteme cu microprocesoare			
Codul cursului	Tipul cursului	An de studiu	Semestrul	Număr de credite
390456020612DO44	Obligatoriu	3	2	5
Tipul de evaluare	Categorია formativă a disciplinei (DF=fundamentală.; DD=domeniu; DS=specialitate; DC=complementară)			
Examen	DD			
Titular activități curs	Prof. univ. dr. ing. Lucian Vintan			
Titular activități seminar / laborator/ proiect	Prof. univ. dr. ing. Adrian Florea			

3. Timpul total estimat

Extinderea disciplinei în planul de învățământ – număr de ore pe săptămână				
Curs	Seminar	Laborator	Proiect	Total
3	-	2	-	5
Extinderea disciplinei în planul de învățământ – Total ore din planul de învățământ				
Curs	Seminar	Laborator	Proiect	Total ($NOAD_{sem}$)
42	-	28	-	70

Distribuția fondului de timp pentru studiu individual		Nr.ore
Studiul după manual, suport de curs, bibliografie și notițe		30
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren		4
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri		12
Tutoriat:		5
Examinări:		4
Total ore alocate studiului individual ($NOSI_{sem}$)		55
Total ore pe semestru ($NOAD_{sem} + NOSI_{sem}$)		125

4. Precondiții (acolo unde este cazul)

De curriculum	Cunoștințe referitoare la disciplinele de <i>Programare in limbaj de asamblare si Electronica digitala</i>
De competențe	Competențe de programare în limbajele C si asamblare

5. Condiții (acolo unde este cazul)

De desfășurare a cursului	Participare activă, lectura suportului de curs Tablă, videoproiector
De desfășurare a sem/lab/pr	Elaborarea și susținerea lucrărilor planificate Sală dotată cu calculatoare având instalate instrumentele necesare susținerii lucrărilor de laborator (v. continutul lucrărilor de laborator)

6. Competențe specifice acumulate

Competențe profesionale	<ul style="list-style-type: none"> • Operarea cu fundamente științifice, ingineresti și ale informaticii • Proiectarea componentelor hardware, software și de comunicații • Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor • Îmbunătățirea performanțelor sistemelor hardware, software și de comunicații • Proiectarea, gestionarea ciclului de viață, integrarea și integritatea sistemelor hardware, software și de comunicații
Competențe transversale	<ul style="list-style-type: none"> • Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesiei • Identificarea, descrierea și derularea proceselor din managementul proiectelor, cu preluarea diferitelor roluri în echipă și descrierea clară și concisă, verbal și în scris, în limba română și într-o limbă de circulație internațională, a rezultatelor din domeniul de activitate • Demonstrarea spiritului de inițiativă și acțiune pentru actualizarea cunoștințelor profesionale, economice și de cultură organizațională

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

Obiectivul general al disciplinei	Cunoașterea principiilor <i>hardware-software</i> care stau la baza funcționării sistemelor de calcul de tip mono și multi-procesor
Obiectivele specifice	<ul style="list-style-type: none"> • Însușirea principalelor concepte relative la paradigma microarhitecturilor de procesare a informației (instrucțiuni, date), de uz general cât și dedicate, abordate sub forma unui sistem interactiv și integrat la nivelul hardware (microarhitectura) – software de baza (compilator & SO) – aplicații. • Principiile de proiectare/optimizare iterativă a ansamblului microarhitectura-compilator-aplicații, bazate, în esență, pe modelarea și simularea acestora (<i>benchmarking</i>), dar și pe metode analitice de evaluare. • Abordarea microarhitecturilor de calcul, simultan, din punct de vedere formativ, informativ și aplicativ (dezvoltarea de aplicații practice)

8. Conținuturi

Curs		Nr. ore
Curs 1	<p>1. Introducere in problematica microarhitecturilor</p> <p>1.1 Structura a unui microsystem de calcul. Rolul modulelor componente si interactiunea acestora (microprocesor, memorii, interfete, periferice, <i>bus</i>-uri de interconectare, semnale de comenzi si stari, intreruperi etc.)</p> <p>1.2 Instructiuni, cicluri-faze, stari. Probleme ale comunicatiei microprocesor – memorii. Memorii DRAM, SRAM, ROM (EPROM, EEPROM), FLASH</p> <p>1.3 Moduri de transfer intre CPU si dispozitivele periferice (<i>polling</i>, intreruperi, DMA)</p> <p>1.4 Sistem de operare, compilator, link-editor, debugger</p> <p>1.5 Metrici de evaluare a performantelor</p> <p>1.6 Modalitati de evaluare a performantelor microsystemelor de calcul</p>	3
Curs 2	<p>2. Arhitectura microprocesorului de "uz general"</p> <p>2.1. Arhitectura setului de instructiuni (ISA – <i>Instruction Set Architecture</i>). Optimizarea ISA in vederea facilitarii compilarilor HLL (<i>High Level Languages</i>) si executiilor programelor obiect pe sistem. Exemple de compilare/executie apeluri proceduri. CISC vs. RISC. "Fuziunea" CISC & RISC. Exemple comerciale</p>	3
Curs 3	<p>2.2 Arhitectura unui microprocesor scalar generic</p> <p>2.2.1 Registrarii interni (clasificare, rol, utilizare software)</p> <p>2.2.2 Structura/Proiectarea caii de date a microprocesorului</p> <p>2.2.3 Structura/Proiectarea caii de control a microprocesorului</p>	3
Curs 4	<p>3. Proiectarea sistemului ierarhizat de memorii intr-o microarhitectura de calcul</p> <p>3.1 Necesitatea ierarhizarii sistemelor de memorii. Problema "<i>Memory-Wall</i>"</p> <p>3.2 Localitati (vecinatati) temporale si spatiale: caracteristici intrinseci ale programelor in executie.</p>	3
Curs 5	<p>3.3 Memorii <i>cache</i>. Functie si structura. Clasificari dupa gradele de asociativitate. Functionare dinamica. Principii de proiectare/implementare. Probleme de coerenta/consistenta si solutii. Compromisuri optime performanta/complexitate/cost: o schema adaptiva tip <i>Selective Victim Cache</i>. Performante</p> <p>3.4 Strategii de reducere a latentei memoriei principale</p>	3
Curs 6	<p>4. Procesoare <i>pipeline</i> scalare cu set optimizat de instructiuni</p> <p>4.1 Problemele hazardurilor (structurale, de date, de ramificatii, alias-uri de memorie). Solutii: vectori coliziune, optimizarea unitatilor secventiale de program prin <i>scheduling</i> static si predictia dinamica a ramificatiilor (<i>branches</i>), <i>memory disambiguation</i> (<i>anti-alias</i>)</p>	3
Curs 7	<p>4.2 Proiectarea <i>pipeline</i> a microprocesorului. Principii de proiectare a unitatii de comandă in vederea detectarii/ eliminarii hazardurilor (<i>control forwarding</i>).</p> <p>4.3 Probleme legate de evenimentele de exceptie in structurile <i>pipeline</i>. Solutii de principiu</p> <p>4.4 Analiza <i>anti-alias</i> a referirilor la memorie. Executia conditionata si speculativa a instructiunilor. Predicarea, <i>scheduling</i> static global (planificator-reorganizator).</p>	3
Curs 8	<p>5. Procesoare cu executii multiple ale instructiunilor (<i>Multiple Instruction</i>)</p>	3

	Issue – MII 5.1. Consideratii generale. Taxonomii (<i>scheduling</i> static, dinamic; dificultatea modelelor hibride) 5.2. Modele si algoritmi de procesare dinamica <i>out-of-order</i> a instructiunilor in microprocesoarele MII. <i>Buffer</i> -ul de reordonare: avantaje si dezavantaje	
Curs 9	5.3. Tehnici de optimizare statica a programelor. Optimizari locale (<i>List Scheduling</i>) si globale (<i>Trace Scheduling, Percolation</i>) 5.4. Tehnici de optimizare aferente buclelor de program (<i>Loop Unrolling, Software Pipelining</i>). Principiile compilarii iterativ-adaptive	3
Curs 10	5.5. <i>Scheduling</i> dinamic vs. <i>scheduling</i> static. Spre o integrare a conceptelor? Studii de caz: Microarhitecturile IA-64 (<i>Merced, Itanium</i>) si HSA (<i>Hatfield Superscalar Architecture</i>) 5.6. Spre o noua generatie arhitecturala de microprocesoare de uz general. Reutilizare dinamica a instructiunilor, predictie generalizata a instr., executii speculative, procesoare cu <i>Checkpoints (Selective Re-Issue & Reuse)</i> , <i>Kilo-Instruction-Processors</i> – o solutie la limitarile <i>buffer</i> -ului de reordonare)	3
Curs 11	6. Microarhitecturi “speciale” 6.1 Microcontrollere – caracteristici arhitecturale specifice 6.2 Microarhitecturi dedicate (<i>Embedded</i>) 6.2.1 Compilatoare si alte instrumente software in calculul dedicat 6.2.2 Optimizări statice ale programelor obiect in microarhitecturile dedicate	3
Curs 12	6.3 Elemente ale arhitecturii sistemelor multimicroprocesor 6.4 Sisteme <i>multi-core</i> si <i>many-core</i> . Arhitecturi, modele de programare, coerenta si consistenta, provocari majore, solutii 6.5 Exploatarea integrata a nivelurilor (gradelor) de paralelism	3
Curs 13	7. Instrumente software utile in analiza si proiectarea microarhitecturilor 7.1 Asamblare, <i>link</i> -editoare, <i>debuggere</i> . <i>Cross</i> -compilatoare. 7.2 Platforme de simulare monolitice respectiv modulare. Clasificare, caracteristici, utilitate (<i>Execution Driven</i> respectiv <i>Trace Driven</i>). Simularea la nivel tranzactional in sistemele <i>many-core</i>	3
Curs 14	7.3 Proiectare: Interfata cu utilizatorul si crearea resurselor. Nucleul functional al simulatorului. 7.4 <i>Benchmarking</i> (SPEC, EEMBC etc.). Optimizatoare de cod obiect (<i>schedulere</i> statice) 7.4 Exemple, aplicatii	3
Total ore curs:		42
Laborator		Nr. ore
Lab 1	Arhitectura microprocesoarelor MIPS R2000/R3000	2
Lab 2	Utilizarea simulatorului SPIM. Evidențierea conceptelor legate de cache-uri – modul de organizare, regulile de mapare, algoritmi de înlocuire a blocurilor conflictuale, strategia de scriere – folosind simulatorul PCSPIM-CACHE	2
Lab 3	Investigații arhitecturale utilizând simulatorul SPIM	2
Lab 4	VERIFICARE PE PARCURS	2
Lab 5	Arhitectura microprocesoarelor DLX	2
Lab 6	Utilizarea simulatorului grafic DLX/Utilizarea simulatorului VLIW-DLX	2
Lab 7	Investigații arhitecturale utilizând simulatorul DLX	2

Lab 8	Simularea/optimizarea interfeței procesor-cache pentru o arhitectură RISC superscalară parametrizabilă	2
Lab 9	VERIFICARE PE PARCURS	2
Lab 10	Optimizarea schemelor de predicție pentru ramificațiile de program în microprocesoarele superscalare (simulator)	2
Lab 11	Metode de reducere a "gap-urilor" tehnologice într-un sistem ierarhizat de memorii. Simulatorul grafic <i>Selective Victim Cache</i>	2
Lab 12	Procesarea <i>out-of-order</i> speculativă a instrucțiunilor. Simulatorul grafic SATSIM.	2
Lab 13	PSATSIM: instrument software de evaluare a complexității și a consumului de putere în microarhitecturile superscalare	2
Lab 14	Verificare finală a cunostintelor acumulate în orele de aplicații	2
Total ore laborator		28

Metode de predare

Prelegeri, problematizări, studii de caz, exerciții, conversații, explicații, demonstrații și dezbateri.	Limba de predare	Română
--	------------------	--------

Bibliografie

Referințe bibliografice recomandate	VINTAN LUCIAN.- <i>Organizarea și proiectarea microarhitecturilor. Note de curs</i> (pdf, 336 pagini A4; practic, cursul de Sisteme cu microprocesoare), URL: http://webspace.ulbsibiu.ro/lucian.vintan/html/Organizarea.pdf
	PATTERSON D., HENNESSY J. - <i>Computer Organization and Design, The Hardware/ Software Interface</i> , Morgan Kaufmann Publishers, 2 nd Edition, 1998 (traducere românească la Editura ALL, 2000)
	FLOREA ADRIAN, VINTAN N. LUCIAN – <i>Simularea și optimizarea arhitecturilor de calcul în aplicații practice</i> , Editura Matrix ROM, București, ISBN 973-685-605-4, 2003 (443 pg. + CD atasat), Bibl. Univ. Sibiu - cota 48.351 (4 ex. la Biblioteca ULBS + 4 ex. schimb interbibliotecar); comenzi la www.matrixrom.ro
	VINȚAN N. LUCIAN – <i>Fundamente ale arhitecturii microprocesoarelor</i> , Editura Matrix Rom, București, ISBN 978-606-25-0276-8, 2016 (547 pg.), v. http://www.matrixrom.ro/romanian/editura/domenii/cuprins.php?cuprins=FA50 ; 2 exemplare la Biblioteca ULBS, cota 04/V64 + 5 schimb inter-bibliotecar
Referințe bibliografice suplimentare	VINTAN N. LUCIAN – <i>Arhitecturi de procesoare cu paralelism la nivelul instrucțiunilor</i> , Editura Academiei Române, București, 2000 (264 pg.), ISBN 973-27-0734-8 – comanda la www.ea.ro , Bibl. ULBS, cota 45.351 (15 ex. la Biblioteca ULBS)
	HAYES J. – <i>Computer Architecture and Organization</i> , Third Edition, McGraw Hill, 1998

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

Se realizeaza prin contacte periodice cu acestia in vederea analizei problemei.

10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Ponderea în nota finală	Obs.*
Curs	Teste pe parcursul semestrului	Lucrare scrisă	10%	CPE
	Examen de semestru	Examen scris	60%	CEF
	Alte activități: prezenta la curs	-	5%	nCPE
Laborator	Activități aplicative	Evaluare orala aplicații realizate Fișă de evaluare seminar	15%	CPE
	Teme / referate		10%	nCPE
Standard minim de performanță				
50% rezultat după însumarea punctajelor ponderate conform coloanei 4				

(*) CPE – condiționează participarea la examen; nCPE – nu condiționează participarea la examen; CEF - condiționează evaluarea finală;

Data completării: 14.10.2017

Data avizării în Departament:.....

	Grad didactic, titlul, prenume, numele	Semnătura
Titular disciplină	Prof. dr. ing. Lucian VINTAN	
Director de departament	Prof. dr. ing. Daniel VOLOVICI	