

FIȘA DISCIPLINEI

Denumirea disciplinei :		Sisteme cu microprocesoare			
Codul disciplinei:		390453220659SO07			
Programul de studii:		Calculatoare si tehnologia informatiei			
Catedra:		Calculatoare si automatizari			
Facultatea:		Inginerie			
Universitatea:		"Lucian Blaga" din Sibiu			
Anul de studiu:	3	Semestrul	6	Tipul de evaluare finală	E
Regimul disciplinei (DI=obligatorie/ DO=opțională/DF=liber aleasă):			DI	Numărul de credite:	5
Categoriza formativă a disciplinei (DF=fundamentală.; DI=ingineresti; DS=specialitate; DC=complementară)					DS
Total ore din planul de învățământ				Total ore pe semestru:	
Titularul disciplinei: Prof. univ. dr. ing. Lucian VINTAN, m.c. ASTR					

Numărul total de ore (pe semestru) din planul de învățământ					
Total ore/ semestru	C	S	L	P	Total
	3		2		5

Obiective:	<ul style="list-style-type: none"> • Inusirea principalelor concepte relative la paradigma microarhitecturilor de procesare a informatiei (instrucțiuni, date), de uz general cât si dedicate, abordate sub forma unui sistem interactiv si integrat la nivelul hardware (microarhitectura)–software de baza (compilator&SO)– aplicații. • Principiile de proiectare/optimizare iterativă a ansamblului microarhitectura-compilator-aplicatii, bazate, in esenta, pe modelarea si simularea acestora (<i>benchmarking</i>), dar si pe metode analitice de evaluare. • Abordarea microarhitecturilor de calcul, simultan, din punct de vedere formativ, informativ si aplicativ (dezvoltarea de aplicatii practice)
Competențe specifice disciplinei	1. Cunoaștere și înțelegere: <ul style="list-style-type: none"> • Cunoasterea aprofundata a paradigmei microsystemelor de procesare a informatiei (instrucțiuni & control aferent, cale de date), de uz general cât si dedicate, abordate sub forma unui sistem interactiv si integrat la nivelurile hardware – software (compilator & SO)– aplicații.

	<p>2. Explicare și interpretare:</p> <ul style="list-style-type: none"> • Functionarea de ansamblu și respectiv la nivelul fiecărui modul component, a unui microsistem integrat de calcul, din punct de vedere al interacțiunilor hardware-software • Dezvoltarea unor deprinderi de analiză și sinteză a sistemelor-subsistemelor de calcul de uz general, respectiv a celor dedicate • Înțelegerea modului de procesare a programelor pe mașinile hardware și a influenței arhitecturii asupra performanțelor software-ului (sistem, aplicații)
	<p>3. Instrumental – aplicative:</p> <ul style="list-style-type: none"> • Dezvoltarea aptitudinilor practice de lucru cu microsisteme de calcul, simulatoare, compilatoare, debuggere, benchmark-uri etc., aferente unor microarhitecturi de calcul consacrate, în vederea proiectării-dezvoltării de aplicații hardware-software integrate
	<p>4. Atitudinale:</p> <ul style="list-style-type: none"> • capacitatea de utilizare a mijloacelor moderne de documentare și de simulare/evaluare a microarhitecturilor; • crearea unui limbaj tehnic adecvat analizelor și dezvoltărilor experimentale în domeniul microsistemelor; • capacitatea studenților de a lucra în echipe de câte 2-4 membri, în vederea efectuării unor lucrări/proiecte de laborator.

Conținutul tematic (descriptori)	TEMATICA CURSURILOR - http://webspaces.ulbsibiu.ro/lucian.vintan/index.html#11		
	Nr. crt.	Denumirea temei	Nr. ore
	0	<p>1. Introducere în problematica microarhitecturilor</p> <p>1.1 Structura a unui microsistem de calcul. Rolul modulelor componente și interacțiunea acestora (microprocesor, memorii, interfețe, periferice, bus-uri de interconectare, semnale de comenzi și stări, întreruperi etc.)</p> <p>1.2 Instrucțiuni, cicluri-faze, stări. Probleme ale comunicății microprocesor – memorii. Memorii DRAM, SRAM, ROM (EPROM, EEPROM), FLASH</p> <p>1.3 Moduri de transfer între CPU și dispozitivele periferice (<i>polling</i>, întreruperi, DMA)</p> <p>1.4 Sistem de operare, compilator, link-editor, debugger</p> <p>1.5 Metrici de evaluare a performanțelor.</p> <p>1.6 Modalități de evaluare a performanțelor microsistemelor de calcul</p>	3 ore

	<p>1</p> <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 (recursive) proceduri. CISC vs. RISC. “Fuziunea” CISC & RISC. Exemple comerciale</p> <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>	5 ore
	<p>2</p> <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. Vecinatatea valorilor instructiunilor – un concept novator si fertil</p> <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>	6 ore
	<p>3</p> <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 (anti-alias)</i></p> <p>4.2 Proiectarea <i>pipeline</i> a microprocesorului. Principii de proiectare a unitatii de comanda 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>	7 ore

	<p>4 5. Procesoare cu executii multiple ale instructiunilor (<i>Multiple Instruction Issue – MII</i>)</p> <p>5.1. Consideratii generale. Taxonomii (<i>scheduling</i> static, dinamic; dificultatea modelelor hibride)</p> <p>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</p> <p>5.3. Tehnici de optimizare statica a programelor. Optimizari locale (<i>List Scheduling</i>) si globale (<i>Trace Scheduling, Percolation</i>)</p> <p>5.4. Tehnici de optimizare aferente buclelor de program (<i>Loop Unrolling, Software Pipelining</i>). Principiile compilarii iterativ-adaptive</p> <p>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>)</p> <p>5.6. Spre o noua generatie arhitecturala de microprocesoare de uz general. Posibile caracteristici arhitecturale: reutilizare dinamica a instructiunilor, predictie generalizata a instr., executii speculative, procesoare cu <i>Checkpoints (Selective Re-Issue & Reuse)</i>, <i>Kilo-Instruction-Processors – o solutie la limitarile buffer-ului de reordonare etc.</i>).</p>	9 ore
	<p>5 6. Microarhitecturi “speciale”</p> <p>6.1 Microcontrollere – caracteristici arhitecturale specifice</p> <p>6.2 Microarhitecturi dedicate (<i>Embedded</i>)</p> <p>6.2.1 Compilatoare si alte instrumente software in calculul dedicat</p> <p>6.2.2 Optimizări statice ale programelor obiect in microarhitecturile dedicate</p> <p>6.3 Elemente ale arhitecturii sistemelor multimicroprocesor</p> <p>6.4 Sisteme <i>multi-core</i> si <i>many-core</i>. Arhitecturi, modele de programare, coerenta si consistenta, provocari majore, solutii</p> <p>6.5 Exploatarea integrata a nivelurilor (gradelor) de paralelism</p>	6 ore

	6	<p>7. Instrumente software utile in analiza si proiectarea microarhitecturilor</p> <p>7.1 Asamblare, <i>link</i>-editoare, <i>debuggere</i>. <i>Cross-compiler</i>.</p> <p>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></p> <p>7.3 Proiectare: Interfata cu utilizatorul si crearea resurselor. Nucleul functional al simulatorului.</p> <p>7.4 <i>Benchmarking</i> (SPEC, EEMBC etc.). Optimizatoare de cod obiect (<i>schedulere</i> statice)</p> <p>7.4 Exemple, aplicatii</p>	6 ore
	TEMATICA SEMINARIILOR/LABORATOARELOR/PROIECTULUI		
	1.	Arhitectura microprocesoarelor MIPS R2000/R3000	2
	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
	3.	Investigații arhitecturale utilizând simulatorul SPIM	2
	4.	Arhitectura microprocesoarelor DLX	2
	5.	Utilizarea simulatorului grafic DLX/Utilizarea simulatorului VLIW-DLX	2
	6.	Investigații arhitecturale utilizând simulatorul DLX	2
	7.	Simularea/optimizarea interfeței procesor-cache pentru o arhitectură RISC superscalară parametrizabilă	2
	8.	Optimizarea schemelor de predicție pentru ramificațiile de program în procesoarele superscalare avansate (simulator)	2
	9.	Metode de reducere a “ <i>gap</i> -urilor” tehnologice într-un sistem ierarhizat de memorii. Simulatorul grafic <i>Selective Victim Cache</i>	4
	10.	Procesarea <i>out-of-order</i> speculativă a instrucțiunilor. Simulatorul grafic SATSIM. PSATSIM: instrument software de evaluare a complexității și a consumului de putere în microarhitecturile superscalare	6
11.	Verificare finală a cunostintelor acumulate in orele de aplicatii	2	

Metode de predare / seminarizare	Expunerea (clasica -deductiva, inductiva si formalizata; expuneri PPT etc.), conversația euristică, problematizare, studii de caz, prelegere intensificată, teme de casa. Se utilizeaza inclusiv si prediect, vechile principii ale educatiei paideice, in virtutea carora, studenti si lector deopotriva, dau si primesc cunostinte.
----------------------------------	---

Stabilirea notei finale (procentaje)	- răspunsurile la examen/colocviu(evaluare finală)	60%
	- teste pe parcursul semestrului	20%
	- răspunsurile finale la lucrările practice de laborator	10%
	- activități gen teme/referate/eseuri/traduceri/proiecte etc.	

	- teme de control	10%
	- alte activități(<i>precizați</i>).....	
	- TOTAL	100%

Descrieți modalitatea practică de evaluare finală, E/V (de exemplu: lucrare scrisă (descriptive și/sau test grilă și/sau probleme etc.), examinare orală cu bilete, colocviu individual ori în grup, proiect etc)

Evaluarea finală va cuprinde rezolvarea a 5-6 probleme cu un pronunțat caracter aplicativ (asigurand inclusiv verificarea stapanirii conceptelor teoretice esentiale), fiecare cuprinzand 2-4 sub-probleme. Subiectele vor acoperi intreaga problematica cuprinsa in programa analitica. Aceste probleme totalizeaza 100 de puncte, repartizate judicios, functie de dificultatea specifica a fiecărei sub-probleme.

Cerințe minime pentru nota 5

- Nota 5 la activități aplicative la laborator, temelor si testelor pe parcurs;
- Nota 5 la examenul propriu-zis (adica minim 50 puncte din cele 100 puncte aferente problemelor examenului final)

Cerințe pentru nota 10

- punctaj maxim pentru toate activitatile din timpul semestrului;
- peste 94 puncte la examenul final.

Pentru rezultate deosebite în activitatea de cercetare se acordă bonificații de până la 2 puncte la nota finală (conform regulamentului de evaluare al Facultății de inginerie).

TOTAL ore studiu individual (pe semestru) = 84

Bibliografia	<p>Minimală obligatorie:</p> <ul style="list-style-type: none"> • VINTAN LUCIAN.- <i>Organizarea si proiectarea microarhitecturilor. Note de curs</i> (pdf, 318 pagini A4; practic, cursul de Sisteme cu microprocesoare), URL: http://webspaces.ulbsibiu.ro/lucian.vintan <p>Complementară:</p> <ul style="list-style-type: none"> • VINTAN N. LUCIAN – <i>Arhitecturi de procesoare cu paralelism la nivelul instructiunilor</i>, Editura Academiei Romane, Bucuresti, 2000 (264 pg.), ISBN 973-27-0734-8 – comanda la www.ear.ro, Bibl. ULBS, cota 45.351 (15 ex. la Biblioteca ULBS) • VINTAN LUCIAN, FLOREA ADRIAN - <i>Sisteme cu microprocesoare - aplicatii</i>, Editura Universitatii "L. Blaga" din Sibiu, ISBN 973-9410-46-4 , Sibiu, 1999 (245 pg.) Bibl. Univ. Sibiu, cota 43.800 (15 ex. intern + 15 ex. schimb interbibliotecar) • VINTAN N. LUCIAN, FLOREA ADRIAN – <i>Microarhitecturi de procesare a informatiei</i>, Editura Tehnica, Bucuresti, ISBN 973-31-1551-7, cota bibl. ULBS 45.797 (16 schimb+14 intern bibl.ULBS), 2000 (312 pg) • FLOREA ADRIAN, VINTAN N. LUCIAN – <i>Simularea si optimizarea arhitecturilor de calcul in aplicatii practice</i>, Editura Matrix ROM, Bucuresti, 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 • VINTAN N. LUCIAN – <i>Prediction Techniques in Advanced Computing Architectures</i> (in limba engleza), Matrix Rom Publishing House, Bucharest, ISBN 978-973-755-137-5, 2007 (292 pg.; 3 ex. ULBS + 7 schimb interbibliotecar; cota Biblioteca ULBS 52.103); http://www.matrixrom.ro/romanian/editura/domenii/informatica.php?id=867#867 ; in format electronic PDF la : http://webspaces.ulbsibiu.ro/lucian.vintan • VINTAN LUCIAN.- <i>Generația următoare</i>, în revista PC-Report, nr.3, Editura Computer Press Agora, ISSN 1220-9856, martie, 2000 • VINTAN LUCIAN.- <i>Procesorul IA-64: între evoluție și revoluție</i>, în revista PC-Report, nr.5, Editura Computer Press Agora, ISSN 1220-9856, mai, 2000 • HENNESSY J., PATTERSON D. - <i>Computer Architecture: A Quantitative Approach</i>, Morgan Kaufmann (Elsevier), 4-th Edition, 2007 • PATTERSON D., HENNESSY J. - <i>Computer Organization and Design, The Hardware/ Software Interface</i>, Morgan Kaufmann Publishers, 2nd Edition, 1998 (traducere romaneasca la Editura ALL, 2000) • FISHER J., FARABOSCHI P., YOUNG C. – <i>Embedded Computing</i>, Morgan Kaufmann Publishers (Elsevier), 2005 • HAYES J. – <i>Computer Architecture and Organization</i>, Third Edition, McGraw Hill, 1998
---------------------	--

Lista materialelor didactice utilizate în procesul de predare:

Tabla și creta; videoproiector și laptop, prezentari PPT ale cursului; curs și îndrumar de aplicații, publicate, existente la Biblioteca ULBS dar și în format electronic pe pagina de web a titularului de curs - <http://webspace.ulbsibiu.ro/lucian.vintan/>.

Rețea de calculatoare, Sisteme operare: Linux, Windows, Instrumente soft: Visual C++ v.6.0, set utilitare GCC, set simulatoare SimpleScalar v.3.0 și M-SIM (SMT), simulatoare LC-2, SPIM, DLX, SATSim, set simulatoare complexe pentru optimizarea microarhitecturilor avansate (dezvoltate local), simulator CACTI și Watch (consum putere), benchmark-uri SPEC 2000, benchmark-uri Stanford-HSA. Frame-ul de cercetare dezvoltare UniSim pentru arhitecturi multi și *many-cores*.

	Grad didactic, titlul, prenume, numele	Semnătura
Coordonator de disciplină	Prof.univ.dr.ing. Lucian VINTAN Membru (c.) al Acad. de Științe Tehnice din România	

22.09.2010

ȘEF CATEDRĂ

Conf. dr. ing. Ioan Z. MIHU