

## FIȘA DISCIPLINEI<sup>1</sup>

### 1. Date despre program

1.1 Instituția de învățământ superior	Universitatea „Politehnica” din Timișoara
1.2 Facultatea <sup>2</sup> / Departamentul <sup>3</sup>	Automatică și Calculatoare / Calculatoare
1.3 Catedra	-
1.4 Domeniul de studii	Calculatoare și Tehnologia informației
1.5 Ciclul de studii	Licență
1.6 Programul de studii / Calificarea	Calculatoare / inginer

### 2. Date despre disciplină

2.1 Denumirea disciplinei	<b>Fundamente de ingineria calculatoarelor</b>						
2.2 Titularul activităților de curs	Prof.dr.ing. Mircea VLĂDUȚIU						
2.3 Titularul activităților de seminar	Drd.ing. Alexandru IOVANOVICI						
2.4 Anul de studiu	3	2.5 Semestrul	1	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei	Obligatorie

### 3. Timpul total estimat (ore pe semestru al activităților didactice)

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	94	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					18
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					10
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					10
Tutoriat					4
Examinări					6
Alte activități					
<b>3.7 Total ore studiu individual</b>	38				
<b>3.8 Total ore pe semestru</b>	104				
<b>3.9 Numărul de credite</b>	4				

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

4.1 de curriculum	<ul style="list-style-type: none"> <li>Arhitectura calculatoarelor, Organizarea calculatoarelor</li> </ul>
4.2 de competențe	<ul style="list-style-type: none"> <li>Cunoștințe elementare de design digital</li> <li>Cunoștințe elementare de matematică</li> </ul>

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

5.1 de desfășurare a cursului	<ul style="list-style-type: none"> <li>Sală de mărime mare, materiale suport: laptop, proiector, tablă</li> </ul>
5.2 de desfășurare a seminarului/laboratorului	<ul style="list-style-type: none"> <li>Laborator cu 15 calculatoare, 10 platforme reconfigurabile diferite (Altera, Xilinx), medii de dezvoltare Quartus și WebPack, tablă</li> </ul>

### 6. Competențe specifice acumulate

Competențe profesionale <sup>4</sup>	<ul style="list-style-type: none"> <li>Operarea cu fundamente științifice, ingineresti și ale informaticii</li> <li>Proiectarea componentelor hardware, software și de comunicații</li> <li>Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor</li> </ul>
--------------------------------------	---

<sup>1</sup> Formularul corespunde Fișei Disciplinei promovată prin OMECTS 5703/18.12.2011 (Anexa3);

<sup>2</sup> Se înscrie numele facultății care gestionează programul de studiu căruia îi aparține disciplina;

<sup>3</sup> Se înscrie numele departamentului căruia i-a fost încredințată susținerea disciplinei și de care aparține titularul cursului;

<sup>4</sup> Aspectul competențelor profesionale va fi tratat cf. Metodologiei OMECTS 5703/18.12.2011. Se vor prelua competențele care sunt precizate în Registrul Național al Calificărilor din Învățământul Superior RNCIS ([http://www.rncis.ro/portal/page?\\_pageid=117,70218&\\_dad=portal&\\_schema=PORTAL](http://www.rncis.ro/portal/page?_pageid=117,70218&_dad=portal&_schema=PORTAL)) pentru domeniul de studiu de la pct. 1.4, programul de studii de la pct. 1.6 din această fișă și materia în cauză

	<ul style="list-style-type: none"> <li>• Îmbunătățirea performanțelor sistemelor hardware, software și de comunicații</li> <li>• Proiectarea, gestionarea ciclului de viață, integrarea și integritatea sistemelor hardware, software și de comunicații</li> </ul>
Competențe transversale	<ul style="list-style-type: none"> <li>• Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura rezolvarea problemei</li> <li>• Demonstarea spiritului de inițiativă și acțiune pentru actualizarea cunoștințelor profesionale, economice și de cultură organizațională</li> <li>• 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</li> </ul>

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

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> <li>• Dobândirea noțiunilor de bază referitoare la sinergia procesor-ierarhie de memorii și a modalităților de estimare a performanței sistemelor de calcul</li> <li>• Stimularea capacității de analiză și evaluare a metodelor prin care performanța unui sistem de calcul poate fi optimizată în scopul creșterii vitezei de execuție sau a micșorării consumului de energie</li> </ul>
7.2 Obiectivele specifice	<ul style="list-style-type: none"> <li>• Construirea unei imagini de ansamblu asupra modului de operare a sistemelor de calcul, cu accent pe procesor și ierarhia de memorii</li> <li>• Consolidarea bazei teoretice însușite de studenți la disciplinele anterioare și dezvoltarea abilităților de operare cu concepte proprii arhitecturii și organizării calculatoarelor</li> <li>• Prezentarea caracteristicilor unui pipeline și dezvoltarea abilităților de analiză a performanțelor și de proiectare a acestuia în scopul optimizării funcționării</li> <li>• Prezentarea caracteristicilor unei memorii cache și dezvoltarea abilităților de analiză a performanțelor și de proiectare a acesteia în scopul optimizării funcționării</li> <li>• Instruirea practica a studenților asupra caracteristicilor sistemelor de memorie virtuală</li> </ul>

## 8. Conținuturi

8.1 Curs	Număr de ore	Metode de predare
<b>1. Metrici de evaluare a performanței în sistemele de calcul</b> 1.1. Tendințe în tehnologia și utilizarea sistemelor de calcul 1.2. Măsurarea performanței. Programe benchmark 1.3. Principii cantitative în designul sistemelor de calcul 1.4. Performanța procesorului și componentele sale 1.5. Ierarhii de memorii și implicația acestora în performanță	6	Prelegere interactivă susținută de prezentări PPT, conversația, dezbaterile, studiul de caz, brainstorming, analiza SWOT
<b>2. Controlul pipeline în sistemele de calcul</b> 2.1. Prezentare generală 2.2. O arhitectură DLX cu pipeline 2.3. Problemele fundamentale: hazardurile 2.4. Hazardul structural 2.5. Hazardul de date 2.6. Hazardul de control 2.7. Excepții în DLX 2.8. Extinderea pipeline-ului pentru operații multiciclu 2.9. Probleme de design privind setul de instrucții 2.10. Pipeline-ul MIPS R4000 2.11. Concluzii	10	
<b>3. Memorii cache și ierarhii de memorii</b> 3.1. Memorii cache 3.1.1. Replasarea blocurilor și probleme specifice. 3.1.2. Structura cache la Alpha AXP 21064 3.1.3. Criterii de performanță. Ameliorarea performanței cache-urilor 3.2. Reducerea cache misses 3.2.1. Mărirea dimensiunii blocurilor VS mărirea asociativității 3.2.2. Victim cache și cache set-asociativ 3.2.3. Hardware pre-fetching pentru instrucții și date 3.2.4. Optimizări de compilator 3.3. Reducerea cache miss penalty 3.4. Reducerea hit time 3.5. Memoria principală. Tehnologii și organizare 3.6. Memoria virtuală. Replasarea blocurilor de memorie și translatarea adreselor 3.7. Protecția în memoria virtuală 3.8. Ierarhia de memorie la Alpha AXP 21064	12	

Bibliografie 1. D.A. Patterson, J.L. Hennessy. Computer Architecture. A Quantitative Approach. Morgan Kaufmann Publishers Inc., 2011 2. R.E. Bryant, D.R. O'Hallaron. Computer Systems: A Programmer's Perspective. Addison-Wesley, 2010 2. W. Stallings. Computer Organization and Architecture. Designing for Performance. Prentice Hall, 2012 3. D. Tabak. Advanced Microprocessors. McGraw-Hill, Inc, 1995 4. M. Vlăduțiu. Arhitectura și organizarea calculatoarelor. Vol I. Editura Politehnica, 2011		
8.2 Seminar/laborator	Număr de ore	Metode de predare
1. Analiza alocării resurselor hardware în timpul execuției unei secvențe de cod. Identificarea situațiilor de dependență.	2	Expunere temă, prelegerea interactivă, abordare euristică, dezbatere, controversă creativă, problematizare, algoritmizare, brainstorming.
2. Analiza pipeline-ului DLX pentru operații cu întregi și identificarea hazardurilor.	2	
3. Analiza încărcării pipeline-ului pentru o secvență de cod cu operații în virgulă flotantă.	2	
4. Desfășurarea buclilor de cod. Identificarea și eliminarea ciclurilor de stall datorate hazardurilor. Analiza dependențelor existente într-o secvență de cod. Listarea problemelor apărute și posibilități de rezolvare hardware-software	4	
5. Analiza unei secvențe de cod prin metodele scoreboard și Tomasulo	4	
6. Desfășurarea de bucle pentru procesorul DLX în varianta superscalară. Identificarea situațiilor de dependență	2	
7. Analiza și simularea modului de funcționare a algoritmilor de replasare pentru memorii cache	2	
8. Analiza și evaluarea parametrilor cache pentru o secvență de cod, într-o ierarhie de memorie cu caracteristici impuse	2	
9. Analiza unor scenarii de execuție pentru o secvență de cod rulată pe mașini cu ierarhii de memorie de caracteristici diferite	2	
10. Analiza mecanismului de translatare a adreselor și a problemelor de replasare într-o ierarhie complexă de memorie virtuală	2	
11. Analiza tehnicilor VLIW	2	
12. Recapitulare. Încheierea activității.	2	
Bibliografie 1. D.A. Patterson, J.L. Hennessy. Computer Architecture. A Quantitative Approach. Morgan Kaufmann Publishers Inc., 2011 2. R.E. Bryant, D.R. O'Hallaron. Computer Systems: A Programmer's Perspective. Addison-Wesley, 2010 3. M. Vlăduțiu. Arhitectura și organizarea calculatoarelor. Vol I. Editura Politehnica, 2011		

### 9. Corelarea conținutului disciplinei cu cerințele specialiștilor din domeniu și cu așteptările angajatorilor reprezentativi

<ul style="list-style-type: none"> <li>Cunoștințele legate de optimizarea execuției instrucțiilor într-un sistem de calcul reprezintă o parte fundamentală din pregătirea teoretică și practică a unui absolvent de studii superioare în domeniul IT</li> <li>Totalitatea angajatorilor din domeniul IT (atât software, cât și hardware) este direct interesată de optimizarea, în general, a funcționării sistemelor de calcul. Disciplina răspunde direct acestui scop conferind cunoștințe care au un impact imediat asupra performanței în funcționarea sistemelor de calcul. De asemenea, subiectele tratate contribuie la luarea unor decizii potrivite în design-ul de ansamblu al oricărui sistem de calcul.</li> </ul>
---

### 10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Prezentarea unor subiecte impuse sub formă de sinteză	Examinare scrisă	40 %
	Rezolvarea problemelor propuse	Examinare scrisă	10 %
	Gradul de aprofundare a subiectelor	Examinare scrisă	10 %
10.5 Seminar /laborator	Rezolvarea problemelor propuse	Prezentarea rezolvărilor, răspunsuri la întrebări. Demonstrarea rezolvării prin rularea pe platforma hardware	20 %
	Implementarea lucrărilor corespunzătoare de laborator	Examinare orală și pe platforma hardware	10 %
	Prezența	Evidența prezenței	10 %
10.6 Standard minim de performanță (volumul de cunoștințe minim necesar pentru promovarea disciplinei și modul în care se verifică stăpânirea lui)			
<ul style="list-style-type: none"> <li>Cunoștințe de bază legate de principalele aspecte care intervin în funcționarea unui pipeline și a unei memorii cache</li> </ul>			

- Estimarea indicatorilor de performanță pentru un pipeline și o memorie cache
- Subiectele de examen trebuie să totalizeze cel puțin 5 puncte

#### 11. Compatibilitate internațională

- University of California at Berkeley, Computer Science 252 – Graduate Computer Architecture
- Swansea University (UK), High Performance Microprocessors
- MIT, 6.033 – Computer System Engineering SMA 5501

Data completării

Semnătura titularului de curs

Semnătura titularilor de seminar

Prof.dr.ing. Mircea VLĂDUȚIU

Drd.ing. Alexandru IOVANOVICI

.....

.....

Data avizării în departament

Semnătura directorului de departament

Prof. dr. Ing. Vladimir Ioan CREȚU

.....