

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Babeș-Bolyai Cluj-Napoca
1.2 Facultatea	Facultatea de Matematica și Informatică
1.3 Departamentul	Departamentul de Informatică
1.4 Domeniul de studii	Informatică
1.5 Ciclul de studii	Licenta
1.6 Programul de studiu / Calificarea	Informatică - germană

2. Date despre disciplină

2.1 Denumirea disciplinei	Arhitectura sistemelor de calcul						
2.2 Titularul activităților de curs	Gaceanu Radu						
2.3 Titularul activităților de seminar	Gaceanu Radu						
2.4 Anul de studiu	1	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	1 sem + 2 lab
3.4 Total ore din planul de învățământ	70	Din care: 3.5 curs	28	3.6 seminar/laborator	42
Distribuția fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie și notițe					20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					10
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					20
Tutoriat					10
Examinări					20
Alte activități:					
3.7 Total ore studiu individual	80				
3.8 Total ore pe semestru	150				
3.9 Numărul de credite	6				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	•
4.2 de competențe	•

5. Condiții (acolo unde este cazul)

5.1 De desfășurare a cursului	• Sală de curs, proiector, laptop
5.2 De desfășurare a	• Laborator cu calculatoare

seminarului/laboratorului	
---------------------------	--

6. Competențele specifice acumulate

Competențe profesionale	<p>C6.1 Identificarea conceptelor și modelelor de baza pentru sisteme de calcul și rețele de calculatoare.</p> <p>C6.2 Identificarea și explicarea arhitecturilor de bază pentru organizarea și gestiunea sistemelor și a rețelelor.</p>
Competențe transversale	<p>CT1 Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională</p> <p>CT3 Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare în limba română și într-o limbă de circulație internațională</p>

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> Cunoașterea modelelor arhitecturale ale calculatoarelor, funcționarea procesorului, utilizarea sistemelor de reprezentare a informației în calculator.
7.2 Obiectivele specifice	<ul style="list-style-type: none"> Însușirea de către studenți a modelelor arhitecturale ale calculatoarelor, funcționarea procesorului, a utilizării sistemelor de reprezentare a informației în calculator. Inițiere în programarea în limbaj de asamblare, ceea ce asigură înțelegerea arhitecturii și funcționării unui microprocesor. Inițiere în arhitecturile sistemelor de întreruperi, cu particularizarea la mașinile 80x86. Conștientizarea influenței pe care principiile funcționale de bază ale arhitecturii von Neumann le au asupra modului de implementare a limbajelor de programare de nivel înalt; Conștientizarea impactului arhitectural asupra tehnicilor de proiectare și implementare a limbajelor de programare de nivel înalt.

8. Conținuturi

8.1 Curs	Metode de predare	Observații
1. Reprezentarea datelor: date elementare, reprezentări binare și ordini de plasare, organizarea și memorarea datelor	Expunerea, studiul de caz, exemple, demonstrații	
2. Reprezentarea datelor: codificarea caracterelor,	Expunerea, studiul de caz,	

reprezentarea cu semn si fara semn, cod complementar, conversii, conceptul de depasire	exemple, demonstratii	
3. Arhitectura sistemelor de calcul: principiile von Neumann, organizarea unui SC, unitatea centrala, ceasul sistem, calculator pe n biti, memoria, dispozitivele periferice, performantele unui SC, (S4); (seminar saptamanile S3/S4);	Expunerea, studiul de caz, exemple, demonstratii	
4. Arhitectura sistemelor de calcul: arhitectura microprocesorului 80x86 – structura, registri, calculul de adresa, moduri de adresare, adrese far si near	Expunerea, studiul de caz, exemple, demonstratii	
5. Elementele limbajului de asamblare: formatul unei linii sursa, expresii, tipuri de accesare a operanzilor, operatori	Expunerea, studiul de caz, exemple, demonstratii	
6. Elementele limbajului de asamblare: interfața DOS , directive pentru definirea segmentelor, pentru definirea datelor, LABEL, EQU, PROC, INCLUDE, blocuri repetitive si macrouri	Expunerea, studiul de caz, exemple, demonstratii	
7. Instructiuni ale limbajului de asamblare: instructiuni de transfer, conversii, operatii aritmetice cu semn si fara semn, operatii de deplasare si rotire de biti, operatii logice pe biti	Expunerea, studiul de caz, exemple, demonstratii	
8. Instructiuni ale limbajului de asamblare: instructiuni de salt conditionat si neconditionat, instructiuni de ciclare, instructiuni pe siruri	Expunerea, studiul de caz, exemple, demonstratii	
9. Întreruperi: clasificare, instructiuni specifice lucrului cu întreruperi, formatele COM si EXE	Expunerea, studiul de caz, exemple, demonstratii	
10. Redirectarea întreruperilor: programe TSR, instalarea si dezinstalarea programelor TSR, depanarea programelor TSR, redirectarea întreruperilor în cadrul SO Windows	Expunerea, studiul de caz, exemple, demonstratii	
11. Implementarea apelului de subprograme si programare multimodul: cod de apel, cod de intrare, cod de iesire, directivele PUBLIC, EXTRN, GLOBAL, legarea de module TASM cu module scrise în limbaje de nivel înalt	Expunerea, studiul de caz, exemple, demonstratii	
12. Programare low-level în limbaje de nivel înalt: inserare de cod masina, asamblare inline, proceduri si functii assembler, accesarea registrilor si apelarea de întreruperi, proceduri si functii interrupt	Expunerea, studiul de caz, exemple, demonstratii	
13. Extensii x86: modul de lucru protejat, extensii arhitecturale si noi instructiuni introduse la nivelul evolutiei familiei de procesoare 80x86	Expunerea, studiul de caz, exemple, demonstratii	
14. Programarea în limbaj de asamblare sub Windows: apeluri sistem în modul de lucru protejat, limitari ale sistemului de întreruperi, asamblarele MASM si NASM, asamblorul inline Visual C++	Expunerea, studiul de caz, exemple, demonstratii	

Bibliografie:		
<ol style="list-style-type: none"> 1. Axel Bottcher, Rechneraufbau und Rechnerarchitektur, Springer Verlag, Berlin, Heidelberg, New York, 2006. 2. Oberschelb Walter, Vossen Gottfried, Rechneraufbau und Rechnerstrukturen, Oldenburg, Muenchen, Wien, 2000. 3. R. Richter, P. Sander, W. Stucky, Der Rechner als System, Organisation, Daten, Programme, Teubner, Stuttgart, 1997. 4. Christian Siemers, Rechnerarchitektur I/II, http://www.in.tu-clausthal.de/uploads/media/Rechnerarchitektur_Skript_02.pdf 5. P. Marwedel, Rechnerarchitektur, https://eldorado.tu-dortmund.de/bitstream/2003/20362/1/rechner.pdf 6. Ch. Märtin, Rechnerarchitekturen : CPUs, Systeme, Software - Schnittstellen, Fachbuchverlag, Leipzig, 2002 7. Al. Vancea, F. Boian, D. Bufnea, A. Gog, A. Darabant, A. Sabau – Arhitectura calculatoarelor. Limbajul de asamblare 80x86., Editura Risoprint, Cluj-Napoca, 2005. 8. A. Gog, A. Sabau, D. Bufnea, A. Sterca, A. Darabant, Al. Vancea – Programarea în limbaj de asamblare 80x86. Exemple si aplicatii., Editura Risoprint, Cluj-Napoca, 2005. 9. Randal Hyde – The Art of Assembly Programming, No Starch Press, 2003. (http://homepage.mac.com/randyhyde/webster.cs.ucr.edu/www.artofasm.com/DOS/index.html) 10. Boian F. M. Sisteme de operare interactive. Ed. Libris, Cluj, 1994 11. Boian F. M. De la aritmetica la calculatoare. Ed. Presa Universitara Clujeana, Cluj, 1996 12. Boian F.M. Vancea A. Arhitectura calculatoarelor, suport de curs. Facultatea de Matematica si Informatica, Centrul de Formare Continua si Invatamânt la Distanta., Ed. Centrului de Formare Continua si Invatamânt la Distanta, Cluj, 2002, 13. Knuth D.E. Tratat de programarea calculatoarelor; vol 3: Algoritmi seminumerici. Ed. Tehnica, Bucuresti, 1985 		
8.2 Seminar	Metode de predare	Observații
1. Reprezentarea datelor si a informatiei	Dialogul, dezbaterea, studiul de caz, exemple	<i>Seminarul are loc ca activitate de două ore, din două în două săptămâni.</i>
2. Arhitectura sistemelor de calcul: principiile von Neumann	Dialogul, dezbaterea, studiul de caz, exemple	
3. Procesorul 80x86	Dialogul, dezbaterea, studiul de caz, exemple	
4. Programarea in limbaj de asamblare	exemple, proiecte practice	
5. Programarea in limbaj de asamblare	exemple, proiecte practice	
6. Programare low-level în limbaje de nivel înalt	exemple, proiecte practice	
7. Programarea în limbaj de asamblare sub Windows	exemple, proiecte practice	
Laborator	Metode de predare	Observații
1. Reprezentarea datelor, operatii in diferite baze de numerație	Dialogul, dezbaterea, studiul de caz, exemple	
2. Reprezentarea informatiei	Dialogul, dezbaterea, studiul de caz, exemple	
3. Caracteristici si componente ale arhitecturii	Dialogul, dezbaterea, studiul de caz, exemple	

sistemelor de calcul Von Neumann		
4. Arhitectura sistemelor de calcul Von Neumann - procesorul 80x86	Dialogul, dezbateră, studiul de caz, exemple	
5. Introducere în programarea în limbaj de asamblare	Dialogul, dezbateră, studiul de caz, exemple	
6. Instrucțiuni de bază în programarea în limbaj de asamblare	Dialogul, dezbateră, studiul de caz, exemple	
7. Programarea în limbaj de asamblare	Dialogul, dezbateră, studiul de caz, exemple, proiecte practice	
8. Programarea în limbaj de asamblare	Dialogul, dezbateră, studiul de caz, exemple	
9. Programarea în limbaj de asamblare	Dialogul, dezbateră, studiul de caz, exemple	
10. Programe TSR	Dialogul, dezbateră, studiul de caz, exemple, proiecte practice	
11. Programare multimodul	Dialogul, dezbateră, studiul de caz, exemple, proiecte practice	
12. Programare low-level în limbaje de nivel înalt	Dialogul, dezbateră, exemple, proiecte practice	
13. Extensii arhitecturale și noi instrucțiuni introduse la nivelul evoluției familiei de procesoare 80x86	Dialogul, dezbateră, exemple, proiecte practice	
14. Programarea în limbaj de asamblare sub Windows	Dialogul, dezbateră, exemple, proiecte practice	

Bibliografie

- BOIAN F. M., VANCEA A., IURIAN S., IURIAN M. Programare avansată de sistem și aplicații IBM-PC, lito. Universitatea "Babes-Bolyai", 1996
- STALLINGS W. Computer Organization and Architecture; Principles, Structure and Function. Mc. Millan Pub. Comp. New-York, London, 1987
- <http://homepages.fh-friedberg.de/euler/wi/skript.pdf>
- Andrew S. Tanenbaum, Computerarchitektur. Strukturen - Konzepte - Grundlagen Addison-Wesley Verlag, 2005

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

- Acest curs există în programul de studiu al tuturor universităților importante din România și străinătate
- Acest curs asigură cunoștințele de bază pe care orice programator trebuie să le aibă

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	- cunoasterea principiilor de baza ale domeniului	Examen scris	60%
	- aplicarea acestor concepte in rezolvarea de probleme		
10.5 Seminar/laborator	- implementarea in limbaj de asamblare	Teme laborator	20%
		Examen practic	20%
10.6 Standard minim de performanță			
<ul style="list-style-type: none">Pentru promovare este necesara obtinerea notei minim 5 la examenul scris, examenul practic si temele de laborator			

Data completării

13.12.2013

Titular de curs

Gaceanu Radu

Titular de seminar

Gaceanu Radu

Data avizării în departament

20.12.2013

Director de departament

Prof. Univ. Dr. Bazil Parv