

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Babeș-Bolyai
1.2 Facultatea	Facultatea de Matematică și Informatică
1.3 Departamentul	Departamentul de Informatică
1.4 Domeniul de studii	Știința calculatoarelor
1.5 Ciclul de studii	Licență
1.6 Programul de studiu / Calificarea	Informatică, linia Română

2. Date despre disciplină

2.1 Denumirea disciplinei	Verificarea și Validarea Sistemelor Soft						
2.2 Titularul activităților de curs	Lect. Dr. Camelia Chisăliță-Crețu						
2.3 Titularul activităților de seminar	Lect. Dr. Camelia Chisăliță-Crețu						
2.4 Anul de studiu	3	2.5 Semestrul	6	2.6. Tipul de evaluare	E	2.7 Regimul disciplinei	obligatoriu

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	48	Din care: 3.5 curs	24	3.6 seminar/laborator	24
Distribuția fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie și notițe					28
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					28
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					28
Tutoriat					6
Examinări					12
Alte activități:					0
3.7 Total ore studiu individual		102			
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	•
5.2 De desfășurare a seminarului/laboratorului	•

6. Competențele specifice acumulate

Competențe profesionale	<ul style="list-style-type: none"> • C2.3 Utilizarea metodologiilor, mecanismelor de specificare și a mediilor de dezvoltare pentru realizarea aplicațiilor informatice • C2.4 Utilizarea de criterii și metode adecvate pentru evaluarea aplicațiilor informatice • C2.5 Realizarea unor proiecte informatice dedicate
Competențe transversale	<ul style="list-style-type: none"> • 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 • CT3 Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> • Înțelegerea noțiunilor de algoritm parțial și total corect; • Formarea deprinderilor de proiectare a algoritmilor în paralel cu demonstrarea corectitudinii lor; • Cunoașterea metodelor de testare și verificare a sistemelor soft; • Formarea deprinderilor de proiectare a programelor corecte din specificații; • Formarea unui stil modern de programare.
7.2 Obiectivele specifice	<ul style="list-style-type: none"> • Studenții vor ști cum se desfășoară și care sunt pașii unei inspecții, fie a codului sursă fie a specificației din fiecare etapă de dezvoltare a sistemului soft. • Studenții vor ști să prevadă încă din faza de specificare și proiectare crearea unor cazuri de testare care să-i ajute la dezvoltarea unui sistem soft mai robust. • Studenții vor ști să utilizeze instrumentele pentru managementul procesului de testare. • Studenții vor ști să proiecteze cazurile de testare folosind diferite criterii (black-box, white-box).

8. Conținuturi

8.1 Curs	Metode de predare	Observații
1. Verificarea și validarea; Asigurarea calității produselor soft.	Prezentare, Demonstratii, Problematizare	
2. Testarea programelor (1): conceptul de testarea programelor; criterii de testare. testarea black-box, testarea white-box.	Prezentare, Demonstratii, Problematizare	
3. SPI, SQA, CMM. Cleanroom	Prezentare, Demonstratii, Problematizare	

4. Testarea programelor (2): tipuri de testare (T de integrare, T. de sistem, T. de regresie, T de acceptare), automatizarea testarii.	Prezentare, Demonstratii, Problematizare	
5. Testarea aplicatiilor desktop	Prezentare, Demonstratii, Problematizare	
6. Testarea aplicatiilor Web. Selenium Web Driver	Prezentare, Demonstratii, Problematizare	
7. Inspectarea programelor.	Prezentare, Demonstratii, Problematizare	
8. Executia simbolica	Prezentare, Demonstratii, Problematizare	
9. Verificarea modelelor (Model checking)	Prezentare, Demonstratii, Problematizare	
10. Teoria corectitudinii programelor. Evolutia conceptului de corectitudine. Contributiile lui Floyd, Hoare, Dijkstra, Gries, Droomey, Morgan	Prezentare, Demonstratii, Problematizare	
11. Specificarea programelor. Metoda lui Floyd pentru demonstrarea corectitudinii. Dijkstra: cea mai slaba preconditione. Rafinarea din specificatii. Axiomatizarea lui Hoare	Prezentare, Demonstratii, Problematizare	
12. Analiza metodelor de verificare (corectitudine, inspectare, testare, executie simbolica). Verificare si validare. Cum? Care? Cand?	Prezentare, Demonstratii, Problematizare	
Bibliografie		
<p>1. BALANESCU T., Corectitudinea programelor, Editura tehnica, Bucuresti 1995.</p> <p>2. DIJKSTRA, E., A constructive approach to the problem of program correctness, BIT, 8(1968), pg.174-186.</p> <p>3. DIJKSTRA, E., Guarded commands, nondeterminacy and formal derivation of programs, CACM, 18(1975), 8, pg.453-457.</p> <p>4. DROMEY G., Program Derivation. The Development of Programs From Specifications, Addison Wesley Publishing Company, 1989.</p> <p>5. FRENTIU, M., Verificarea corectitudinii programelor, Ed.Univ."Petru-Maior", 2001.</p> <p>6. GRIES, D., The Science of Programming, Springer-Verlag, Berlin, 1981.</p> <p>7. HOARE, C.A.R., An axiomatic basis for computer programming, CACM, 12(1969), pg.576-580, 583.</p> <p>8. Morgan, C., Programing from Specifications, Prentice Hall, NewYork, 1990.</p> <p>B. Internet</p>		
8.2 Seminar / laborator	Metode de predare	Observatii

<p>S1: Specificarea cazurilor de testare pe baza specificatiilor (BBT);</p> <p>L1: Proiectarea cazurilor de testare pe baza specificatiilor (BBT); Test management tool (TestLink) Issue tracker tool (Bugzilla)</p>	<p>Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii</p>	
<p>S2: Specificarea cazurilor de testare pe baza codului sursa (WBT);</p> <p>L2: Proiectarea cazurilor de testare pe baza codului sursa (WBT) Test management tool (TestLink) Issue tracker tool (Bugzilla)</p>	<p>Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii</p>	
<p>S3: Niveluri de testare</p> <p>L3: Niveluri de testare Test management tool (TestLink) Issue tracker tool (Bugzilla) Continuous Integration tool (Jenkins)</p>	<p>Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii</p>	
<p>S4: Lucrare de control BBT + WBT. Inspectare</p> <p>L4: Inspectare Inspection tool Issue tracker tool (Bugzilla)</p>	<p>Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii</p>	
<p>S5: Testare GUI/Web</p> <p>L5: Testare GUI/Web Web testing tool (e.g. Selenium Web Driver) Issue tracker tool (e.g. Bugzilla)</p>	<p>Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii</p>	
<p>S6: Corectitudine (Floyd, Hoare). Rafinare din specificatii.</p> <p>L6: Analiza statica: ESCJava2, JML</p>	<p>Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii</p>	
<p>Bibliografie</p>		

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

- Studentii vor ști cum să aplice metodele de testare pentru un produs soft, metode de testare utilizare în industrie.
- Studentii vor învăța diverse metode de verificare și validare ale unui sistem soft.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	La sfârșitul semestrului se va da o nota E pe examinarea scrisă.	Examen scris	50
10.5 Seminar/laborator	Activitatea de seminar, constând în participarea la discuții și la rezolvarea unor exerciții, va fi notată cu o nota S .	Lucrare de control + Activitatea de Seminar	25
	Activitatea de laborator va fi notată cu nota L .	Teme de laborator	25
10.6 Standard minim de performanță			
<ul style="list-style-type: none">• Studentii vor învăța și aplica metodele de testare ale unui produs soft.• Studentii vor aplica diverse metode de verificare (testare, inspectare) pentru stabilirea corectitudinii unui algoritm.• Cel puțin nota 5 (notare de la 1 la 10) la fiecare componentă din nota finală (examenul scris, laborator, activitatea de seminar).			

Data completării

30.05.2015

Semnătura titularului de curs

Lect. Dr. Camelia Chisăliță-Cretu

Semnătura titularului de seminar

Lect. Dr. Camelia Chisăliță-Cretu

Data avizării în departament

.....

Semnătura directorului de departament

Prof. Dr. Bazil Pârv