

## FIȘA DISCIPLINEI

### 1. Date despre program

1.1 Instituția de învățământ superior	<b>Universitatea Babeș-Bolyai</b>
1.2 Facultatea	<b>Facultatea de Matematică și Informatică</b>
1.3 Departamentul	<b>Departamentul de Informatică</b>
1.4 Domeniul de studii	<b>Informatică</b>
1.5 Ciclul de studii	<b>Postuniversitar</b>
1.6 Programul de studiu / Calificarea	<b>Program postuniversitar de formare și dezvoltare profesională în Informatică</b>

### 2. Date despre disciplină

2.1 Denumirea disciplinei (ro) (en)	<b>Verificare, Validare și Testare Automată</b>						
2.2 Titularul activităților de curs	<b>Lect. Dr. Camelia Chisăliță-Crețu</b>						
2.3 Titularul activităților de seminar	<b>Lect. Dr. Camelia Chisăliță-Crețu</b>						
2.4 Anul de studiu	<b>2</b>	2.5 Semestrul	<b>1</b>	2.6. Tipul de evaluare	<b>VP</b>	2.7 Regimul disciplinei	<b>Obligatorie</b>
2.8 Codul disciplinei	<b>MLR5096</b>						

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

3.1 Număr de ore pe săptămână	<b>4</b>	Din care: 3.2 curs	<b>2</b>	3.3 seminar/laborator	<b>2</b>
3.4 Total ore din planul de învățământ	<b>56</b>	Din care: 3.5 curs	<b>28</b>	3.6 seminar/laborator	<b>28</b>
Distribuția fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie și notițe					<b>14</b>
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					<b>12</b>
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					<b>14</b>
Tutoriat					<b>8</b>
Examinări					<b>18</b>
Alte activități: .....					-
3.7 Total ore studiu individual					<b>66</b>
3.8 Total ore pe semestru					<b>150</b>
3.9 Numărul de credite					<b>7</b>

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

4.1 de curriculum	<ul style="list-style-type: none"> <li>Fundamentele programării și algoritmică, Programare orientată obiect, Programare Web</li> </ul>
4.2 de competențe	<ul style="list-style-type: none"> <li>Abilități medii de programare în limbaje orientate-obiect de nivel înalt</li> </ul>

## 5. Condiții (acolo unde este cazul)

5.1 De desfășurare a cursului	<ul style="list-style-type: none"><li>• Sală de curs cu videoproiector</li></ul>
5.2 De desfășurare a seminarului/laboratorului	<ul style="list-style-type: none"><li>• Laborator cu calculatoare conectate la Internet, servere Web pentru găzduirea site-urilor Web și aplicațiilor Web dezvoltate pe baza tehnologiilor PHP, Java, .NET.</li></ul>

## 6. Competențele specifice acumulate

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

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

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"><li>• Înțelegerea notiunilor de algoritm parțial și total corect;</li><li>• Formarea deprinderilor de proiectare a algoritmilor în paralel cu demonstrarea corectitudinii lor;</li><li>• Cunoașterea metodelor de testare și verificare a sistemelor soft;</li><li>• Formarea deprinderilor de proiectare a programelor corecte din specificații;</li><li>• Formarea unui stil modern de programare.</li></ul>
7.2 Obiectivele specifice	<ul style="list-style-type: none"><li>• 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.</li><li>• 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.</li><li>• Studenții vor ști să utilizeze instrumentele pentru managementul procesului de testare.</li><li>• Studenții vor ști să proiecteze cazurile de testare folosind diferite criterii (black-box, white-box).</li></ul>

## 8. Conținuturi

8.1 Curs	Metode de predare	Observații
1. Verificarea și validarea sistemelor soft	Prezentare, Demonstratii, Problematizare	
2. Inspectarea programelor	Prezentare, Demonstratii, Problematizare	
3. Testarea programelor. Criterii de testare	Prezentare, Demonstratii, Problematizare	
4. Testare black-box <ul style="list-style-type: none"><li>• Împărțirea în clase de echivalență</li><li>• Analiza valorilor limită</li></ul>	Prezentare, Demonstratii, Problematizare	
5. Testare white-box <ul style="list-style-type: none"><li>• Graful fluxului de control. Complexitatea ciclomatică</li><li>• Criterii de acoperire bazate pe fluxul de control</li></ul>	Prezentare, Demonstratii, Problematizare	
6. Testare bazată pe experiență	Prezentare, Demonstratii, Problematizare	
7. Niveluri de testare. Tipuri de testare (1)	Prezentare, Demonstratii, Problematizare	
8. Niveluri de testare. Tipuri de testare (2)	Prezentare, Demonstratii, Problematizare	
9. Testare automată. Tool-uri pentru testare automată	Prezentare, Demonstratii, Problematizare	
10. Testarea aplicațiilor Web <ul style="list-style-type: none"><li>• Modelul Page-Steps-Tests</li><li>• Demo: Serenity BDD + Selenium Web Driver</li></ul>	Prezentare, Demonstratii, Problematizare	
11. Teoria corectitudinii programelor (1) <ul style="list-style-type: none"><li>• Evoluția conceptului de corectitudine</li><li>• Contribuțiile lui Floyd, Hoare</li></ul>	Prezentare, Demonstratii, Problematizare	
12. Teoria corectitudinii programelor (2) <ul style="list-style-type: none"><li>• Contribuțiile lui Dijkstra</li></ul>	Prezentare, Demonstratii, Problematizare	
13. Raportarea bug-urilor. RIMGEN	Prezentare, Demonstratii, Problematizare	
14. Pregătire pentru examen	Prezentare, Demonstratii, Problematizare	

### Bibliografie

#### Carti

1. Frentiu, M., Verificarea și validarea sistemelor soft, Presa Universitară Clujeană, 2010
2. R. S. Pressman, Software engineering: a practitioner's approach, seventh edition, Higher Education, 2010
3. L. Crispin, J. Gregory, Agile testing: a practical guide for testers and agile teams, Addison-Wesley, 2009

4. M. Pezzand, M. Young, Software Testing and Analysis: Process, Principles and Techniques, John Wiley & Sons, 2008
5. K. Naik, P. Tripathy, Software testing and quality assurance. Theory and Practice, A John Wiley & Sons, Inc., 2008
6. J. P. Katoen, Principles of Model Checking, MIT Press, May 2008
7. R. Patton, Software Testing, Sams Publishing, 2005
8. Glenford J. Myers, The Art of Software Testing, John Wiley & Sons, Inc., 2004
9. I. Bernstein, Practical software testing, Springer, 2002
10. Balanescu T., Corectitudinea programelor, Editura tehnica, Bucuresti 1995.
11. Morgan, C., Programing from Specifications, Prentice Hall, NewYork, 1990.
12. Dromey G., Program Derivation. The Development of Programs From Specifications, Addison Wesley Publishing Company, 1989.

### Articole

1. Hoare, C.A.R., An axiomatic basis for computer programming, CACM, 12(1969), pg.576-580, 583.
2. Dijkstra, E., A constructive approach to the problem of program correctness, BIT, 8(1968), pg.174-186.
3. Dijkstra, E., Guarded commands, nondeterminacy and formal derivation of programs, CACM, 18(1975), 8, pg.453-457.

### Tutoriale

Pentru activitatile curs/laborator se vor furniza tutoriale adaptate cerintelor temelor asociate.

8.2 Seminar / laborator	Metode de predare	Observații
<b>L1: Setup proiect</b> Tools: Java, IntelliJ, JUnit, Git	Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii	
<b>L2: Inspectare</b> Types of documents: Specification, Architecture, Source code	Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii	
<b>L3: Testare Black-Box</b> Test techniques: Equivalent Class Partitioning, Boundary Value Analysis	Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii	
<b>L4: Testare White-Box</b> Test techniques: CFG-based; Coverage Tools	Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii	
<b>L5: Managementul testelor. Niveluri de testare. Integrare continuă</b> Tools: TestLink, Jenkins	, Descoperire, Studiu individual, Exercitii	
<b>L6: Testare GUI Web</b> Tools: Selenium Web Driver, Serenity BDD	Prezentare, conversatie, Problematizare, Descoperire, Studiu individual, Exercitii	
<b>L7: Evaluarea finală a activității de laborator</b>	Prezentare, conversatie, Problematizare	

### Bibliografie

(vezi Bibliografia pentru curs)

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

- Studenții vor învăța să utilizeze instrumentele pentru managementul procesului de testare.
- Studenții vor studia diferite metode de testare care se pot aplica produselor soft.
- Studenții vor învăța să proiecteze cazurile de testare folosind diferite tehnici bazate pe criteriile: black-box, white-box.

**10. Evaluare**

Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală (%)
10.4 Curs	Examen scris, notat cu <b>E</b> .	Examen scris în sesiune	50%
	Susținerea unui proiect de testare automată, notat cu <b>P</b> .	Evaluare orală în sesiune	
	<b>Studenții pot alege forma de evaluare (examen scris sau elaborarea proiectului de testare automată)</b>		
10.5 Seminar/laborator	Se vor realiza 6 teme de laborator. Media ponderată pentru a notelor obținute va fi notată cu <b>L</b> .	Evaluare pe parcurs	50%
Observatii:			
<ul style="list-style-type: none"> <li>• <b>Activitatea de laborator nu se re-evaluează în timpul sesiunilor de examinare sau re-examinare.</b></li> <li>• <b>Activitatea de laborator pentru studenții restanțieri trebuie refăcută.</b></li> <li>• Prezența la activitatea de laborator se poate face doar cu grupa din studentul face parte.</li> <li>• Predarea cu întârziere se va penaliza cu 2 puncte din nota acordată.</li> <li>• După depășirea termenului de predare, tema de laborator va fi notată cu 1.</li> <li>• Media finală (M) se obține astfel: <math>M = 50\%L + 50\%E</math> sau P.</li> <li>• Promovarea disciplinei presupune obținerea <math>M \geq 5.00</math>.</li> </ul>			
10.6 Standard minim de performanță			
<ul style="list-style-type: none"> <li>• Studenții vor învăța să utilizeze instrumentele pentru managementul procesului de testare și vor aplica diferite criterii de proiectare a cazurilor de testare (black-box, white-box).</li> <li>• Studenții vor studia diferite metode de verificare (inspectare, testare, demonstrarea corectitudinii).</li> </ul>			

Data completării

Semnătura titularului de curs

Semnătura titularului de seminar

**14 Septembrie 2018**

**Lector dr. Chisăliță-Crețu Camelia**

**Lector dr. Chisăliță-Crețu Camelia**

Data avizării în departament

Semnătura directorului de departament

.....

**...Prof. dr. Anca Andreica.....**