

# FIȘA DISCIPLINEI

## Ingineria programelor 2021-2022

### 1. Date despre program

1.1	Instituția de învățământ superior	Universitatea din Pitești
1.2	Facultatea	Electronică, Comunicații și Calculatoare
1.3	Departamentul	Electronică, Calculatoare și Inginerie Electrică
1.4	Domeniul de studii	Calculatoare și tehnologia informației
1.5	Ciclul de studii	Licență
1.6	Programul de studii / Calificarea	Calculatoare / Programator (251202), Inginer de sistem în informatică (251203), Programator de sistem informatic (251204), Inginer de sistem software (251205)

### 2. Date despre disciplină

2.1	Denumirea disciplinei				<b>Ingineria programelor</b>						
2.2	Titularul activităților de curs				Lect. univ. dr. Cristina TUDOSE						
2.3	Titularul activităților de laborator				Dr. Adriana OPREA / Lect. univ. dr. Cristina TUDOSE						
2.4	Anul de studii	III	2.5	Semestrul	I	2.6	Tipul de evaluare	E	2.7	Regimul disciplinei	D/O

### 3. Timpul total estimat

3.1	Număr de ore pe săptămână	4	3.2	din care curs	2	3.3	laborator	2
3.4	Total ore din planul de învățământ	56	3.5	din care curs	28	3.6	laborator	28
Distribuția fondului de timp								ore
Studiul după manual, suport de curs, bibliografie și notițe								24
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren								19
Pregătire seminarii/laboratoare, teme, referate, portofolii, eseuri								18
Tutorat								2
Examinări								6
Alte activități .....								-
3.7	Total ore studiu individual	69						
3.8	Total ore pe semestru	125						
3.9	Număr de credite	5						

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

4.1	De curriculum	Parcursarea cursurilor de <i>Programarea calculatoarelor și limbaje de programare</i> și <i>Programare orientată pe obiecte</i>
4.2	De competențe	Operarea cu fundamente științifice, ingineresti și ale informaticii

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

5.1	De desfășurare a cursului	Sală de curs cu o capacitate de minim 100 locuri dotată cu videoproiector și ecran de proiecție, 2 table.
5.2	De desfășurare a laboratorului	Sală de laborator de informatică, capacitate maximă 20 studenți/laborator.

### 6. Competențe specifice acumulate

Competențe profesionale	C6 Proiectarea sistemelor inteligente
Competențe transversale	

### 7. Obiectivele disciplinei

7.1 Obiectivul general al disciplinei	- Familiarizarea cu conceptele de bază ale ingineriei software: procese software, cerințe, modelare, testare, dezvoltare.
7.2 Obiectivele specifice	<i>Obiective cognitive</i>

	<ul style="list-style-type: none"> <li>- Cunoașterea etapelor realizării unui proiect software.</li> </ul> <p><i>Obiective procedurale</i></p> <ul style="list-style-type: none"> <li>- Formarea abilităților de modelare, testare și verificare a unui sistem.</li> <li>- Formarea deprinderii de a specifica și verifica formal modelul unui sistem.</li> </ul> <p><i>Obiective atitudinale</i></p> <ul style="list-style-type: none"> <li>- Rigurozitate în realizarea testării și verificării formale în programare.</li> </ul>
--	---

## 8. Conținuturi

8.1. Curs		Metode de predare	Observații Resurse folosite
1	<b>Introducere în Ingineria Programelor</b> 1.1. Scurt istoric al Ingineriei Programelor 1.2. Definiții generale ale Ingineriei Programelor 1.3. Concepte de bază ale Ingineriei Programelor  Timp alocat: 2 ore	Prelegerea Explicația Descrierea și exemplificarea Conversația euristică	Calculator, Videoproiector, Suport documentar
2	<b>Procese software</b> 2.1. Modele de procese software 2.2. Etape și riscuri 2.3. Definirea și managementul cerințe  Timp alocat: 4 ore	Prelegerea Explicația Descrierea și exemplificarea Conversația euristică Studiul de caz	Calculator, Videoproiector, Suport documentar
3	<b>UML (Unified Modelling Language)</b> 3.1. Definirea modelului 3.2. Cazuri de utilizare, diagrame de clase, diagrame de obiecte, diagrame de interacțiune, diagrame de stări, diagrame de activitate 3.3. Studiu de caz  Timp alocat: 8 ore	Prelegerea Explicația Descrierea și exemplificarea Conversația euristică Studiul de caz	Calculator, Videoproiector, Suport documentar
4	<b>Metode de testare a programelor</b> 4.1. Testare funcțională 4.2. Testare structurală 4.3. Alte metode de testare  Timp alocat: 6 ore	Prelegerea Explicația Descrierea și exemplificarea Conversația euristică Studiul de caz	Calculator, Videoproiector, Suport documentar
5	<b>Verificare și validare software</b> 5.1. Generalități 5.2. Revizii, audituri, demonstrații formale 5.3. Metode de specificare formală  Timp alocat: 4 ore	Prelegerea Explicația Descrierea și exemplificarea Conversația euristică Studiul de caz	Calculator, Videoproiector, Suport documentar
6	<b>Managementul unui proiect; standarde de evaluare a calității</b>  Timp alocat: 2 ore	Prelegerea Explicația Descrierea și exemplificarea Conversația euristică Studiul de caz	Calculator, Videoproiector, Suport documentar
7	<b>Metodologia Agilă</b>  Timp alocat: 2 ore	Prelegerea Explicația Descrierea și exemplificarea Conversația euristică Studiul de caz	Calculator, Videoproiector, Suport documentar

### Bibliografie:

1. Filipova O., Vilão R., Software development from A to Z: A deep dive into all the roles involved in the creation of software, 2018
2. Sommerville I., Software engineering, Tenth Edition, 2016.  
<https://iansommerville.com/software-engineering-book/>
3. Dennis A., Wixom B., Tegarden D., Systems Analysis and Design: An Object-Oriented Approach with UML, 5th Edition, Wiley, 2015
4. Mitchell J.L, Black R., Advanced software testing, vol.3, 2nd Edition, 2015
5. Varvara G, Ingineria Programării, Ed. ConsPress, 2013.
6. Novac Ududec C., Ingineria sistemelor de programe, Ed. Alma Mater Bacau, 2011
7. Baier, C., Katoen, J.-P, Principles of Model Checking. The MIT Press, 2008
8. Ipate F., Modelare orientată pe obiecte cu UML, Ed. Universității Pitești, 2001.
9. Turcanu A, Tudose C, Ingineria Programelor-note de curs, format electronic

8.2. Aplicații - Laborator		Metode de predare	Observații Resurse folosite
1	<b>Elaborarea unor modele de procese software pornind de la scenariile date; managementul cerințelor</b>  Timp alocat: 4 ore	Exercițiul Explicația Studiul de caz Dezbateră Lucrul în grup	Calculatoare, Tablă, Suport documentar
2	<b>Modelarea folosind UML: cazuri de utilizare, diagrame de clase, diagrame de interacțiune, diagrame de stări, diagrame de activitate</b> 2.1. Studii de caz pornind de la scenariile date 2.2. Utilizarea StarUML/ArgoUML pentru realizarea diagramelor 2.3. Generarea de cod folosind StarUML  Timp alocat: 16 ore	Exercițiul Explicația Studiul de caz Dezbateră Lucrul în grup	Calculatoare, Tablă, Suport documentar
3	<b>Testarea și verificarea programelor</b> 3.1. Exemplificarea unor metode de testare: funcțională, structurală, bazată pe mutații 3.2. Folosirea unor instrumente automate de testare: JUnit, MuJava  Timp alocat: 4 ore	Exercițiul Explicația Studiul de caz Dezbateră Lucrul în grup	Calculatoare, Tablă, Suport documentar
4	<b>Managementul proiectelor și managementul calității</b> 4.1. Studiu de caz 4.2. Utilizarea unor instrumente dedicate managementului proiectelor  Timp alocat: 4 ore	Exercițiul Explicația Studiul de caz Dezbateră Lucrul în grup	Calculatoare, Tablă, Suport documentar
Bibliografie: 1. Filipova O., Vilão R., Software development from A to Z: A deep dive into all the roles involved in the creation of software, 2018 2. Sommerville I., Software engineering, Tenth Edition, 2016. <a href="https://iansommerville.com/software-engineering-book/">https://iansommerville.com/software-engineering-book/</a> 3. Dennis A., Wixom B., Tegarden D., Systems Analysis and Design: An Object-Oriented Approach with UML, 5th Edition, Wiley, 2015 4. Mitchell J.L, Black R., Advanced software testing, vol.3, 2nd Edition, 2015 5. Varvara G, Ingineria Programării, Ed. ConsPress, 2013. 6. Novac Ududec C., Ingineria sistemelor de programe, Ed. Alma Mater Bacau, 2011 7. Baier, C., Katoen, J.-P, Principles of Model Checking. The MIT Press, 2008 8. Ipate F., Modelare orientată pe obiecte cu UML, Ed. Universității Pitești, 2001. 9. Turcanu A, Tudose C, Ingineria Programelor-note de curs, format electronic			

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

Competențele dobândite în cadrul disciplinei permit absolvenților să utilizeze eficient modelele și tehnicile specifice ingineriei programelor în rezolvarea cerințelor aferente practicii și cercetării în domeniul programării calculatoarelor și tehnologiei informației.

**10. Evaluare**

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Evaluare finală	Probă scrisă	50%
10.5 Laborator	Activitate laborator Proiect	Verificare soluții Verificare proiect	20% 30%
10.6 Standard minim de performanță	* Note de minim 5 la activitatea de laborator, la proiect și la evaluarea finală (rezolvarea în proporție de 50% a cerințelor); nota finală minim 5. * Set de cunoștințe minimale pentru promovarea examenului final: - Cunoașterea principalelor noțiuni privind etapele elaborării unui proiect software; - Cunoașterea principalelor tehnici de modelare UML - Cunoașterea unor noțiuni fundamentale de testare și validare a programelor.		

Data completării  
20.09.2021

Titular de curs  
Lect. univ. dr. Cristina TUDOSE

Titular de laborator  
Dr. Adriana OPREA,  
Lect. univ. dr. Cristina TUDOSE

Data aprobării în Consiliul departamentului, 27.09.2021

Director de departament,  
(prestator)  
Conf. univ. dr. Doru CONSTANTIN

Director de departament,  
(beneficiar)  
Prof. univ. dr. ing. Gheorghe Șerban