

FIȘA DISCIPLINEI

Tehnici avansate de programare

Anul universitar 2020-2021

1. Date despre program

1.1	Instituția de învățământ superior	Universitatea din Pitești
1.2	Facultatea	Științe, Educație Fizică și Informatică
1.3	Departamentul	Matematică-Informatică
1.4	Domeniul de studii	Informatică
1.5	Ciclul de studii	Licență
1.6	Programul de studii / Calificarea	Informatică / Informatică

2. Date despre disciplină

2.1	Denumirea disciplinei		Tehnici avansate de programare								
2.2	Titularul activităților de curs		Lect. univ. dr. Cristina Tudose								
2.3	Titularul activităților de laborator		Lect. univ. dr. Cristina Tudose								
2.4	Anul de studii	3	2.5	Semestrul	2	2.6	Tipul de evaluare	Examen	2.7	Regimul disciplinei	O

3. Timpul total estimat

3.1	Număr de ore pe săptămână	5	3.2	din care curs	2	3.3	seminar	1	3.4	laborator	2
3.5	Total ore din planul de inv.	60	3.6	din care curs	24	3.7	seminar	12	3.8	laborator	24
Distribuția fondului de timp											Ore
Studii după manual, suport de curs, bibliografie și notițe											30
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren											30
Pregătire seminarii/laboratoare, teme, referate, portofolii, eseuri											40
Tutoriat											9
Examinări											6
Alte activități											-
3.9	Total ore studiu individual	115									
3.10	Total ore pe semestru	175									
3.11	Număr de credite	7									

4. Precondiții (acolo unde este cazul)

4.1	De curriculum	cunoștințe de programare
4.2	De competențe	capacitatea de sinteză

5. Condiții (acolo unde este cazul)

5.1	De desfășurare a cursului	Sală dotată cu videoprojector
5.2	De desfășurare a seminarului/laboratorului	Laborator de informatică

6. Competențe specifice acumulate

Competențe profesionale	<ul style="list-style-type: none"> Programarea în limbaje de nivel înalt; Dezvoltarea și întreținerea aplicațiilor informatice; Utilizarea bazelor teoretice ale informaticii și a modelelor formale.
Competențe transversale	<ul style="list-style-type: none"> 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; Desfășurarea eficientă a activităților organizate într-un grup interdisciplinar și dezvoltarea capacităților empatică de comunicare inter-personală, de relaționare și colaborare cu grupuri diverse; 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.

7. Obiectivele disciplinei

7.1	Obiectivul general al disciplinei	Disciplina are ca obiectiv general însușirea de către studenți a cunoștințelor de bază, metodelor și tehnicilor privind tehnicile de programare de bază și a celor avansate, a modalităților de determinare prin calcul, implementare și de aplicare adecvată la situații concrete a acestor metode și tehnici.
7.2	Obiectivele specifice	<p><i>Obiective cognitive:</i></p> <ul style="list-style-type: none"> ► Cunoașterea principiilor fundamentale și tehnicilor de programare specifice limbajului Java. <p><i>Obiective procedurale:</i></p> <ul style="list-style-type: none"> ► Formarea deprinderilor și abilitatea de a utiliza și de a realiza interfețe grafice, de proiectare a sistemelor distribuite și a aplicațiilor Web. <p><i>Obiective atitudinale:</i></p> <ul style="list-style-type: none"> ► Rigurozitate în proiectarea și implementarea algoritmilor specifici tehnicilor de programare.

8. Conținuturi

8.1. Curs	Nr. ore	Metode de predare	Observații Resurse folosite
-----------	---------	-------------------	--------------------------------

1.	Recapitularea facilităților principale de programare orientată pe obiecte prezente în limbajul Java	2	Explicația Algoritmizarea Descrierea și exemplificarea Demonstrația Problematizarea Conversația euristică Exercițiul	Calculator Videoprojector
2.	Interfata grafică cu utilizatorul: Awt.	2		
3.	Interfata grafică cu utilizatorul: Swing.	2		
4.	Gestionari de poziționare; componente Swing; tratarea evenimentelor.	2		
5.	Interfata grafică cu utilizatorul: JavaFX.	2		
6.	Servicii Java de acces la resurse; Servicii de nume; Conectare la baze de date	2		
7.	Serializarea obiectelor, fire de execuție, socketuri	2		
8.	Comunicații Java folosind serviciile standard ale pachetului java.net; Transmiterea și recepționarea prin UDP; Transmiterea și recepționarea prin TCP Tehnologia CGI	2		
9.	Apelul la distanță al metodelor; Tehnologia Remote Procedure Call; Tehnologia Remote Method Invocation; Clase Stub și Skeleton; RMI cu căutare folosind un serviciu de nume	2		
10.	Dezvoltarea aplicațiilor distribuite eterogene; Standardul CORBA; Limbajul IDL (Interface Definition Language); Mapare IDL în Java; Object Request Broker și Portable Object Adapter	2		
11.	Procesarea documentelor XML; Metalimbajul eXtensible Markup Language; Procesarea bazată pe evenimente; SAX (Simple API for XML Parsing); Prelucrarea arborilor asociați documentelor XML; DOM (Document Object Model)	2		
12.	Tehnologii Web; Tehnologia java Servlet; Tehnologia JSP (Java Server Pages)	2		

Bibliografie

1. M. Aleksy, A. Korhous, M. Shader: Implementing Distributed Systems with Java and Corba, Springer, 2008.
2. T. Bălănescu, Ș. Mocanu: Interfețe grafice în Java, Editura Fundației România de Măine, 2007.
3. F. M. Boian, R. F. Boian: Tehnologii fundamentale Java pentru aplicații Web, Editura Albastră, 2004.
4. D. Gries, P. Gries: Multimedia Introduction to Programming Using Java, Springer, 2011.
5. Ș. Tănasă, C. Olaru, Ș. Andrei: Java, de la 0 la expert, Polirom, 2006.
6. D. Anastasiu Popescu, Bazele programării - Java după C++, ISBN: 978-973-88037-9-4

8.2. Aplicații – Seminar + Laborator

	Nr. ore	Metode de predare	Observații Resurse folosite
1	12	Proiectarea și implementarea aplicațiilor stand-alone și applet	Calculator Videoprojector
2	4	Serviciul de nume JNDI; Scenariu de conectare la o bază de date utilizând JDBC	
3	4	Consultarea unui server Time; Aplicație RemoteDir	
4	4	Implementarea unui obiect serviant; Aplicație RMI cu utilizarea serviciului de nume JNDI	
5	4	Definirea unei interfețe IDL și translatarea ei în Java; Implementarea obiectelor serviant și a obiectului server; Implementarea obiectelor client	
6	4	Utilizare SAX pentru prelucrarea informațiilor dintr-un document XML; Crearea și manipularea unui arbore DOM asociat unui document XML	
7	4	Aplicație: un servlet care primește ca argumente două numere întregi și calculează suma lor. Aplicație JSP: gestionarea notelor unei grupe de studenți.	

Bibliografie

1. M. Aleksy, A. Korhous, M. Shader: Implementing Distributed Systems with Java and Corba, Springer, 2008.
2. T. Bălănescu, Ș. Mocanu: Interfețe grafice în Java, Editura Fundației România de Măine, 2007.
3. F. M. Boian, R. F. Boian: Tehnologii fundamentale Java pentru aplicații Web, Editura Albastră, 2004.
4. D. Gries, P. Gries: Multimedia Introduction to Programming Using Java, Springer, 2011.
5. Ș. Tănasă, C. Olaru, Ș. Andrei: Java, de la 0 la expert, Polirom, 2006.
6. D. Anastasiu Popescu, Bazele programării - Java după C++, ISBN: 978-973-88037-9-4

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

Coroborarea conținuturilor disciplinei cu așteptările comunității epistemice se realizează prin stabilirea de întâlniri cu principalii actori de pe piața IT locală (RoWeb, Lisa, Prodinf, Kepler, Osf, Endava, etc.), precum și prin vizite la firmele de profil și schimburi de bune practici cu cadre didactice din alte universități.

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 (algoritmi și probleme)	50%
10.5 Laborator	Temă de casă Activitate (rezolvarea problemelor propuse)	Verificare temă de casă Verificare soluții	30% 20%
10.6 Standard minim de performanță	*Set de cunoștințe minimale pentru promovarea examenului final: cunoașterea principiilor fundamentale ale unui limbaj de programare; cunoașterea elementelor de bază, implementări ale unor algoritmi de bază specifici tehnicilor de programare.		

Data completării
20.09.2020

Titular de curs
Lect.univ.dr. Cristina TUDOSE

Titular de laborator
Lect.univ.dr. Cristina TUDOSE

Data avizării în Departament
23.09.2020

Director Departament (*prestator*)
Conf.univ.dr. Doru CONSTANTIN

Director Departament (*beneficiar*)
Conf.univ.dr. Doru CONSTANTIN