

Numele disciplinei:	Aplicații JAVA						
Facultatea:	FH		Domeniu:		ISI		
Program de studii	AIA						
Cod plan:	21		Cod disciplină:		847		
Anul de studiu:	4	Forma de examinare: (E-Examen; C- Colocviu; P-Proiect; A/R- Admis/ Respins)		C	Credite ECTS (CR):	E(C)	3
Semestrul:	7					P	0
Categoriza disciplinei: (DF - Fundamentală; DD - Domeniu; DS -Specialitate; DC - Complementară; DA - Aprofundare)							DS
Tipul disciplinei: (OB-Obligatorie; OP-Opțională; FC-Facultativă)							OP
Numărul de ore de activitate didactică pe semestru:		Activitate didactică neasistată		33	Activitate didactică asistată parțial		0
Activitate didactică asistată integral:	42	Din care, săptămânal:					
		Curs		Seminar		Laborator	Proiect
		2		0		1	0

Departament	DIH
Cadru didactic titular:	Șef lucrări Ramona-Oana Flangea

Obiectivele disciplinei (descrierea rezultatelor învățării):
1. Cunoștințe - Cunoașterea fundamentelor limbajului Java: sintaxă, tipuri de date, controlul fluxului, funcții și metode. - Înțelegerea paradigmei programării orientate pe obiecte: clase, obiecte, moștenire, polimorfism, încapsulare, abstractizare. - Familiarizarea cu bibliotecile standard Java: colecții (List, Set, Map), stream-uri, manipularea fișierelor, threading și concurență. - Cunoașterea tehnologiilor Java pentru aplicații de desktop și web: Swing, JavaFX, JSP, Servlets, Spring Framework. - Înțelegerea gestionării bazelor de date în Java: JDBC, Hibernate, JPA. - Cunoașterea principiilor de testare a aplicațiilor Java: JUnit, Mockito, teste unitare, teste de integrare. - Înțelegerea conceptelor de gestionare a excepțiilor și utilizarea gestionării memoriei în Java (garbage collection). 2. Abilități - Dezvoltarea de aplicații Java desktop utilizând Swing și JavaFX pentru interfețe grafice. - Implementarea de aplicații web Java folosind JSP, Servlets, Spring MVC. - Utilizarea tehnologiilor de persistență în aplicații (ex: JDBC, Hibernate, JPA pentru gestionarea bazei de date). - Implementarea principiilor programării orientate pe obiecte și a modelelor de design (ex: Singleton, Factory, Observer) în aplicații Java. - Utilizarea tehnicilor de concurență în aplicații multi-threading pentru a rezolva probleme de performanță. - Realizarea de aplicații mobile (Android) folosind limbajul Java. - Testarea aplicațiilor Java și utilizarea framework-urilor de testare (ex: JUnit, TestNG). - Utilizarea framework-urilor moderne Java (ex: Spring, Hibernate) pentru dezvoltarea de aplicații robuste și scalabile.

3. Responsabilitate și autonomie - Proiectarea, dezvoltarea și implementarea independentă a unor aplicații Java complete (de la aplicații desktop la aplicații web). - Respectarea bunelor practici de codare și a principiilor programării orientate pe obiecte (coerență, reutilizare, modularizare). - Asumarea responsabilității pentru calitatea și performanța aplicațiilor dezvoltate, inclusiv testarea acestora în diferite scenarii de utilizare. - Documentarea aplicațiilor Java: comentarii în cod, generarea de documentație externă (ex: Javadoc). - Participarea activă la dezvoltarea aplicațiilor în cadrul unor echipe agile sau de proiect, urmând procesele de versionare și gestionare a codului sursă (ex: Git). - Analiza și rezolvarea problemelor complexe de software, propunerea de soluții eficiente și testarea acestora.	
Descrierea cursului:	
1. Curs	1. Concepte și paradigme în programarea orientată pe obiecte (2ore) 2. Structuri de control în Java și tipuri de date abstracte (2ore) 3. Clase și Obiecte. Tablouri (2ore) 5. Pachete. Interfețe Java (2ore) 6. Moștenire și polimorfism. Clasa Object. Clase interne (4ore) 7. Erori și excepții în Java (4ore) 8. Dezvoltarea interfețelor grafice în Java. Mecanismul de tratarea a evenimentelor în Java. (4ore) 9. Java Collection Framework - SET si LIST (4ore) 10. Programare concurenta si multithreading (4ore)
2. Seminar/ Laborator/ Proiect/ Practică	1. Familiarizarea cu mediile de dezvoltare Eclipse si IntelliJ (2ore) 2. Tipuri primitive în Java, Variabile și expresii în Java (2ore) 3. Controlul fluxului și clase în Java (2ore) 4. Interfețe Java (2ore) 5. Tehnici de reutilizare a claselor - Moștenirea si polimorfism în Java (2ore) 6. Dezvoltarea interfețelor grafice utilizator. Se atribuie tema de miniproiect (2ore) 7. Lucru cu colectii in java – Lucru cu structuri de tip lista, set-uri, map (2ore)
3. Bibliografie	1. Ștefan Tănasă, Cristian Olaru, Ștefan Andrei, Java de la 0 la expert, Ed. Polirom 2003 2. Călin Văduva, Programare în Java, Editura Albastră, 2001. 3. Bruce Eckel, Thinking in Java, Third Edition, Prentice Hall PTR, 2002.

Examinarea:	Ponderea fiecărui criteriu în nota finală
1. Examinarea finală	
2. Verificarea cunoștințelor pe parcurs	
Seminar	
Laborator	50%
Proiect (proiectul nu are notă distinctă)	
3. Evaluări periodice:	
3.1 Examinări scrise / orale	50%
3.2 Teme, rapoarte, etc.	
4. Alte criterii (se vor specifica)	
Scurtă descriere a procedurii de examinare finală:	
Examinare finală prin evaluare asistată de calculator utilizând platforma https://moodle.utcb.ro	

Estimarea numărului total de ore pe semestru pentru studiu individual			
Tipul de activitate individuală	Nr. ore	Tipul de activitate individuală	Nr. ore
1. Studiu notițe de curs	6	8. Studiu pentru examinarea finală	6
2. Studiu bibliografie obligatorie	3	9. Ședințe de consultații	0
3. Studiu bibliografie suplimentară	0	10. Documentare practică pe teren	0
4. Pregătire activități specific disciplinei	6	11. Studiu la bibliotecă adițional	0
5. Pregătire teme	0	12. Studiu resurse internet	6
6. Studiu pentru evaluări periodice scrise	6	13. Alte activități (se enumeră)	0
7. Studiu pentru evaluări periodice orale	0	Numărul total de ore:	33

Data:	Decan
01 octombrie 2024	Conf. dr. ing. Alexandru Dimache
	Director de departament:
	Conf. dr. ing. Neculoiu Giorgian
	Titular de disciplină:
	Șef lucrări Ramona-Oana Flangea