

Numele disciplinei:	Programarea calculatoarelor și limbaje de programare III						
Facultatea:	FH		Domeniu:		ISI		
Program de studii	AIA						
Cod plan:	21		Cod disciplină:		790		
Anul de studiu:	2	Forma de examinare: (E-Examen; C- Colocviu; P-Proiect; A/R- Admis/ Respins)		E	Credite ECTS (CR):	E(C)	6
Semestrul:	3					P	0
Categoriza disciplinei: (DF - Fundamentală; DD - Domeniu; DS -Specialitate; DC - Complementară; DA - Aprofundare)							DF
Tipul disciplinei: (OB-Obligatorie; OP-Opțională; FC-Facultativă)							OB
Numărul de ore de activitate didactică pe semestru:		Activitate didactică neasistată		66	Activitate didactică asistată parțial		0
Activitate didactică asistată integral:	84	Din care, săptămânal:					
		Curs		Seminar		Laborator	Proiect
		3	0		3	0	

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

Obiectivele disciplinei (descrierea rezultatelor învățării):
1. Cunoștințe Înțelegerea principiilor fundamentale ale programării orientate pe obiecte: încapsulare, moștenire, polimorfism, abstracție: - Cunoașterea structurii unui program Java: sintaxă, clase, metode, obiecte. - Familiarizarea cu concepte avansate: interfețe, clase abstracte, excepții, colecții, fluxuri de date, expresii. - Cunoașterea mecanismelor de gestionare a memoriei și a ciclului de viață al obiectelor în Java. - Înțelegerea relațiilor între OOP și alte paradigme de programare, pentru a permite o tranziție facilă către alte limbaje OOP. 2. Abilități - Aplicarea principiilor OOP în proiectarea și implementarea de soluții software în Java. - Dezvoltarea de aplicații Java robuste, modulare și reutilizabile. - Utilizarea mediilor de dezvoltare (IDE-uri precum IntelliJ, Eclipse) pentru scrierea, testarea și depanarea codului. - Capacitatea de a interpreta și adapta cod sursă scris de alții. - Implementarea unor structuri de date și algoritmi de bază folosind conceptele OOP. 3. Responsabilitate și autonomie - Dezvoltarea autonomă a unor proiecte software simple și medii, respectând principiile OOP. - Exersarea disciplinei de codare curată, documentată și ușor de întreținut. - Asumarea responsabilității în cadrul lucrului în echipă prin colaborare pe proiecte de grup (versionare cu Git, code reviews). - Gestionarea timpului și a sarcinilor în cadrul dezvoltării de proiecte software cu termene de livrare. - Evaluarea și îmbunătățirea continuă a propriului cod prin feedback și testare.
Descrierea cursului:

1. Curs	1. Noțiuni introductive și elemente de bază ale limbajului Java (3 ore). 2. Programarea orientată pe obiecte (POO) (3 ore) 3. Siruri de caractere (6 ore) 4. Aspecte de detaliu ale programării orientate obiect (6 ore) 5. Pachete de clase de obiecte. Mostenirea multiplă în Java (6 ore) 6. Tratarea erorilor (3 ore) 7. Fluxuri de intrare/ieșire. Fișiere (6 ore) 8. Fire de execuție (3 ore) 9. Interfața grafică în limbajul Java (6 ore)
2. Seminar/ Laborator/ Proiect/ Practică	1. Aplicații cu elemente de bază ale limbajului Java - Eclipse IDE (citire, afișare, calcul suma cifrelor unui număr, suma numerelor naturale) (3 ore) 2. Aplicații cu variabile de tip referință, tipuri structurate de date - tablouri unidimensionale și bidimensionale (3 ore) 3. Aplicații cu siruri de caractere (concatenare, comparare, căutare, înlocuire, separare elemente lexicale) (6 ore) 4. Aplicații cu concepte de bază ale POO (încapsulare, moștenire, polimorfism), definirea claselor, metode (6 ore) 5. Aplicații de creare a interfețelor, clase și interfețe generice (6 ore) 6. Aplicații pentru tratarea excepțiilor Java (creare, semnalare, interceptare) (3 ore) 7. Aplicații cu fișiere (fluxuri de intrare/ieșire) (6 ore) 8. Aplicații cu fire de execuție (creare, pornire, sincronizare) 9. Aplicații grafice cu Windows Builder (6 ore)
3. Bibliografie	1. Georgescu, H., Introducere în universul Java, Editura Tehnică, 2002. 2. Sorin, T., Huțanu, V., Bazele programării în JAVA, Editura L&S Info-Mat, 2005. 3. Marinescu M., Programarea orientată obiect cu limbajul Java în 14 lecții, Editura CONSPRESS, 2008.

Examinarea:	Ponderea fiecărui criteriu în nota finală
1. Examinarea finală	50%
2. Verificarea cunoștințelor pe parcurs	
Seminar	
Laborator	10%
Proiect (proiectul nu are notă distinctă)	
3. Evaluări periodice:	
3.1 Examinări scrise / orale	40%
3.2 Teme, rapoarte, etc.	
4. Alte criterii (se vor specifica)	
Scurtă descriere a procedurii de examinare finală:	
Evaluarea finală se va realiza prin: 1. test tip grila, asistat de calculator pe platforma https://moodle.utcb.ro/login/index.php , 2. test practic de rezolvarea a unei probleme transpuse în limbajul de programare Java	

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	12
2. Studiu bibliografie obligatorie	12	9. Ședințe de consultații	6
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	12
6. Studiu pentru evaluări periodice scrise	12	13. Alte activități (se enumeră)	0
7. Studiu pentru evaluări periodice orale	0	Numărul total de ore:	66

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