



FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea “Alexandru Ioan Cuza” din Iași
1.2 Facultatea	Facultatea de Economie și Administrarea Afacerilor
1.3 Departamentul	Contabilitate, Informatică Economică și Statistică
1.4 Domeniul de studii	Cibernetica, Statistica și Informatica Economică
1.5 Ciclul de studii	Licență
1.6 Programul de studii / Calificarea	Informatică Economică

2. Date despre disciplină

2.1 Denumirea disciplinei	Programare II						
2.2 Titularul activităților de curs	Conf.dr. CĂTĂLIN STRÎMBEI						
2.3 Titularul activităților de seminar	Conf. dr. SABINA NECULA, Lect.dr. IONUȚ HRUBARU						
2.4 An de studiu	3	2.5 Semestru	1	2.6 Tip de evaluare	E	2.7 Regimul disciplinei*	OB

* OB – Obligatoriu / OP – Opțional

3. Timpul total estimat (ore pe semestru și activități didactice)

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar/laborator	2
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar/laborator	28
Distribuția fondului de timp					Ore
Studiu după manual, suport de curs, bibliografie și altele					20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					15
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					25
Tutoriat					5
Examinări					4
Alte activități					
3.7 Total ore studiu individual					69
3.8 Total ore pe semestru					125
3.9 Număr de credite					5

4. Precondiții (dacă este cazul)

4.1 De curriculum	Programare I, Algoritmi și logica programării, Baze de date
4.2 De competențe	Introducere în limbaje de programare, algoritmi, structuri de date, SQL

5. Condiții (dacă este cazul)

5.1 De desfășurare a cursului	Sala curs cu videoproiector și laptop: JDK, IntelliJ IDEA, Eclipse IDE for JEE Developers, PostgreSQL
5.2 De desfășurare a seminarului/laboratorului	Sala laborator, access internet inclusiv wireless: JDK, IntelliJ IDEA, Eclipse IDE for JEE Developers, PostgreSQL



6. Competențe specifice acumulate

Competențe profesionale	<p>C1.1 Implementarea de soluții orientate obiect reutilizabile și integrabile în sisteme informatice (1 credit)</p> <p>C 1.2 Explicarea structurilor de date orientate obiect, a instrucțiunilor și claselor de probleme pentru a construi componente integrabile în sisteme software complexe (1 credit)</p> <p>C1.3 Actualizarea limbajelor, tehnicilor și metodelor de programare orientate obiect, astfel încât componentele software construite să reflecte stadiul dezvoltării IT&C (1 credit)</p> <p>C1.4 Recunoașterea tipurilor de aplicații care necesită lucrul cu baze de date, resurse multimedia și tehnologii client-server în vederea realizării de componente integrabile în sisteme informatice (1 credit)</p> <p>C1.4 Evidențierea prin studii și analize a impactului utilizării sistemelor de gestiune a bazelor de date, resurselor multimedia și a tehnologiilor client-server din organizație cu luarea în considerare a optimizării resurselor, a proceselor și a eficienței economice (1 credit)</p>
Competențe transversale	<p>CT1 Identificarea rolurilor și responsabilităților într-o echipă plurispecializată și aplicarea de tehnici de relaționare și muncă eficientă în cadrul echipei.</p>

7. Obiectivele disciplinei (din grila competențelor specifice acumulate)

7.1 Obiectivul general	<p>Înșușirea conceptelor și teoriilor fundamentale cu privire la limbajele, tehnicile și metodele de programare orientate obiect. Dezvoltarea de aplicații pentru afaceri pe principii orientate obiect.</p>
7.2 Obiectivele specifice	<p>Înțelegerea rolului, locului și impactului implementării soluțiilor elaborate pe principiile programării orientate obiect în cadrul (contextul) procesului de dezvoltare (software engineering) a componentelor reutilizabile ale aplicațiilor economice.</p> <p>Dobândirea aptitudinilor necesare dezvoltării soluțiilor software pentru afaceri folosind platforme moderne bazate pe limbaje de programare orientate obiect în contextual sistemelor software complexe.</p> <p>Dobândirea cunoștințelor de bază privind integrarea componentelor reutilizabile orientate obiect pornind de la arhitecturi client-server cu baze de date până la arhitecturi stratificate pentru Web și client desktop.</p>

**8. Conținut**

8.1	Curs	Metode de predare	Observații (ore și referințe bibliografice)
1.	Capitolul 1. Obiecte, clase și aplicații. Tipuri Java fundamentale <ul style="list-style-type: none">• Principiul abstractizării cu obiecte• Clase și obiecte în limbajul Java• Funcții și metode statice• Execuția aplicațiilor Java și consola Java• Sistemul de tipuri al limbajului Java<ul style="list-style-type: none">○ Lucru cu primitive și referințe (autoboxing)○ Lucru cu tablouri○ Lucru cu numere și șiruri de caractere○ Lucru cu date calendaristice și timp○ Tipuri enumerative: Enum(s)○ Tipuri de date: clase Record○ Tipuri excepții	Prelegere curs	4 ore
2.	Capitolul 2 Principii OO: abstractizare, compunere și moștenire <ul style="list-style-type: none">• Compunerea claselor• Moștenire și sub-tipizare• Interfețe și tipuri abstracte de date• Clase abstracte• Clase pentru date: tipuri Record• Șabloane de proiectare orientată obiect: Singleton, Factory Method și Builder (flow API)	Prelegere curs	4 ore
3.	Capitolul 3. Polimorfism și genericitate <ul style="list-style-type: none">• Polimorfism• Genericitate• Clase interne și anonime• Tehnici generice specifice pentru:<ul style="list-style-type: none">○ Comparabilitate○ Gestionarea valorilor null cu tipuri <i>Optional</i>• Tehnici de casting pattern matching	Prelegere curs	2 ore
4.	Capitolul 4. Principii de programare funcțională în Java <ul style="list-style-type: none">• Principii funcționale în programarea orientată obiect• Funcții în Java• Interfețe funcționale și expresii Lambda• Structuri de control funcționale• Tipuri de date și expresii funcționale	Prelegere curs	2 ore



5.	Capitolul 5. Structuri de date fundamentale în Java: Colecții <ul style="list-style-type: none">• Structuri de date pentru liste și secvențe• Cozi și stive (Queues, Stacks)• Seturi simple și ordonate (TreeSets)• Tabele asociative și tabele-hash (Maps and HashMaps)• Structuri complexe: arbori și grafuri (Trees and Graphs)	Prelegere curs	2 ore
6.	Capitolul 6. Procesarea colecțiilor în Java cu fluxuri de tip Stream <ul style="list-style-type: none">• Fluxul de procesare funcțional specific stream-urilor• Operații specifice Stream.API<ul style="list-style-type: none">○ Filtrare○ Mapare○ Sortare○ Jonționare○ Operații de agregare<ul style="list-style-type: none">▪ Map-Reduce▪ Grouping By	Prelegere curs	2 ore
7.	Capitolul 7. Platforma Java: I/O API și lucru cu fișiere <ul style="list-style-type: none">• Fluxuri (stream-uri) I/O<ul style="list-style-type: none">○ InputStream-uri și OutputStream-uri• Consola I/O: Standard Output și Standard Input• Sisteme de fișiere: căi (Paths) și fișiere (Files)<ul style="list-style-type: none">○ Modelul conceptual Java NIO○ Operații cu fișiere	Prelegere curs	2 ore
8.	Capitolul 8. Platforma Java: Conectivitate JDBC baze de date SQL <ul style="list-style-type: none">• Modelul conceptual (componentele) JDBC.API• Scrierea interogărilor JDBC și gestionarea rezultatelor• Operații CRUD cu JDBC.API	Prelegere curs	2 ore
9.	Capitolul 9. Platforma Java: Modelul entităților afacerii cu JPA și JavaBean-Validation <ul style="list-style-type: none">• Principii de bază Domain Driven Design: Delimitarea domeniului aplicației• Persistența în baze de date SQL: JDBC	Prelegere curs	4 ore



	<ul style="list-style-type: none">• Java Persistence API: Mapare OR, Obiecte în tranzacții• Interogări OO cu JPA: JPA Query Language, JPA Criteria API• Validarea datelor în modelul domeniului cu JavaBean API		
10.	Capitolul 10. Platforma Java: Aplicații Web și Desktop <ul style="list-style-type: none">• MVC și arhitectura web• Cadrul de lucru web Vaadin: componente arhitecturale, formulare web, conversia și validarea datelor• MVC și arhitectura JavaFx pentru aplicații desktop• Cadrul de lucru JavaFx: componente arhitecturale, formulare desktop, conversia și validarea datelor	Prelegere curs	4 ore
8.2	Seminar / Laborator	Metode de predare	Observații (ore și referințe bibliografice)
1.	Sintaxa Java. Lucru cu obiecte	Exemplu practic Tema individuala Proiect practic	2
2.	Structuri de date simple. Obiecte complexe. Recursivitate	Exemplu practic Tema individuala Proiect practic	4
3.	Polimorfism structural	Exemplu practic Tema individuala Proiect practic	2
4.	Polimorfism funcțional. Genericitate	Exemplu practic Tema individuala Proiect practic	2
5.	Structuri de date: Java Collections.API	Exemplu practic Tema individuala Proiect practic	4
6.	Programare funcțională în lucru cu colecții: Streams.API	Exemplu practic Tema individuala Proiect practic	2
7.	Structuri de intrare/ieșire și lucru cu fișiere: I/O Streams și Java NIO	Exemplu practic Tema individuala Proiect practic	2
8.	Structuri de integrare SQL: JDBC	Exemplu practic Tema individuala Proiect practic	2
9.	Model și persistența: Java Persistence API (JPA)	Exemplu practic Tema individuala Proiect practic	4
10.	Aplicații Java Desktop și Web	Exemplu practic Tema individuala Proiect practic	4



Bibliografie

Cătălin Strîmbei, *Dezvoltarea aplicațiilor orientate obiect pe platforma Java*, Ed.Univ.Al.I.Cuza Iasi, 2010, ISBN: 978-973-703-609-4
Ștefan Tănasă, Cristian Olaru, Ștefan Andrei, *Java de la 0 la expert*, 2011, ISBN: 978-973-46-2405-8
Herbert Schildt, *Java™ A Beginner's Guide Ninth Edition*, McGraw Hill, 2022
Iuliana Cosmina, *Java 17 for Absolute Beginners*, Apress 2022
Kishori Sharan, *Beginning Java 17 Fundamentals: Object-Oriented Programming in Java 17*, Apress, 2021
Joshua Bloch, *Effective Java Third Edition*, Addison-Wesley 2018
Kathy Sierra, Bert Bates, and Trisha Gee, *Head First Java™*, 3rd Edition, O'Reilly Media, 2022

Referințe suplimentare

Antonio Goncalves, *Beginning Java EE 7*, Apress Manning Publications, 2013
Mike Keith, Merrick Schincariol, Massimo Nardone, *Pro JPA 2 in Java EE 8: An In-Depth Guide to Java Persistence APIs*, Apress, 2018
Alejandro Duarte, *Practical Vaadin Developing Web Applications in Java*, Apress, 2021
Stephen Chin, Johan Vos, James Weaver, Dean Iverson, *The Definitive Guide to Modern Java Clients with JavaFX 17. Cross-Platform Mobile and Cloud Development*, Apress, 2021
Vlad Mihalcea, *High-Performance Java Persistence. Get the most out of your persistence layer*, Lean Publishing, 2020

Eric Evans, *Domain-driven design:tackling complexity in the heart of software*, Addison-Wesley, 2004
Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, *Design Patterns: elements of reusable object-oriented software*, Addison-Wesley professional computing series, 1995
Pierre-Yves Saumont, *Functional Programming in Java: How functional techniques improve your Java programs*, First Edition, Manning, 2017

9. Coroborarea conținutului disciplinei cu așteptările reprezentanților comunității, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere în nota finală (%)
10.4 Curs	Examen	Test grilă	50
10.5 Seminar/ Laborator	EVP	Proiect practic Evaluare activitate laborator	50
10.6 Standard minim de performanță			
EVP(Proiect Practic, Evaluare activitate laborator) >= 5.00 Examen(Test grilă) >= 5.00			

Data completării
20.09.2023

Titular de curs
Conf.dr. CĂTĂLIN STRÎMBEI

Titular de seminar
Conf.dr. Sabina NECULA
Lect.dr. Ionuț HRUBARU

Data avizării în departament

Director de departament
Prof.dr. Mircea Asandului

27.09.2023