FIŞA DISCIPLINEI

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea de Vest din Timișoara
1.2. Facultatea	Fizică și Matematică
1.3. Departamentul	Matematică
1.4. Domeniul de studii	Matematică
1.5. Ciclul de studii	licență
1.6. Programul de studii / calificarea*	Matematică informatică / Inspector de specialitate matematician Cod COR 212003; Referent de specialitate matematician Cod COR 212004; Matematician Cod COR 212009; Consilier statistician Cod COR 212011; Expert statistician Cod COR 212012; Inspector de specialitate statistician Cod COR 212013; Referent de specialitate statistician Cod COR 212014.

2. Date despre disciplină

2.1. Denumirea disciplinei			Programare 2				
2.2. Titularul activităților de curs			Asist. univ. dr. Ciuclea Ioana				
2.3. Titularul activităților de seminar			Asist. univ. dr. Ciuclea Ioana				
2.4. Anul de studii	2	2.5. Semestrul	1	2.6. Tipul de evaluare	E	2.7. Regimul disciplinei	DS/ DOP

3. Timpul total estimat (ore pe semestru al activităților 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 notițe	10
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate	7
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	19
Examinări	4
Tutorat	4

3.7. Total ore studiu individual	44
3.8. Total ore pe semestru	100
3.9. Număr de credite	4

4. Precondiții (acolo unde e cazul)

4.1. de curriculum	Programare I (programare în limbajul C)
4.2. de competențe	Cunoștințe de rezolvare de probleme în limbajul de programare C

5. Condiții (acolo unde e cazul)

5.1. de desfășurare a cursului	Sală de curs cu tablă și videoproiector, computer/laptop cu conexiune la internet/ Google Classroom

5.2. de desfășurare a seminarului/laboratorului	Sală de laborator (calculatoare cu Java instalat) Google Classroom

6. Obiectivele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei

Obiective generale: Familiarizarea cu un limbaj de programare orientat obiect (limbajul Java) si cu paradigmele programării orientate obiect prin realizarea de aplicatii specifice.

Rezultate așteptate ale învățării

Cunoștințe	C4. Cunoașterea și înțelegerea terminologiei specifice; C8. Cunoașterea și înțelegerea normelor generale de etică și deontologie profesio nală, specifice domeniului de studii; C9. Cunoașterea noțiunilor fundamentale de informatică legate de algoritmi și structuri de date, logică și principii de demonstrare, modele și limbaje formale, structuri discrete și modele computaționale; C11. Cunoașterea conceptelor și a metodologiilor privind analiza, proiectarea și implementarea aplicațiilor informatice: etapele unui proces de dezvoltare a unui produs software de la analiză și modelare la testare și validare;
Abilități	A1. Abilitatea de a abstractiza, formaliza și generaliza materialul matematic; A2. Efectuarea rapidă și prescurtată a lanțului de raționamente și operații necesare rezolvării de probleme; A3. Abilitatea de a releva regula și tipul de soluții transferabile în rezolvarea problemelor similare; A5. Capacitatea de a raționa logic și ordonat; A6. Abilitatea de a condensa raționamentele; A7. Capacitatea de restructurare permanentă a experienței anterioare, de a descoperi soluții multiple; A10. Abilitatea de a identifica modele formale/computaționale adecvate, de a utiliza instrumente de modelare și de calcul științific, de a analiza eficiența unui algoritm sau a utilizării unei structuri de date; A12. Abilitatea de a identifica algoritmi și structuri de date adecvate unei probleme concrete, de a aplica principiile de dezvoltare a unei aplicații informatice și de a implementa algoritmi într-un limbaj de programare; A13. Abilitatea de a utiliza medii/instrumente/platforme de programare specifice fiecărei etape din dezvoltarea unui sistem informatic; A14. Abilitatea de a utiliza instrumente informatice pentru gestiunea proiectelor;
Responsabilitate și autonomie	R1. Gestionarea de activități și proiecte complexe, bazate pe cunoștințele și aptitudinile enumerate în timpul formării profesionale și, ulterior, la locul de muncă; R2. Asumarea responsabilității pentru luarea deciziilor în situații imprevizibile, în procesul de formare și, ulterior, la locul de muncă; R3. Asumarea responsabilității pentru propria formare profesională

7. Conținuturi*

7.1. Curs	Metode de predare	Observații
Curs 1. Concepte de baza ale programarii orientate pe obiecte. Primul program Java. Programe si	Prelegere participativă, expunere, problematizare, demonstraţie,	Referințele 1,2,4 - slide-uri curs classroom

modele. De la sursa la executie.	dialog interactiv cu studenţii. Utilizare Google Classroom	
Curs 2. Clase in Java: Componenta unei clase. Crearea si initializarea obiectelor in Java. Reguli de vizibilitate sau Drepturi de acces. Initializarea campurilor unui obiect. Constructori. Membrii statici ai claselor.	Prelegere participativă, expunere, problematizare, demonstraţie, dialog interactiv cu studenţii. Utilizare Google Classroom	Referințe 1,4,5,6 - slide-uri curs classroom
Curs 3. Compararea obiectelor. Compararea stringurilor. Conversii de date. Clase infasuratoare. Mostenirea in Java: Definitii. Exprimarea relatiei de mostenire in Java. Reguli de vizibilitate in contextul relatiei de mostenire. Constructorii si mostenirea. Operatorul instanceof . Pachete in Java. Structurarea programelor Java in pachete. Numele pachetelor si ale claselor dintr-un pachet. Accesul la clasele unui pachet. Accesul la membrii claselor din pachete diferite. Gestionarea fisierelor aferente programelor structurate in pachete. Relatii intre pachete.	Prelegere participativă, expunere, problematizare, demonstraţie, dialog interactiv cu studenţii. Utilizare Google Classroom	Referințele 1,3,5 - slide-uri curs classroom
Curs 4. Interfete in Java: Interfata obiectelor. Interfetele in limbajul Java. Implementarea obiectelor. Asocierea operatiilor cu obiectele. Relatii intre interfete si clase.	Prelegere participativă, expunere, problematizare, demonstraţie, dialog interactiv cu studenţii. Utilizare Google Classroom	Referințe 3,5,6 - slide-uri curs classroom
Curs 5. Tratarea exceptiilor in Java: Metode clasice de tratare a exceptiilor. Mecanismul de emitere-captare a excptiilor. Instructiunea throw. Blocurile try-catch. Secventa finally. Executia unui bloc try-catch-finally. Exceptii predefinite ale limbajului Java.	Prelegere participativă, expunere, problematizare, demonstraţie, dialog interactiv cu studenţii. Utilizare Google Classroom	Referințele 3,5,6 - slide-uri curs classroom
Curs 6-7. Lucrul cu colectii de obiecte in Java: Generalitati. Interfata Collection. Interfata List. Interfata Set. Interfata Map. Iteratorii.	Prelegere participativă, expunere, problematizare, demonstraţie, dialog interactiv cu studenţii. Utilizare Google Classroom	Referințele 3,5 - slide-uri curs classroom
Curs 8. Interfete grafice in Java: Introducere. Crearea ferestrei unei	Prelegere participativă, expunere, problematizare, demonstraţie,	Referințele 3,5,6

aplicatii grafice. Elemente de baza privind tratarea evenimentelor in Java.	dialog interactiv cu studenţii. Utilizare Google Classroom	- slide-uri curs classroom
Curs 9 - 10. Interfete grafice in Java: Pachetul Swing.	Prelegere participativă, expunere, problematizare, demonstraţie, dialog interactiv cu studenţii. Utilizare Google Classroom .	Referinţele 3,5,6 - slide-uri curs classroom
Curs 11 – 12. Fluxuri I/O. Serializarea datelor.	Prelegere participativă, expunere, problematizare, demonstraţie, dialog interactiv cu studenţii. Utilizare Google Classroom	Referinţele 3,5 - slide-uri curs classroom
Curs 13. Conectarea la baze de date prin limbajul Java.	Prelegere participativă, expunere, problematizare, demonstraţie, dialog interactiv cu studenţii. Utilizare Google Classroom	Referinţe 3,4,5,6. - slide-uri curs classroom
Curs 14. Recapitulare examen.	Prelegere participativă, expunere, problematizare, demonstraţie, dialog interactiv cu studenţii. Utilizare Google Classroom	

Bibliografie

1. K. Arnold, J. Gosling - "The Java Programming Language. Second Edition", Addison-Wesley, 1997
2. D. Arnow, G. Weiss - "Introduction to Programming Using Java. An Object-Oriented Approach", Addison-Wesley, 1998
3. Ştefan Tănasă – Java de la 0 la expert, Ed. Polirom, Iaşi, 2005
4. Horia Georgescu – Introducere in universul Java, Ed Tehnică, 2002
5. Cristian Frăsinaru – Curs practic de Java, Ed. MatrixRom, 2005
6. Victoria Iordan, Flavia Micotă – Introducere în Java, Ed. Eurostampa, 2010

7.2. Seminar/laborator	Metode de predare/ invăţare	Observaţii
1. Scrierea unor aplicaţii simple Java cu Eclipse, compilare, execuţie (pentru a prezenta structura unui program).	Problematizare, dialog interactiv cu studenţii, învăţare prin colaborare	Studenţii au acces la sinteza aferentă tematicii de laborator şi la enunţurile problemelor recomandate spre rezolvare Cadrul didactic oferă detalii suplimentare, răspunde întrebărilor studenţilor şi verifică/ evaluează modul în care studenţii au rezolvat problemele.
2. Operaţii cu tablouri. Folosirea metodelor pentru obiecte din clasele String şi StringBuffer. Scrierea unor programe simple Java în care datele de intrare sunt preluate fie din linia de comandă,	Problematizare, dialog interactiv cu studenţii, învăţare prin colaborare	idem

fie generate aleator.		
3. Folosirea modificatorilor unei clase (public, abstract, final). Folosirea modificatorilor de acces în declaraţiile membrilor unei clase	Problematizare, dialog interactiv cu studenţii, învățare prin colaborare	idem
4. Însuşirea fenomenului de moştenire în Java prin crearea claselor derivate. Folosirea claselor abstracte.	Problematizare, dialog interactiv cu studenţii, învățare prin colaborare	idem
5. Utilizarea interfeţelor.	Problematizare, dialog interactiv cu studenţii, învățare prin colaborare	idem
6. Utilizarea colecţiilor.	Problematizare, dialog interactiv cu studenţii, învățare prin colaborare	idem
7. Utilizarea colecţiilor.	Problematizare, dialog interactiv cu studenţii, învățare prin colaborare	idem
8. Tratarea excepțiilor.	Problematizare, dialog interactiv cu studenţii, învățare prin colaborare	idem
9. Lucrul cu fișieire, fluxuri de date.	Problematizare, dialog interactiv cu studenţii, învățare prin colaborare	idem
10. Serializarea obiectelor	Problematizare, dialog interactiv cu studenţii, învățare prin colaborare	idem
11. Proiectarea interfețelor grafice.	Problematizare, dialog interactiv cu studenţii, învățare prin colaborare	idem
12. Proiectarea interfețelor grafice.	Problematizare, dialog interactiv cu studenţii, învățare prin colaborare	idem
13. Prezentare proiecte	Problematizare, dialog interactiv cu studenţii, învățare prin colaborare	idem
14. Prezentare proiecte	Problematizare, dialog interactiv cu studenţii	idem
Bibliografie 1. Idem curs		

8. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului

Conţinutul este în concordanță cu structura cursurilor similare de la alte universități și acoperă aspectele fundamentale necesare familiarizării cu problematica programării orientate obiect. Abilitatea de a identifica, proiecta, implementa și analiza probleme care se pot rezolva folosind principii orientate obiect este esențială pentru orice activitate din domeniul informaticii. Competențele oferite de această disciplină sunt necesare unui specialist IT pentru a identifica soluții de rezolvare a unor probleme concrete, indiferent de domeniul specific de activitate.

9. Utilizarea instrumentelor bazate pe inteligența artificială generativă

Utilizarea IIAgen nu este permisă pentru realizarea sarcinilor definite la secțiunea pentru evaluare pentru curs. Utilizarea IIAgen este permisă pentru realizarea sarcinilor definite la secțiunea de evaluare pentru laborator doar pentru generarea de idei, clarificarea conceptelor, verificarea sintaxei și debugging. Nu este permisă utilizarea IIAgen pentru generarea de soluții complete pentru sarcinile de lucru din cadrul laboratoarelor.

10. Evaluare*

Tip de activitate	9.1. Criterii de evaluare**	9.2. Metode de evaluare***	9.3. Pondere din nota finală
10.4. Curs	Însușirea cunoștințelor despre terminologia programării orientate obiect și caracteristici ale limbajului Java. Identificarea de: stucturi de date abstracte, metode de rezolvare a diferitelor prelucrări asupra structurilor de date identificate.	Examen scris	50%
10.5. Seminar/laborator	Capacitatea de a proiecta o aplicație după anumite criterii date, operând cu date conținute într-un fișier binar și comunicând cu utilizatorul prin intermediul unei interfețe grafice.	Proiect individual (prezentare săptămâna 13 - 14)	50%
10.6 Standard minim de performanță Nota finală se calculează ca medie ponderată a notelor acordate pentru componentele specificate la 10.4 și 10.5. Examenul se consideră promovat dacă media este cel puțin 5 . Pentru promovarea activității de laborator și participarea la examenul scris este necesară realizarea cu succes a 6 lucrări de laborator.			

Data completării
15.09.2026

Titular de disciplină (curs)
Asist. univ. dr. Ciuclea Ioana

Titular de disciplină (laborator)
Asist. univ. dr. Ciuclea Ioana

Data avizării departament
18.09.2026

Director departament
Prof. univ. dr. Bogdan Sasu