

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	UNIVERSITATEA "LUCIAN BLAGA" DIN SIBIU
1.2 Facultatea / Departamentul	DE ȘTIINȚE - DEP. MATEMATICĂ ȘI INFORMATICĂ
1.3 Catedra	
1.4 Domeniul de studii	INFORMATICĂ
1.5 Ciclul de studii	LICENȚĂ
1.6 Programul de studii/Calificarea	INFORMATICĂ

2. Date despre disciplină

2.1 Denumirea disciplinei				Metode avansate de programare			
2.2 Titularul activităților de curs				Lector univ. Ralf Fabian			
2.3 Titularul activităților de seminar				Lector univ. Ralf Fabian			
2.4 Anul de studiu	II	2.5 Semestrul	4	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei	0

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: 3.2 curs	28	3.3 seminar/laborator	28
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					26
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					18
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					30
Tutoriat					5
Examinări					5
Alte activități					
3.7 Total ore studiu individual					84
3.9 Total ore pe semestru					56
3.10 Numărul de credite					5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	•
4.2 de competențe	•

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	<ul style="list-style-type: none"> ● sală de curs cu videoproiector
5.2. de desfășurare a seminarului/laboratorului	<ul style="list-style-type: none"> ● laborator informatică cu videoproiector

6. Competențele specifice acumulate

Competențe profesionale	<ul style="list-style-type: none"> ● Cunoașterea și utilizarea adecvată a noțiunilor teoretice fundamentale ale limbajului de programare Java și a mediului de programare specific. ● Abilitatea de a transpune concepte teoretice în implementări practice. ● Potențialul de a valorifica la nivel de înțelegere competențele dobândite prin această disciplină. ● Capacitatea de a explica modul de construcție a unei aplicații Java. ● Capacitatea de a interpreta rezultatele obținute. ● Explicarea conceptelor și noțiunilor însușite. ● Folosirea deprinderilor dobândite în punerea și rezolvarea de probleme practice. ● Aptitudini de prezentare și promovare a propriilor rezultate obținute.
Competențe transversale	<ul style="list-style-type: none"> ● Utilizarea tehnicii de calcul pentru rezolvarea de probleme practice. ● Capacitatea de a implementa algoritmi în limbajul Java. ● Capacitatea de a proiecta și realiza aplicații complexe care utilizează elemente de interfață grafică. ● Capacitatea de a utiliza, înțelege și modifica codul sursă a aplicațiile deja implementate, conform unor cerințe date. ● Aprofundarea cunoștințelor de proiectare și programare orientată obiect. Însușirea de deprinderi în manevrarea mediilor integrate de dezvoltare. ● Perspective noi privind conceptele deja cunoscute. ● Lucrul în echipă. ● Respectarea cu atenție cerințelor și a termenilor pentru finalizare și evaluare a proiectelor. ● Dezvoltarea atitudinii pozitive față de muncă și responsabilitate pentru propria pregătire profesională.

7. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

7.1 Obiectivul general al disciplinei	<ol style="list-style-type: none"> a) Inițierea în programarea orientată obiect cu limbajul Java; b) Însușirea noțiunilor generale despre platforma Java. c) Familiarizarea studenților cu arhitectura aplicațiilor bazate pe limbajul Java. d) Prezentarea terminologie și a conceptelor specifice limbajului pentru aplicații desktop, web, și mobile; e) Familiarizarea studenților cu noțiunile necesare activităților de laborator;
7.2 Obiectivele specifice	<ol style="list-style-type: none"> f) Însușirea cunoștințelor generale pentru programarea în mediul Eclipse/Netbeans

	<p>a) Fixarea deprinderilor de realizare a unei aplicații software funcționale cu limbajul Java, prin parcurgerea tuturor etapelor necesare și reflectarea lor într-o documentație completă.</p> <p>b) Aprofundarea cunoștințelor de programare ale studenților.</p> <p>c) Formarea unor deprinderi de lucru pentru realizarea de aplicații folosind mediul de programare Eclipse/Netbeans.</p> <p>d) Utilizarea corectă a noțiunilor de programare pe obiecte, și identificarea cazurilor când o problemă poate fi generalizată sau soluția poate fi eficientizată.</p> <p>e) Încurajarea studiului individual și al lucrului în echipă, prin prezentarea unui proiect de semestru.</p>
--	--

8. Conținuturi

8.1 Curs	Metode de predare	Observații
Introducere. Organizarea sistemelor de calcul. Istoric C/C++/Java. Medii de dezvoltare pentru Java. Platforma de dezvoltare. Pachete și componente principale.	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Introducere în aplicații Java. Lexicul limbajului. Tipuri de date primitive. Instrucțiuni de control. Vectori, clasa Arrays	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Clase și obiecte în Java. Organizarea claselor în pachete și archive. Generarea automată a documentației	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Interfețe în Java. Structuri de date. Colecții de obiecte. Genericitate.	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Tratarea excepțiilor. Fluxuri de date. Serializarea obiectelor. Filtrare pe fluxuri de date. Lucrul cu fișiere.	expunerea, explicatia, conversatie frontala, problematizare,	

	modelare	
Interfața grafică în Java. Gestionare de poziționare a componentelor. Caracteristici AWT. Tratarea evenimentelor de interfață. Tipuri de evenimente și ascultătoare (Listener). Tehnici de tratare.	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Interfața grafică cu API ul Swing. Componente simple: Ferestre, panouri, etichete, butoane, margini.	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Componente SWING avansate: JList, JComboBox, JTable, JTree, JSpinner, JMenue, JFileChooser, JOptionPane etc.	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Grafică cu Swing. Desenarea componentelor. Suprafața de desenare. Contextul grafic. Desenarea de text și figuri geometrice 2D. Fonturi și Culori. Utilizarea imaginilor	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Fire de execuție în Java. Realizarea animațiilor grafice.	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Lucrul cu baze de date relaționale prin JDBC. Persistența datelor cu tehnologia Hibernate.	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Programare în rețea. Aplicații client/server, P2P	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Tehnologii Java pentru dezvoltarea aplicațiilor web. Applet uri	expunerea, explicatia, conversatie frontala, problematizare, modelare	

Server de aplicații. Servlets. Java Server Pages. Java Server Faces	expunerea, explicatia, conversatie frontala, problematizare, modelare	
Bibliografie		
<ol style="list-style-type: none"> 1. Documentația oficială de la Oracle http://docs.oracle.com/javase/7/docs/ 2. Tutoriale oficiale de la Oracle: http://docs.oracle.com/javase/tutorial/ 3. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, The Java Language Specification, Third Edition, http://docs.oracle.com/javase/specs/ 4. H. M. Deitel, P. J. Deitel - <i>Java How to Program, Ninth Edition</i>, Editura Prentice Hall 2011. 5. Allen B. Downey, <i>Think Java, How to Think Like a Computer Scientist</i>, 2012, http://www.greenteapress.com/thinkapjava/ 6. David J. Eck, <i>Introduction to Programming Using Java</i>, Sixth Edition, http://math.hws.edu/javanotes/ 7. Cristian Frăsinaru - <i>Curs Practic de Java</i>, Editura MatrixRom 2007. 		
8.2 Seminar/laborator	Metode de predare	Observații
Recapitulare și aprofundarea de concepte și noțiuni fundamentale ale programării orientate obiect. Familiarizarea cu mediul de dezvoltare. Stabilirea echipelor pentru temele de implementare. Împărțirea proiectelor. Elaborarea de aplicații simple Java. Familiarizarea cu modul de compilare și executare a unei aplicații Java	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Definirea și implementarea claselor în Java. Familiarizarea studenților cu moștenirea în Java. Prezentarea principalelor pachete JAVA SDK. Aplicații folosind clasele și moștenirea în Java	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Lucrul cu Interfețe în Java. Prezentarea modului de lucru cu interfețe. Aplicații folosind interfețe și structuri de date implementate cu ajutorul colecțiilor de obiecte.	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	

Prezentarea modului de lucru cu excepții în Java. Aplicații cu fluxuri de date și tratarea excepțiilor.	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Familiarizarea studenților cu tehnica de serializare. Aplicații cu serializarea obiectelor simple. Accesarea fișierelor și aplicații cu fișiere în JAVA	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Utilizarea pachetelor grafice. Realizarea de aplicații cu componente grafice și exemplificarea utilizării gestionarilor de poziționare.	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Realizarea de aplicații cu componente grafice și cu tratarea evenimentelor de interfață. Familiarizarea cu noțiunile de Listener și tehnicile de tratare a evenimentelor	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Prezentarea particularităților componentelor SWING. Realizarea unei aplicații grafice complexe care să utilizeze componentele SWING și celelalte tehnici învățate (tratarea evenimentelor, tratarea excepțiilor, etc.)	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Utilizare clasei Thread în Java. Definirea, implementarea și rularea grupurilor de fire de execuție în JAVA. Tehnici de sincronizarea a firelor de execuție în Java	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Prezentarea obiectelor de lucru cu baze de date JDBC. Aplicații în care se	problematizarea, metoda	

exemplifică conectarea la o baza de date și accesarea datelor	exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Utilizarea componentelor pachetului java.net. Dezvoltarea de aplicații client server care implementează comunicarea în rețea folosind socketuri, TCP/IP și Datagramme.	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Dezvoltarea de Applet-uri pentru animații grafice. Adaptarea Applet-urilor pentru a putea fi rulate și ca programe normale și viceversa.	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Configurarea unui server de aplicații. Crearea și rularea unui Servlet care permite accesul la o bază de date. Transmiterea de imagini de la un Servlet.	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
Dezvoltarea unei pagini web cu JSP și comparativ cu JSF.	problematizarea, metoda exercitiului, fise de lucru, conversatie frontala, conversatie individuala	
<p>Bibliografie</p> <ol style="list-style-type: none"> 1. Documentația oficială de la Oracle http://docs.oracle.com/javase/7/docs/ 2. Tutoriale oficiale de la Oracle: http://docs.oracle.com/javase/tutorial/ 3. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, The Java Language Specification, Third Edition, http://docs.oracle.com/javase/specs/ 4. H. M. Deitel, P. J. Deitel – <i>Java How to Program, Ninth Edition</i>, Editura Prentice Hall 2011. 5. Allen B. Downey, <i>Think Java, How to Think Like a Computer Scientist</i>, 2012, http://www.greenteapress.com/thinkapjava/ 6. David J. Eck, <i>Introduction to Programming Using Java</i>, Sixth 		

Edition, <http://math.hws.edu/javanotes/>

7. Cristian Frăsinaru - *Curs Practic de Java*, Editura MatrixRom 2007.

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

• Conținutul disciplinei a fost stabilit ținând cont de interacțiunile constructive ale cadrelor didactice, studenților și a reprezentanților din mediul economic, științific, în cadrul manifestărilor științifice, întâlnirilor de lucru și activităților de practică și dezvoltare de proiecte a studenților.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs		Proba practică	40%
10.5 Seminar/laborator		Susținerea proiectului de semestru	60%
10.6 Standard minim de performanță			
<ul style="list-style-type: none">- Pentru predare se va folosi prelegerea, dezbateră, învățarea prin cooperare, alternând expunerea și explicațiile cu întrebările, exemplificarea numerică, vizuală și exercițiul; dezbateră și conversația profesor - student și student - student.- mijloace multimedia: videoproiector- stil de predare interactiv: alternarea mijloacelor multimedia cu mijloacele clasice (tabla de scris); utilizarea de animații/vizualizări în cadrul predării, pentru ilustrarea funcționării metodelor/algoritmilor prezentați (instruire asistată de calculator);- lucrări de laborator disponibile studenților, la cerere pe suport magnetic.- Sunt valabile regulamentele oficiale ale universității în legătură cu prezenta studenților la activitățile didactice și cu cazurile de copiat și plagiat.- Promovarea examenului este condiționată de predarea completă a lucrărilor de proiect.- Prezenta la orele de laborator este obligatorie.			

Data completării
27.09.2014

Semnătura titularului de curs
.....

Semnătura titularului de seminar
.....

Data avizării în catedră
28.09.2016

Semnătura directorului de departament
prof.univ.dr. Mugur Acu