

## FISA DISCIPLINEI

### PROGRAMARE JAVA ȘI SOFTWARE MATEMATIC (P.J.S.M.)

#### 1. Date despre program

1.1 Instituția de învățământ superior	Universitatea POLITEHNICA din București
1.2 Facultatea	Facultatea de Stiinte Aplicate
1.3 Departamentul	Departamentul de Metode si Modele Matematice
1.4 Domeniul de studii	Stiinte Ingineresti Aplicate
1.5 Ciclul de studii	MASTER
1.6 Programul de studii/Calificarea	TCSI/Specialist SIG/IT COD.COR 252901

#### 2. Date despre disciplină

2.1 Denumirea disciplinei				Programare Java și software matematic (P.J.S.M.)			
2.2 Titularul activităților de curs				dr. Irina Badralexi			
2.3 Titularul activităților de seminar				dr. Irina Badralexi			
2.4 Anul de studiu	I	2.5 Semestrul	II	2.6 Tipul de evaluare	Examen	2.7 Regimul disciplinei	Obligatoriu

#### 3. Timpul total estimat (ore pe semestru al activităților didactice)

3.1 Număr de ore pe săptămână din care	1	3.2 curs	1	3.3 seminar/laborator	1/2
3.4 Total ore din planul de învățământ din care	56	3.5 curs	14	3.6 seminar/laborator	14/28
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					30
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate					5
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					10
Tutoriat					0
Examinări					3
Alte activități					0
3.7 Total ore studiu individual		48			
3.9 Total ore pe semestru		104			
3.10 Numărul de credite		4			

#### 4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Nu este cazul
4.2 de competențe	Programarea calculatoarelor Structuri de date și algoritmi

#### 5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	Nu este cazul
5.2 de desfășurare a seminarului	Prezența obligatorie la seminarii

### 6. Competențe specifice acumulate

Competențe profesionale	<ul style="list-style-type: none"> <li>- Dezvoltarea capacității de a folosi în domeniul cercetării informatice cunoștințe de matematică modernă;</li> <li>- Dezvoltarea abilităților de folosire a sistemelor de operare UNIX și LINUX;</li> <li>- Dezvoltarea capacității de a proiecta și administra rețele de calculatoare;</li> <li>- Dezvoltarea capacității de a proteja serverele de atacurile informatice și de a asigura un trafic informațional securizat;</li> <li>- Dezvoltarea capacității de lucru în echipă;</li> <li>- Dezvoltarea capacității de cercetare științifică;</li> <li>- Disciplina își propune crearea abilităților de a aplica cunoștințele generale privind orientarea spre obiecte și limbajul Java pentru a dezvolta aplicații și sisteme software, pornind de la specificarea cerințelor și până la executia, depanarea și interpretarea rezultatelor.</li> </ul>
Competențe transversale	Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesiei.

### 7. Obiectivele disciplinei (reieșind din grila de competențe specifice acumulate)

7.1 Obiectivul general al disciplinei	Insusirea principiilor de baza ale programarii orientate spre obiecte (OO) exemplificate pentru limbajul Java, și modul lor de aplicare pentru dezvoltarea sistemelor software. Dezvoltarea de aplicații utilizând software matematic (Matlab, Maple).
7.2 Obiective specifice	<p><b>1. Cunoștințe teoretice - Cunoaștere și înțelegere:</b> Cunoașterea principiilor de baza ale programarii orientate pe obiecte și a tipurilor fundamentale de diagrame UML.</p> <p><b>2. Deprinderi dobândite - Explicare și interpretare:</b> Formarea unei gândiri algoritmice și a pașilor necesari scrierii unui program. Formarea deprinderilor necesare dezvoltării, compilării și executiei programelor Java folosind command-line cât și un IDE (Eclipse sau IntelliJ IDEA). Gestionarea excepțiilor.</p> <p><b>3. Abilități dobândite - Instrumental-aplicative:</b> Înțelegerea unui cod Java sau a unui script Matlab. Transpunerea unor probleme simple în cod Java și/sau cu ajutorul unui software matematic.</p> <p><b>4. Atitudinale:</b> Capacitatea de lucru în echipă pentru rezolvarea unor probleme practice. Responsabilitate și corectitudine în activitățile desfășurate.</p>

## 8. Conținuturi

8.1 Curs	Metode de predare	Observații
<b>1. Introducere in abordarea orientata spre obiecte (OO)</b> 1.1. Obiectul cursului si relatia cu alte cursuri 1.2. Scurt istoric al programarii. Evolutia catre abordarea OO 1.3. Caracteristicile si principiile abordarii OO	Predarea se bazează pe folosirea videoproietorului (acoperind funcția de comunicare și demonstrativă); metodele de comunicare orală utilizată sunt metoda expositivă și metoda problematizării, utilizate frontal. Materialele de curs sunt: notele și prezentările de curs, culegeri de probleme propuse (teoretice și cu rezolvare pe calculator). Toate materialele sunt disponibile în format electronic, prin situl cursului.	1 ore
<b>2. Orientarea spre obiecte in limbajul Java</b> 2.1. Elementele de baza. Tipurile de date referinta 2.2. Obiecte si clase. Metode (operatii) si campuri (attribute). 2.3. Particularitati Java. Cateva clase de biblioteca de uz general 2.4. Clase si relatii intre clase. Asociere, delegare, agregare, compunere 2.5. Generalizare, specializare si mostenire 2.6. Clase abstracte si interfete Java 2.7. Polimorfismul metodelor		4 ore
<b>3. Proiectarea aplicatiilor in limbajul Java</b> 4.1. Proiectarea arhitecturala si de detaliu a sistemelor software 4.2. Criterii de alegere a arhitecturii sistemelor software 4.3. Organizarea iterativa a proiectelor software 4.4. Proiectarea in vederea reutilizarii. 4.5. Framework-uri si componente reutilizabile in sistemele software 4.6. Proiectarea interfetelor grafice pentru utilizabilitate sporita.		4 ore
4. Prezentarea functiilor si comenzilor de baza in Matlab si a aplicatiilor de tip TOOLBOX		2 ore
5. Rezolvarea simbolică a unor ecuații și sisteme diferențiale		1 ore
6. Reprezentare grafica 2D si 3D. Manipularea obiectelor grafice in Matlab.		2 ore

**Bibliografie**

1. E. Popovici, “Note de curs” in format electronic, <http://www.elcom.pub.ro/discipline/POO-Java> si <http://www.elcom.pub.ro/discipline/isc>.
2. T. Radulescu, “Ingineria software orientata pe obiecte”, Editura Matrix Rom, București, 2000.
3. E. Popovici, “Indrumar de laborator” in format electronic, <http://www.elcom.pub.ro/discipline/POO-Java> si <http://www.elcom.pub.ro/discipline/isc>.
4. Gorunescu M., 2006, Calculând cu imagini în Matlab, Editura Albastră Cluj-Napoca;
5. Green P, Rosenberg J., 2000, Calculus with Matlab, [http:// www. math.umd.edu/- jmr/24l/calc.htm](http://www.math.umd.edu/~jmr/24l/calc.htm).
6. [http://www.edumanager.ro/community/documente/modelare\\_matematica\\_prin\\_matlab.pdf](http://www.edumanager.ro/community/documente/modelare_matematica_prin_matlab.pdf)

8.2 Seminar/Laborator	Metode de predare	Observații
1. Obiecte si clase – abstractizare, operatii, attribute, instantiere, comparatii	Predarea se bazează pe folosirea videoproiectorului (acoperind funcția de comunicare și demonstrativă); metoda de comunicare orală utilizată este metoda problematizării, utilizate frontal. Materialele didactice sunt postate pe platforma educațională a UPB.	1 ora/ 2 ore
2. Relatii între clase – asociere, delegarea responsabilitatilor, agregare		2 ore /4 ore
3. Generalizare si specializare – mostenire, extindere, polimorfism		2 ore /4 ore
4. Dezvoltarea aplicatiilor Java si ingineria inversa UML. Modelarea statica. Tipuri fundamentale de diagrame UML		2 ore / 4 ore
5. Modelarea dinamica. Crearea diagramelor UML de comunicare si MSC		1 ore / 2 ore
6. Dezvoltarea documentatiei JavaDoc		1 ore / 2 ore
7. Crearea, analiza și implementarea de algoritmi pentru rezolvarea unor probleme de mecanică în Matlab		2 ore / 4 ore
8. Utilizarea instrumentelor de desenare pentru editarea graficelor.		2 ore / 4 ore
9. Predarea temelor de casa. Verificare laborator		1 ora / 2 ore

**Bibliografie**

1. E. Popovici, “Note de curs” in format electronic, <http://www.elcom.pub.ro/discipline/POO-Java> si <http://www.elcom.pub.ro/discipline/isc>.
2. T. Radulescu, “Ingineria software orientata pe obiecte”, Editura Matrix Rom, București, 2000.
3. E. Popovici, “Indrumar de laborator” in format electronic, <http://www.elcom.pub.ro/discipline/POO-Java> si <http://www.elcom.pub.ro/discipline/isc>.
4. Gorunescu M., 2006, Calculând cu imagini în Matlab, Editura Albastră Cluj-Napoca;
5. Green P, Rosenberg J., 2000, Calculus with Matlab, [http:// www. math.umd.edu/- jmr/24l/calc.htm](http://www.math.umd.edu/~jmr/24l/calc.htm).

6. [http://www.edumanager.ro/community/documente/modelare\\_matematica\\_prin\\_matlab.pdf](http://www.edumanager.ro/community/documente/modelare_matematica_prin_matlab.pdf)

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

Cursul are ca obiectiv însușirea principiilor de bază ale programării orientate spre obiecte (OO) exemplificate pentru limbajul Java, și modul lor de aplicare pentru dezvoltarea sistemelor software.

**10. Evaluare**

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere în nota finală
10.4 Curs	-cunoașterea noțiunilor teoretice fundamentale; - cunoașterea modului de aplicare a teoriei la probleme specifice; - analiza diferențială a tehnicilor și metodelor teoretice.	Examen parțial, cu posibilitate de degrevare a materiei, pondere 33%.  Examen final, pondere 33%	66%
10.5 Seminar/Laborator	- cunoașterea aplicării, pe exemple concrete a elementelor teoretice exemplificate în cadrul cursului.	Notare în timpul semestrului, teme de casă.	33%
10.6 Standard minim de performanță			
Cunoașterea noțiunilor teoretice de bază prezentate la curs și rezolvarea unor probleme tip.			

Data completării  
**01.09.2018**

Semnătura titularului de curs  
dr. Irina Badralexi

Semnătura titularului de aplicații  
dr. Irina Badralexi

Data avizării în departament  
.....

Semnătura sefului de departament  
prof. dr. Mircea Olteanu

Responsabil program master  
lector dr. Emil Simion