

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Inginerie Electrică și Știința Calculatoarelor
1.3 Departamentul	Automatică și Tehnologia Informației
1.4 Domeniul de studii de licență ¹⁾	Mecatronica și robotică
1.5 Ciclu de studii ²⁾	Licență
1.6 Programul de studii/ Calificarea	Robotică

2. Date despre disciplină

2.1 Denumirea disciplinei	Programarea calculatoarelor și limbaje de programare II							
2.2 Titularul activităților de curs	Conf. dr. ing. Dominic Mircea KRISTÁLY							
2.3 Titularul activităților de seminar/ laborator/ proiect	dr. ing. Cornelia RĂȘNOVEANU							
2.4 Anul de studiu	1	2.5 Semestrul	2	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei	Conținut ³⁾	DF
							Obligativitate ³⁾	DI

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/ proiect	0/2/0
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar/ laborator/ proiect	0/28/0
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					19
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					8
Pregătire seminarii/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					8
Tutoriat					8
Examinări					1
Alte activități					-
3.7 Total ore studiu individual	44				
3.8 Total ore pe semestru	100				
3.9 Numărul de credite⁵⁾	4				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	<ul style="list-style-type: none"> Parcurgerea cursurilor: <i>Programarea calculatoarelor și limbaje de programare I</i>
4.2 de competențe	<ul style="list-style-type: none"> C1. Aplicarea cunoștințelor fundamentale de cultură tehnică generală și de specialitate pentru rezolvarea problemelor tehnice specifice domeniului.

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	<ul style="list-style-type: none"> tablă videoproiector
5.2 de desfășurare a seminarului/ laboratorului/ proiectului	<ul style="list-style-type: none"> rețea de calculatoare programe specializate îndrumar de laborator

6. Competențe specifice acumulate

Competențe profesionale	<ul style="list-style-type: none"> C1. Aplicarea cunoștințelor fundamentale de cultură tehnică generală și de specialitate pentru rezolvarea problemelor tehnice specifice domeniului.
Competențe transversale	<ul style="list-style-type: none">

7. Obiectivele disciplinei (reieșind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> • Disciplina își propune să inițieze studenții în programarea calculatoarelor în limbajul Java și în programarea orientată pe obiecte.
7.2 Obiectivele specifice	<ul style="list-style-type: none"> • Cunoașterea modului de funcționare a programelor Java în contextul portabilității. • Cunoașterea sintaxei limbajului Java. • Asimilarea conceptelor specifice programării orientată pe obiecte (clase, obiecte, moștenire, polimorfism). • Crearea de abilități de programare a aplicațiilor Java. • Utilizarea corectă a termenilor specifici programării orientate pe obiecte și ai limbajului de programare Java.

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
1. Introducere în limbajul Java. a. Caracteristicile limbajului Java. b. Crearea unei aplicații simple. c. Structura lexicală a limbajului Java. d. Tipuri de date.	prezentare interactivă cu videoproiector prelegere clasică explicație problematizare	4 ore	
2. Obiecte și clase. a. Crearea claselor. b. Ciclul de viață al unui obiect.		6 ore	
3. Organizarea fișierelor. a. Organizarea fișierelor .java și .class. b. Arhive JAR.		2 ore	
4. Excepții. a. Ce sunt excepțiile. b. Excepții la execuție.		3 ore	
5. Fluxuri (Intrări/Ieșiri). a. Clasificarea fluxurilor. b. Folosirea fluxurilor. c. Fluxuri pentru lucrul cu fișiere.		3 ore	
6. Interfețe. a. Definierea și implementarea unei interfețe. b. Moștenirea multiplă prin intermediul interfețelor.		3 ore	
7. Pachete. a. Crearea unui pachet. b. Importul unei clase sau interfețe.		1 ore	
8. Colecții. a. Implementări ale colecțiilor.		2 ore	
9. Serializarea. a. Serializarea obiectelor. b. Controlul serializării.		2 ore	
10. Interfața grafică. a. Componente SWING. b. Tratarea evenimentelor. c. Folosirea ferestrelor. d. Folosirea componentelor.		2 ore	
Bibliografie [1] Frăsinaru, C. – <i>Curs practic de Java</i> , Ediție electronică, 2008. [2] Horstmann, C.S., Cornell, G. – <i>Core Java 2-Fundamentals</i> , vol. 1, Ed. Sun Microsystem Press, 2001. [3] Danciu, D., Mardale, G. – <i>Arta programării în JAVA. Elemente-suport fundamentale</i> , Editura Albastră, 2004. [4] Eckel, B. – <i>Thinking in Java. Fourth Edition</i> , Prentice Hall, 2006.			
8.2 Seminar/ laborator/ proiect	Metode de predare- învățare	Număr de ore	Observații
Crearea și rularea programelor Java din linie de comandă.	conversație exerciții evaluare	2 ore	
Crearea și rularea programelor Java cu ajutorul platformei Eclipse.		2 ore	
Structura unui program Java.		2 ore	
Tipuri de date primitive.		2 ore	
Citirea de la tastatură.		2 ore	
Structuri de control.		2 ore	
Tablouri.		2 ore	

Șiruri de caractere.		2 ore	
Clase și programare orientată obiect.		2 ore	
Moștenirea.		2 ore	
Tratarea excepțiilor.		2 ore	
Interfețe.		2 ore	
Fluxuri de intrare / ieșiere.		2 ore	
Interfețe grafice.		2 ore	
Bibliografie [1] Kristaly, D.M., Vulpe, A. – <i>Limbajul de programare Java. Îndrumar de laborator</i> , Ed. Universității Transilvania din Brașov, 2009.			

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

Disciplina, prin conținutul oferit studenților, aparține domeniului tehnologiei informației și are o abordare orientată spre practică, pentru a crea studenților acele competențe în programare cerute de piața muncii.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Corectitudinea soluției oferite Respectarea regulilor de editare și întocmire a unui program	Evaluare prin examen practic: – rezolvare de probleme; biletele conțin 1 subiect cu mai multe subpuncte. Pentru fiecare subiect se specifică baremul de notare, care se comunică studenților odată cu subiectele.	65%
	Prezența la curs	Se constată pe parcursul semestrului	5%
10.5 Seminar/ laborator/ proiect	Aplicarea corectă a metodei de rezolvare pentru problema data	Evaluare pe parcurs. Evaluare prin probă practică – colocviu de laborator.	30%
	Corectitudinea soluției		
	Exemplificarea rulării programelor		
10.6 Standard minim de performanță			
<ul style="list-style-type: none"> Crearea structurii de bază a programelor și a structurilor de date necesare. 			

Data completării

07.10.2016

Semnătura titularului de curs,
Conf. dr. ing. Dominic Mircea KRISTÁLY

Semnătura titularului de seminar/
laborator/ proiect

Data avizării în departament

.....

.....
Semnătura directorului de departament,
Prof. univ. dr. ing. Sorin Aurel MORARU

.....

Notă:

- 1) Domeniul de studii - *se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor și al specializărilor/ programelor de studii universitare în vigoare)* ;
- 2) Ciclul de studii - *se alege una din variantele: Licență/ Master/ Doctorat;*
- 3) Regimul disciplinei (conținut) - *se alege una din variantele: DF (disciplină fundamentală)/ DD (disciplină din domeniu)/ DS (disciplină de specialitate)/ DC (disciplină complementară) - pentru nivelul de licență; DAP (disciplină de aprofundare)/ DSI (disciplină de sinteză)/ DCA (disciplină de cunoaștere avansată) - pentru nivelul de masterat;*
- 4) Regimul disciplinei (obligativitate) - *se alege una din variantele: DI (disciplină obligatorie)/ DO (disciplină opțională)/ DFac (disciplină facultativă);*
- 5) Un credit este echivalent cu 25 – 30 de ore de studiu (activități didactice și studiu individual).