

FI A DISCIPLINEI

Programarea Obiect-Orientat

Anul universitar 2019-2020

1. Date despre program

1.1	Institu ia de învățământ superior	Universitatea din Pitești
1.2	Facultatea	Electronica, Comunicatii si Calculatoare
1.3	Departamentul	Electronica, Calculatoare si Inginerie Electrica
1.4	Domeniul de studii	INGINERIE ELECTRONICĂ, TELECOMUNICAȚII ȘI TEHNOLOGII INFORMATIONALE
1.5	Ciclul de studii	Licență
1.6	Programul de studii / Calificarea	Rețele și software de telecomunicații/ Inginer emisie (215301); Inginer proiectant comunicații (215310).

2. Date despre disciplină

2.1	Denumirea disciplinei					PROGRAMAREA OBIECT-ORIENTAT					
2.2	Titularul activit ilor de curs					.L. dr. ing. Cosmin TIRBU					
2.3	Titularul activit ilor de laborator					.L. dr. ing. Cosmin TIRBU					
2.4	Anul de studii	II	2.5	Semestrul	I	2.6	Tipul de evaluare	Examen	2.7	Regimul disciplinei	D/O

3. Timpul total estimat

3.1	Număr de ore pe săptămână	3	3.2	din care curs	2	3.3	laborator	1
3.4	Total ore din planul de învăț.	42	3.5	din care curs	28	3.6	laborator	14
Distribuția fondului de timp								ore
Studiul după manual, suport de curs, bibliografie și notițe								12
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren								10
Pregătirea seminariilor/laboratoare, teme, referate, portofolii, eseuri								6
Tutoriat								
Examinări								5
Alte activități								
3.7	Total ore studiu individual	33						
3.8	Total ore pe semestru	75						
3.9	Număr de credite	3						

4. Precondiții (acolo unde este cazul)

4.1	De curriculum	
4.2	De competențe	Competențe acumulate la disciplina Programarea calculatoarelor și limbaje de programare

5. Condiții (acolo unde este cazul)

5.1	De desfășurare a cursului	Sală dotată cu tablă de scris
5.2	De desfășurare a seminarului/laboratorului	Laboratorul disciplinei (sala T 203), calculatoare, internet, limbajul de programare Java

6. Competențe specifice acumulate

Competențe profesionale	C.3. Aplicarea cunoștințelor, conceptelor și metodelor de bază privitoare la arhitectura sistemelor de calcul, microprocesoare, limbaje și tehnici de programare ; - IPC C.6. Utilizarea limbajelor și instrumentelor specializate pentru inginerie software, cu orientare către sistemele de telecomunicații integrate – IPC
Competențe transversale	CT.1. Analiza metodică a problemelor întâlnite în activitate, identificând elementele pentru care există soluții consacrate, asigurând astfel îndeplinirea sarcinilor profesionale; - IPC

7. Obiectivele disciplinei

7.1	Obiectivul general al disciplinei	Realizarea de aplicații software folosind programarea orientată pe obiecte (Studiul principiilor programării orientate pe obiect, dezvoltarea de aplicații cu obiecte, studiul limbajului Java)
7.2	Obiective specifice	<i>Obiective cognitive</i> - cunoașterea conceptului de POO

	<ul style="list-style-type: none"> - cunoașterea principiilor utilizate în POO; <p><i>Obiective procedurale</i></p> <ul style="list-style-type: none"> - formarea deprinderilor și abilitatea de a proiecta o aplicație bazată pe POO; - formarea deprinderilor pentru utilizarea de instrumente software pentru punerea la punct a unui proiect software în limbajul Java <p><i>Obiective atitudinale</i></p> <ul style="list-style-type: none"> - să surprindă diferențele între diversele tipuri de limbaje de programare; - să rezolve problemele legate de programarea orientată pe obiecte;
--	--

8. Conținuturi

8.1. Curs		Metode de predare	Observații Resurse folosite
1.	Generalități. Clase elementare; Mașina virtuală Java; Portabilitate la nivel binar – 2 ore	Prelegere Dezbateri	Tablă de scris
2.	Tipuri primitive de date. Clase și obiecte. Instantierea obiectelor. – 2 ore	Prelegere Dezbateri	Tablă de scris
3.	Variabile de instanță. Variabile locale. Specificatori de acces - 2 ore	Prelegere Dezbateri	Tablă de scris
4.	Constructorii. Aplicații cu clase elementare – 2 ore	Prelegere Dezbateri	Tablă de scris
5.	Vecori intrinseci – 2 ore	Prelegere Dezbateri	Tablă de scris
6.	Clasa Vector. Matrici. – 2 ore	Prelegere Dezbateri	Tablă de scris
7.	Aplicații cu vectori intrinseci, vectori și matrici – 2 ore	Prelegere Dezbateri	Tablă de scris
8.	Șiruri de caractere – 2 ore	Prelegere Dezbateri	Tablă de scris
9.	Clasele String, StringBuffer – 2 ore	Prelegere Dezbateri	Tablă de scris
10.	Clasa StringTokenizer – 2 ore	Prelegere Dezbateri	Tablă de scris
11.	Clase derivate. – 2 ore	Prelegere Dezbateri	Tablă de scris
12.	Interfete. Clase abstracte – 2 ore	Prelegere Dezbateri	Tablă de scris
13.	Excepții – 2 ore	Prelegere Dezbateri	Tablă de scris
14.	Fisiere – 2 ore	Prelegere Dezbateri	Tablă de scris

Bibliografie

1. Țirbu, C. – POO, note de curs, suport scris și electronic
2. Lalani S., Jamsa K., - JAVA, Ed. ALL, 1997.
3. Tanasa S., Olaru C., Andrei S. – Java de la 0 la expert, Ed. Polirom, 2003
4. Ene, Al., Țirbu, C. – Programare orientată pe obiecte. Teorie și aplicații în JAVA, Ed. Univ. din Pitești, 2008

8.2. Aplicații – Laborator		Metode de predare	Observații Resurse folosite
1.	Instrucțiuni elementare : atribuire, decizie, repetare (I), (II) – 2 ore	Lucrul pe platforme de laborator	PC cu limbajul Java instalat
2.	Dezvoltarea și utilizarea de clase elementare (I), (II) – 2 ore	Lucrul pe platforme de laborator	PC cu limbajul Java instalat
3.	Vecori. Matrici (I), (II) – 2 ore	Lucrul pe platforme de laborator	PC cu limbajul Java instalat
4.	Șiruri de caractere (I), (II)- 2 ore	Lucrul pe platforme de laborator	PC cu limbajul Java instalat
5.	Moștenire. Clase derivate. Clase abstracte. Polimorfism 2 ore	Lucrul pe platforme de laborator	PC cu limbajul Java instalat
6.	Fisiere (I), (II) -2 ore	Lucrul pe platforme de laborator	PC cu limbajul Java instalat
7.	Proba practică. Verificarea deprinderilor și abilităților teoretice și practice dobândite de fiecare student – 2 ore		

Bibliografie

1. Țirbu, C. – POO, lucrări de laborator în format scris și electronic.

9. Coroborarea conținuturilor disciplinei cu activitățile reprezentative ale comunității epistemice, asociațiilor profesionale și angajatori din domeniul aferent programului

- Conținutul disciplinei a fost conceput ca urmare a studiului fișelor disciplinelor similare predate la programele de studii similare la universități din țară (Timișoara, Cluj, Iași, București)
- În vederea actualizării și îmbunătățirii conținutului disciplinei, cadrul didactic a participat cu lucrări științifice la conferințe (ECAI), a publicat la reviste lucrări din domeniu (University of Pitesti Scientific Bulletin, Series: Electronics and Computers Science).
- întâlniri de lucru cu specialiști din producție și angajatori (Visionware, Lisa Draxlmaier, Renault Technologie Roumanie);
- Competențele dobândite la disciplină permit absolvenților să lucreze ca Proiectant de software, Analist programator în domeniul software, Programator de sistem informatic,

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota final
10.4 Curs	Capacitatea de a corela cunoștințele și de a le aplica în cazuri de proiectare și de implementare. Înțelegerea și aplicarea corectă a problematicei tratate, capacitatea de analiză și sinteză.	a) Susținere orală a temei de casă dat la curs b) Test scris- verificare pe parcurs c) Examen scris (sesiune de examene) - verificare cunoștințe teoretice	10% 20% 50%
10.5 Laborator	Rezolvarea studiilor de caz folosind cunoștințele cumulate la laborator și curs.	Evaluarea rezolvării studiilor de caz.	20%
10.6 Standard minim de performanță	<p>Distribuția punctajului minim pe activități este la alegerea studentului, dar cu respectarea următoarelor cerințe: Notă minimă 5 la activitatea de laborator și prezență la laborator 100% (1 punct) și nota minimă 5 la examenul final (2.5 puncte).</p> <p>Set de cunoștințe minimale pentru promovarea verificărilor finale:</p> <ol style="list-style-type: none"> Lucrul cu clase elementare în limbajul Java (să scrie o clasă elementară în Java, să folosească corect instrucțiunile de atribuire, decizie, repetare) Specificatori de acces (să cunoască specificatorii de acces și să-i folosească: <i>private</i>, <i>static</i>, <i>public</i>) Definirea de constructori într-o clasă scrisă în limbajul Java (definire, să scrie un constructor pentru o clasă). Polimorfism parametric (definire, exemplificare). Vectori intrinseci (lucrul cu vectorii intrinseci). Matrici (lucrul cu matrici în limbajul Java). Clasa Vector. (să poată folosi câteva metode din această clasă) Șiruri de caractere: Clasa <i>StringTokenizer</i> (să poată prelucra șiruri de caractere, să folosească metodele de bază din clasa <i>StringTokenizer</i>) Clase derivate (să poată deriva o clasă) Lucrul cu fișiere în limbajul Java (să poată citi și scrie într-un fișier, folosind limbajul Java) 		

Data completării
12.09.2019

Titular de curs
.L. dr. ing. Cosmin TIRBU

Titular de laborator
.L. dr. ing. Cosmin TIRBU

Data avizării în departament
19.09.2019

Director de departament
Prof.univ.dr. ing. GH. ERBAN