

# FIȘA DISCIPLINEI<sup>1)</sup>

## 1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Petrol-Gaze din Ploiești
1.2. Facultatea	Inginerie Mecanică și Electrică
1.3. Departamentul	Automatică, Calculatoare și Electronică
1.4. Domeniul de studii universitare	Calculatoare și Tehnologia Informației
1.5. Ciclul de studii universitare	Licență
1.6. Programul de studii universitare	Calculatoare

## 2. Date despre disciplină

2.1. Denumirea disciplinei	Programare orientată pe obiecte
2.2. Titularul activităților de curs	Prof. dr. ing. Mihaela Oprea
2.3. Titularul activităților seminar/laborator	Șef lucrări dr. mat. Mădălina Cărbureanu
2.4. Titularul activității proiect	
2.5. Anul de studiu	II
2.6. Semestrul *	3
2.7. Tipul de evaluare	Examen
2.8. Categoria formativă** / regimul*** disciplinei	DD / O

\* numărul semestrului este conform planului de învățământ;

\*\* DF - Discipline fundamentale; DD - discipline de domeniu; DS - discipline de specialitate; DC - discipline complementare, DA - disciplina de aprofundare, DSI- disciplina de sinteza.

\*\*\* obligatorie = O; opțională = A; facultativă = L

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

3.1. Număr de ore pe săptămână	5	din care: 3.2. curs	2	3.3. Seminar/laborator	3	3.4. Proiect	
3.5. Total ore din planul de învățământ	70	din care: 3.6. curs	28	3.7. Seminar/laborator	42	3.8. Proiect	
3.9. Distribuția fondului de timp							ore
Studiu după manual, suport de curs, bibliografie și notițe							20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren							15
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri							20
Tutoriat							7
Examinări							8
Alte activități							10
3.10 Total ore studiu individual	80						
3.11. Total ore pe semestru	150						
3.12. Numărul de credite	6						

## 4. Precondiții (acolo unde este cazul)

4.1. de curriculum	➤ Programarea calculatoarelor și limbaje de programare
--------------------	--

<sup>1)</sup> Adaptare după Ordinul Ministrului educației, cercetării, tineretului și sportului nr. 5 703/2011 privind implementarea Codului național al calificărilor din învățământul superior, publicat în Monitorul Oficial al României, partea I, nr.880 bis / 13.XII.2011

	➤ Proiectarea algoritmilor
4.2. de competențe	<ul style="list-style-type: none"> <li>➤ Operarea cu fundamente ale informaticii specifice programării calculatoarelor într-un limbaj de programare (limbajul C);</li> <li>➤ Utilizare structuri de date și algoritmi.</li> </ul>

## 5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	➤
5.2. de desfășurare a seminarului/laboratorului	➤ Laborator dotat cu tehnică de calcul și medii de programare, C++ și Java.

## 6. Competențe specifice acumulate

<b>Competențe profesionale</b>	<ul style="list-style-type: none"> <li>➤ Cunoașterea și înțelegerea principalelor concepte și fundamente din domeniul informaticii aplicate (CpS-1) – concepte și fundamente programării orientate pe obiecte;</li> <li>➤ Operarea cu metode și concepte moderne din știința calculatoarelor (CpS-3) – operarea cu metode și concepte specifice programării orientate pe obiecte;</li> <li>➤ Cunoașterea și utilizarea limbajului de programare orientat pe obiecte C++ pentru aplicații practice (CpS-4);</li> <li>➤ Proiectarea și implementarea orientată pe obiecte a aplicațiilor, utilizând principiile de management de proiect (CpS-9) – dezvoltarea unei aplicații orientată pe obiecte în cadrul unei echipe de lucru.</li> </ul>
<b>Competențe transversale</b>	<ul style="list-style-type: none"> <li>➤ Comportarea onorabilă, responsabilă, în spiritul eticii profesionale și respectării legislației curente (inclusiv a drepturilor de proprietate intelectuală), pentru a asigura reputația statutului de student și a profesiei alese (CtS-1);</li> <li>➤ Demonstrarea spiritului de integrare, de inițiativă și de identificare a problemelor și responsabilităților din cadrul unei echipe de lucru interdisciplinare și plurispecializate (CtS-3);</li> <li>➤ Abilitatea de a conduce eficient un colectiv de lucru, de a comunica în bune condiții, de a lua decizii în timp real, de a distribui sarcini și verifica îndeplinirea acestora la toate nivelurile subordonate (CtS-7).</li> </ul>

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

7.1. Obiectivul general al disciplinei	➤ La sfârșitul cursului, studentul va fi capabil să dezvolte aplicații software, orientate pe obiecte, urmând etapele de analiză, proiectare și implementare într-un limbaj de programare orientat pe obiecte (limbajul C++/Java).
7.2. Obiectivele specifice	<p>La sfârșitul cursului, studentul va fi capabil</p> <ul style="list-style-type: none"> <li>➤ să identifice, să definească și să explice conceptele fundamentale ale programării orientate pe obiecte;</li> <li>➤ să aplice conceptele fundamentale și metodele programării orientate pe obiecte;</li> <li>➤ să analizeze, să proiecteze și să implementeze aplicații software, folosind metoda orientată pe obiecte.</li> </ul>

## 8. Conținuturi

8.1. Curs	Nr.ore	Metode de predare	Observații
Introducere în programarea orientată pe obiecte	2	interactivă și convențională, centrată pe student	Suport de curs în format tipărit și electronic
Limbajul C++. Încapsulare. Clase și metode	6	interactivă și convențională, centrată pe student	Suport de curs în format tipărit și electronic
Redefinirea operatorilor în limbajul C++	2	interactivă și convențională, centrată pe student	Suport de curs în format tipărit și electronic
Clase de bază și clase derivate. Moștenirea în limbajul C++	4	interactivă și convențională, centrată pe student	Suport de curs în format tipărit și electronic
Funcții virtuale și polimorfism în limbajul C++	2	interactivă și convențională, centrată pe student	Suport de curs în format tipărit și electronic
Stream-uri de intrare/ieșire și fișiere în limbajul C++	2	interactivă și convențională, centrată pe student	Suport de curs în format tipărit și electronic
Ingineria programării orientate pe obiecte	4	dezbatere studii de caz, centrare pe student în relație cu dezvoltarea abilităților practice	Suport de curs în format tipărit și electronic
Limbajul Java	6	interactivă și convențională, centrată pe student	Suport de curs în format electronic
<b>Bibliografie</b> [1] M. Oprea (2017), <i>Programare orientată pe obiecte – Limbajul C++</i> , Editura Universității Petrol-Gaze din Ploiești. [2] M. Oprea (2003), <i>Programare orientată pe obiecte – Exemple în limbajul C++</i> , Editura Matrix Rom, București. [3] H. M. Deitel, P.J. Deitel (2001, 2016), <i>C++ How to program</i> , Prentice Hall. [4] H. M. Deitel, P.J. Deitel (1999, 2015), <i>Java™ How to program</i> , Prentice Hall. [5] S. Reiss (1999), <i>A practical introduction to software design in C++</i> , John Wiley & Sons. [6] C. Spircu, I. Lopătan (1995), <i>Analiza, proiectarea și programarea orientate spre obiecte</i> , Editura Teora, București. [7] B. Stroustrup (1986), <i>The C++ programming language</i> , Addison Wesley.			
8.2. Seminar / laborator	Nr. ore	Metode de predare	Observații
Recapitulare - limbajul C, structuri de date și algoritmi	6	clasică, centrată pe student și pe rezultatele învățării	îndrumar de laborator
Abstractizarea datelor. Trecerea de la tipul struct la clase în limbajul C++	6	clasică, centrată pe student și pe rezultatele învățării	îndrumar de laborator
Încapsulare. Clase și metode în limbajul C++	6	clasică, centrată pe student și pe rezultatele învățării	îndrumar de laborator
Redefinirea operatorilor în limbajul C++	2	clasică, centrată pe student și pe rezultatele învățării	îndrumar de laborator
Clase de bază și clase derivate. Moștenirea în limbajul C++	4	clasică, centrată pe student și pe rezultatele învățării	îndrumar de laborator
Funcții virtuale și polimorfism în limbajul C++. Fișiere în limbajul C++	3	clasică, centrată pe student și pe rezultatele învățării	îndrumar de laborator

Analiza, proiectarea și implementarea unei aplicații orientate pe obiecte	7	dezbatere studii de caz, centrare pe student în relație cu dezvoltarea abilităților practice	îndrumar de laborator
Limbajul Java – aplicații	8	clasică, centrată pe student și pe rezultatele învățării	îndrumar de laborator
Bibliografie			
[1] M. Oprea, M. Cărbureanu (2018), <i>Programare orientată pe obiecte – Îndrumar de laborator</i> , Editura Universității Petrol-Gaze din Ploiești.			
[2] M. Oprea (2003), <i>Programare orientată pe obiecte – Exemple în limbajul C ++</i> , Editura Matrix Rom, București.			
<b>8.3. Proiect</b>	Nr. ore	Metode de predare	Observații
Bibliografie			

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

- Conținuturile disciplinei sunt specifice domeniului dezvoltării software-ului orientat pe obiecte și sunt coroborate cu așteptările comunității epistemice, a asociațiilor profesionale și angajatorilor ce activează în acest domeniu.

## 10. Evaluare

Tip activitate	10.1. Criterii de evaluare	10.2. Metode de evaluare	10.3. Pondere din nota finală
10.4. Curs	Examinare finală	Lucrare scrisă cu subiecte teoretice și aplicație	50%
	Teste la curs	Lucrare scrisă, examinare orală	10%
10.5. Seminar/laborator	Activitate laborator și verificări periodice	Examinare orală, lucrare scrisă	30%
	Temă de laborator	Lucrare scrisă descriptivă și Program	10%
10.6. Proiect			
10.7. Standard minim de performanță			
➤ Cunoașterea conceptelor fundamentale ale programării orientate pe obiecte, la nivel teoretic și practic, de programare în limbajul C++/Java.			
➤ Elemente fundamentale de analiză, proiectare și implementare a aplicațiilor orientate pe obiecte.			

Data  
completării  
20.09.2021

Semnătura titularului de curs

Semnătura titularului de  
seminar/laborator

Semnătura titularului de proiect

Data avizării în  
departament  
28.09.2021

Director de departament  
Conf. dr. ing. PRICOP Emil  
(Semnătură)

Decan  
Conf. dr. ing. DINIȚĂ Alin  
(Semnătură)