

FI A DISCIPLINEI¹⁾

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Petrol-Gaze din Ploiești
1.2. Facultatea	Litere și Științe
1.3. Departamentul	Informatic , Tehnologia Informației, Matematică și Fizică
1.4. Domeniul de studii universitare	Informatic
1.5. Cîdul de studii universitare	Licență
1.6. Programul de studii universitare	Informatic

2. Date despre disciplin

2.1. Denumirea disciplinei	Ingineria Program rii
2.2. Titularul activit ilor de curs	Conf. dr. ing. Monica Vladuiu
2.3. Titularul activit ilor aplicative	Asist. Dr. Elia Dragomir
2.4. Anul de studiu	III
2.5. Semestrul*	5
2.6. Tipul de evaluare	E
2.7. Categoria formativ ** / regimul*** disciplinei	F0/ O

*num rul semestrului este conform planului de învățământ;

** fundamental = F0; de domeniu = D1; de specialitate = S2; complementar = C3

***obligatorie = O; op țional = A; facultativ = L

3. Timpul total estimat (ore pe semestru al activit ilor didactice)

3.1. Num r de ore pe s pt mînt	4	din care: 3.2. curs	2	3.3. Seminar/laborator	2
3.4. Total ore dîn planul de învățământ	56	din care: 3.5. curs	28	3.6. Seminar/laborator	28
3.7. Distribu ia fondului de timp					ore
Studiu dup manual, suport de curs, bibliografie i noti e					30
Documentare suplimentar în bibliotec , pe platformele electronice de specialitate i pe teren					10
Preg țire seminarii/laboratoare, teme, referate, portofolii i eseuri					30
Tutoriat					9
Examini					-
Alte activit i					45
3.7. Total ore studiu individual	124				
3.8. Total ore pe semestru	180				
3.9. Num rul de credite	6				

4. Precondi ii (acolo unde este cazul)

4.1. de curriculum	Programarea Procedural , Programarea Procedural Avansat , Algoritmi i Structuri de Date, Programare Orientat pe Obiecte
4.2. de competen e	Cunoa țerea program rii procedurale (paradigm i limbaj, fundamente i elemente avansate) Cunoa țerea fundamentelor analizei, proiect rii i program rii orientate pe obiecte Cunoa țterea principalelor structuri de date si a algoritmilor de prelucrare a acestora

5. Condi ii (acolo unde este cazul)

5.1. de desf urare a cursului	sal de curs multimedia cu videoproiector i conexiune la Internet
5.2. de desf urare a seminarului/laboratorului	sal de laborator echipat cu re ea de calculatoare i software corespunz țor pentru dezvoltare de aplica ii în Java

¹⁾ Adaptare dup Ordinul Ministrului educației, cercet rii, tineretului i sportului nr. 5 703/2011 privind implementarea Codului na țional al calific rilor dîn învățământul superior, publicat în Monitorul Oficial al României, partea I, nr.880 bis / 13.XII.2011

6. Competen e specifice acumulate

Competen e profesionale	<ul style="list-style-type: none"> • C2.1 Identificarea de metodologii adecvate de dezvoltare a sistemelor software; • C2.2 Identificarea i explicarea mecanismelor adecvate de specificare a sistemelor software; • C2.3 Utilizarea metodologiilor, mecanismelor de specificare i a mediilor de dezvoltare pentru realizarea aplica iilor informatice; • C2.4 Utilizarea de criterii i metode adecvate pentru evaluarea aplica iilor informatice; • C2.5 Realizarea unor proiecte informatice dedicate; • C3.2 Identificarea si explicarea modelelor informatice de baza adecvate domeniului de aplicare; • C3.3 Utilizarea modelelor si instrumentelor informatice si matematice pentru rezolvarea problemelor specifice domeniului de aplicare; • C4.2 Interpretarea de modele matematice i informatice (formale); • C4.3 Identificarea modelelor si metodelor adecvate pentru rezolvarea unor probleme reale; • C4.5 Încorporarea de modele formale în aplica ii specifice din diverse domenii.
Competen e transversale	<ul style="list-style-type: none"> • CT1. Aplicarea regulilor de munc organizat i eficient , a unor atitudini responsabile fa de domeniul didactic, tiin ific i profesional, în vederea valorific rii creative a propriului poten ial, cu respectarea principiilor i normelor de etic profesional ; • CT2. Desf urarea eficient a activit ilor organizate în echip i dezvoltarea capacit ilor empatice i de comunicare inter-personal , de rela ionare i colaborare cu persoane i grupuri diverse implicate în dezvoltarea i utilizarea de sisteme software; • CT3. Utilizarea de metode i tehnici eficiente de înv are, informare, cercetare i dezvoltare a capacit ilor de valorificare a cuno tin elor, dar i de adaptare la cerin ele unei societ i dinamice i în continu schimbare, precum i dezvoltarea capacit ii de a comunica eficient i profesionist atât în limba român , cât i într-o limb de circula ie interna ional , prin însu irea i folosirea adecvat a limbajului de specialitate.

7. Obiectivele disciplinei (reie ind din grila competen elor specifice acumulate)

7.1. Obiectivul general al disciplinei	<p><i>Dobîndirea de c tre studen i a cuno tin elor fundamentale privind dezvoltarea de sisteme i aplica ii software complexe i aplicarea lor corespunz toare în practica dezvolt rii acestor sisteme, al turi de deprinderea de a lucra în echipe mici de programatori i de a prezenta conving tor rezultatele efortului de echip .</i></p>
7.2. Obiectivele specifice	<p><i>Dup parcurgerea disciplinei studen ii vor putea s :</i></p> <ul style="list-style-type: none"> • fac diferen a între dezvoltarea de programe de mici dimensiuni si abordarea cu metode ingineresti a dezvolt rii sistemelor i aplica iilor software complexe, de mari dimensiuni; • rezume corect principalele caracteristici ale unui pachet software „de încredere”; • s descrie i s aplice în mod adecvat principiile Codului de practic i etic profesional în Ingineria Program rii; • descrie i s analizeze comparativ principalele modele ale procesului de dezvoltare de software; • descrie principalele caracteristici, particularit i i probleme din managementul dezvolt rii de sisteme mari de programe; • explice importan a factorilor umani în ingineria program rii; • explice i s aplice adecvat principiile ingineriei sistemelor i aplica iilor software complexe, ale specific rii, proiect rii, dezvolt rii i valid rii lor, în contextul folosirii unor platforme integrate i sub inciden a unui management adecvat; • dezvolte pachete integrate de programe cu structuri de date fundamentale, folosind limbajul Java; • lucreze în cadrul unor echipe mici de programatori pentru a finaliza i prezenta corespunz tor un proiect software atât c tre management, cât i c tre client .

8. Conținuturi

8.1. Curs	Nr. ore	Metode de predare	Observații
1. Ingineria programării (IP). Software bine făcut. Diversitatea sistemelor software. Activități și principii ale IP. Încrederea în sistemele software. Etica în IP.	6	<ul style="list-style-type: none"> • prelegeri active și angajante; • supervizare și mentorat “deschise”; • învățarea prin descoperire; • învățare pe grupuri; • învățare bazată pe proiecte și pe studii de caz; • învățare bazată pe rezolvarea de probleme; • învățare centrat pe student; • learning by doing; • brainstorming; • învățare hibridă cu folosirea resurselor educaționale open; • învățare reflectiv etc. 	
2. Modele ale procesului software. Modelul cascada. Dezvoltarea incrementală și iterativă. IP orientată pe reutilizare. Prototipizarea. Modelul spirala al lui Boehm. Procesul unificat Rational. Metode și procese de tip Agile (eXtreme Programming, Scrum). Free și Open Source Software.	4		
3. Managementul proiectelor software. Generalități, particularități și obiective. Funcții și activități. Factori umani. Structuri manageriale. Productivitatea programatorului. Planificarea și realizarea orarului. Mentenanța și evoluția software-ului. Managementul configurațiilor. Documentația. Asigurarea calității.	4		
4. Proiectarea sistemelor de programe - Proiectarea programelor și a sistemelor software. Modelarea proiectării sistemelor software - proiectare orientată pe obiecte, proiectare orientată pe funcții. Proiectarea interfeței utilizator.	4		
5. Specificația sistemelor de programe - Fundamente ale specificației programelor. Specificarea cerințelor. Validare și prototipizare. Specificarea software-ului (axiomatică, algebrică, bazată pe model)	2		
6. Validarea sistemelor de programare - Verificarea și validarea programelor. Tehnici de testare. Verificare statică. Utilitare pentru testare și depanare	2		
7. Practici de programare, tehnici și medii integrate de dezvoltare.	2		
Bibliografie <ol style="list-style-type: none"> 1. Braude, E. J., <i>Software engineering: Modern approaches</i>, New Jersey, John Wiley & Sons, 2011* 2. Ghezzi, C., <i>Fundamentals of Software Engineering</i>, New Jersey, Prentice Hall, 2003* 3. Jacobson, I., <i>The essence of software engineering: Applying the SEI kernel</i>, Boston, San Francisco, Addison-Wesley, 2013* 4. Jones, C., <i>Software engineering best practices: Lessons from successful projects in top companies</i>, New-York, McGraw Hill, 2010* 5. McConell, S., <i>Code Complete: A practical handbook of software construction</i>, Microsoft Press, 2004* 6. Sommerville I., <i>Software Engineering</i>, Pearson, Boston New York, 2011* 7. Serbanati L. D., <i>Integrating tools for software development</i>, Yourdon Press, 1993 8. Vlădoiu, M., Constantinescu Z., Moise, G., <i>Ingineria Programării. Fundamente</i>, Ed. UPG Ploiești, 2015 9. Resurse educaționale disponibile la http://www.unde.ro/cursuri/IP/ și http://www.unde.ro/cursuri/OCW/ <p>* Disponibile la biblioteca departamentului ITIMF</p>			
8.2. Seminar / laborator/proiect	Nr. ore	Metode de predare	Observații
1. Programare orientată pe obiecte - noțiuni fundamentale, clase și obiecte, atribute și metode, mostenire și polimorfism. Un exemplu de complexitate medie în C++ și Java. Comparatie. Implementare folosind NetBeans. Operații de intrare-iesire.	2	<ul style="list-style-type: none"> • învățarea prin descoperire; • învățare pe grupuri; • învățare bazată pe proiecte, 	
2. Programare în Java în linie de comandă. Operații de intrare-iesire.	2		

3, 4. Tablouri in Java. Implementare atit in NetBeans, cit si in linie de comanda.	4	<ul style="list-style-type: none"> • înv țare bazată pe rezolvarea de probleme; • înv are centrat pe student; • learning by doing; • brainstorming; • înv țare hibridă; • folosirea resurselor educa ționale open; • înv țare reflectivă etc. 		
5. Sortari in java (metoda bulelor, prin insertie, prin selectie).	2			
6. Liste inlantuite in Java. Lista cu header (cu doua capete). Iteratori in Java.	2			
7. Liste inlantuite sortate in Java. Sortarea listelor prin insertie.	2			
8. Stive in Java. Implementare cu tablouri si cu liste inlantuite. Inversarea unui sir de caractere, imperecherea delimitatorilor.	2			
9. Cozi in java - implementare cu tablouri si cu liste inlantuite.	2			
10. Liste dublu inlantuite in Java. Alegere teme proiect.	2			
11. Arbori binari in Java - implementare folosind referinte. Constructie si parcurgere RSD (preordine). Calculul inalțimii. Cautarea unei chei.	2			
12. Arbori binari in Java Constructie si parcurgere SRD (inordine) si SDR (postordine). Verificarea echilibrării arborelui.	2			
13, 14. Dezvoltarea unei interfete grafice utilizator folosind elemente de design (etichete, butoane, casete de validare, liste de optiuni, zone de text, layout managers etc.). Integrarea unui set de programe pentru a crea o aplicatie Java unica.	4			
Bibliografie <ol style="list-style-type: none"> 1. Waite M., Lafore R., <i>Structuri de date i algoritmi în Java</i>, Teora, 2000 2. Lemay L., Cadenhead R., <i>Java 2 fara profesor in 21 de zile</i>, Teora, 2000 3. Chan M. C., Griffith S. W., Iasi A. F., <i>Java - 1001 secrete pentru programatori</i>, Teora, 2000 4. Sedgewick, R., <i>Introduction to programming in Java. An interdisciplinary approach</i>, Pearson Education Limited, 2014* 5. Vi doiu, M., Constantinescu Z., Moise, G., <i>Ingineria Program rii. Fundamente</i>, Ed. UPG Ploie ti, 2015 6. Resurse educationale disponibile la http://www.unde.ro/cursuri/IP/ i http://www.unde.ro/cursuri/OCW/ <p>* Disponibile la biblioteca departamentului i TIMF</p>				

9. Coroborarea con inuturilor disciplinei cu a tept rile reprezentan ilor comunit ii epistemice, asocia ilor profesionale i angajatori reprezentativi din domeniul aferent programului

- Con inuturile disciplinei corespund cu a tept rile reprezentan ilor comunit ii epistemice, asocia ilor profesionale i angajatorilor reprezentativi din domeniul aferent programului, a a dup cum rezult din prezenta fi , dar i din fi a specializ rii, acestea fiind în concordan deplin cu CNCIS i COR;
- Disciplina de fa respect recomand rile IEEE/CS i ACM legate de planul de înv mânt i de con inuturile necesare pentru specializarea Informatic / tiin a Calculatoarelor;
- Disciplina de fa exist în planurile de înv mânt al tuturor marilor universit i din România i din str in tate.

10. Evaluare

Tip activitate	10.1. Criterii de evaluare	10.2. Metode de evaluare	10.3. Pondere din nota final
10.4. Curs	Dobândirea competen elor	<i>Proiect</i> : dezvoltare sistem software in echipa mic de programatori,	Documenta ie 40% Aplica ie informatic 30%

10.5. Seminar/ laborator/proiect	profesionale și transversale specifice disciplinei	aplicație informatică integrată și documentație <i>Examinare orală</i> pe teme fundamentale din conținutul cursului	Examinare orală 20% Din oficiu 10%
10.6. Standard minim de performanță			
<ul style="list-style-type: none"> • RNCIS: Realizarea și întreținerea unor aplicații informatice pentru rezolvarea unor probleme reale de complexitate medie; realizarea componentelor informatice pentru o aplicație dedicată de complexitate medie; modelarea și rezolvarea unor probleme cu grad mediu de complexitate, folosind cunoștințe de matematică și informatică; • <i>Realizarea și prezentarea proiectului dezvoltat care să conțină minim punctele 1, 2, 3, 4 și 8-documentația utilizator din lista cu conținutul proiectului prezentat la sfârșitul capitolului 2, Managementul software</i> 			

Data completării

26.09.2016

Semnătura titularului de curs

Semnătura titularului de seminar/laborator

Data avizării în departament

29.09.2016

Semnătura directorului de departament