

FIȘA DISCIPLINEI¹⁾

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Petrol-Gaze din Ploiesti
1.2. Facultatea	Litere si Stiințe
1.3. Departamentul	Informatică, Tehnologia Informației, Matematică, Fizică
1.4. Domeniul de studii universitare	Informatică
1.5. Ciclul de studii universitare	Licență
1.6. Programul de studii universitare	Informatică

2. Date despre disciplină

2.1. Denumirea disciplinei	Tehnici avansate de programare
2.2. Titularul activităților de curs	Lector dr. Daniela Tudorică
2.3. Titularul activităților seminar/laborator	Lector dr. Daniela Tudorică
2.4. Titularul activității proiect	-
2.5. Anul de studiu	II
2.6. Semestrul*	4
2.7. Tipul de evaluare	Examen
2.8. Categoria formativă** / regimul*** disciplinei	DS/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	4	din care: 3.2. curs	2	3.3. laborator	2	3.4. Proiect	-
3.5. Total ore din planul de învățământ	56	din care: 3.6. curs	28	3.7. laborator	28	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							25
Tutoriat							5
Examinări							4
Alte activități							-
3.10 Total ore studiu individual	69						
3.11. Total ore pe semestru	125						
3.12. Numărul de credite	5						

4. Precondiții

4.1. de curriculum	Fundamentele programării, Algoritmi fundamentali, Structuri de date, Metode avansate de programare, Programare orientata pe obiecte
4.2. de competențe	Proiectarea algoritmilor, cunoștințe de programare în limbajul C/C++

5. Condiții

5.1. de desfășurare a cursului	<ul style="list-style-type: none">➤ sală de curs multimedia pentru realizarea de prelegeri, conversații, dezbateri➤ suport electronic: www.ls.upg-elearning.ro https://timf.upg-ploiesti.ro/cursuri/TAP
--------------------------------	--

¹⁾ 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

5.2. de desfășurare a seminarului/laboratorului	➤ laborator dotat cu sisteme de calcul cu mediu integrat de dezvoltare instalat (Visual Studio) pentru scrierea de programe în limbajul C#
---	--

6. Competențe specifice acumulate

Competențe profesionale	<p>C1. PROGRAMAREA ÎN LIMBAJE DE NIVEL ÎNALT</p> <ul style="list-style-type: none"> ➤ C1.2 EXPLICAREA UNOR APLICAȚII SOFTWARE EXISTENTE, PE NIVELURI DE ABSTRACȚARE (ARHITECTURĂ, PACHETE, CLASE, METODE) UTILIZAND IN MOD ADECVAT CUNOȘTIȚELE DE BAZĂ ➤ C1.3 ELABORAREA CODURILOR SURSĂ ADECVATE ȘI TESTAREA UNITARĂ A UNOR COMPONENTE INTR-UN LIMBAJ DE PROGRAMARE CUNOSCUT, PE BAZA UNOR SPECIFICAȚII ➤ C1.4 TESTAREA UNOR APLICAȚII PE BAZA UNOR PLANURI DE TEST C1.5 DEZVOLTAREA DE UNITĂȚI DE PROGRAM ȘI ELABORAREA DOCUMENTAȚIILOR AFERENTE <p>C2. DEZVOLTAREA ȘI ÎNTREȚINEREA APLICAȚIILOR INFORMATICE</p> <ul style="list-style-type: none"> ➤ C2.1. IDENTIFICAREA DE METODOLOGII ADECVATE DE DEZVOLTARE A SISTEMELOR SOFTWARE ➤ C2.2. IDENTIFICAREA ȘI EXPLICAREA MECANISMELOR ADECVATE DE SPECIFICARE A SISTEMELOR SOFTWARE ➤ C2.3 UTILIZAREA METODOLOGIILOR, MECANISMELOR DE SPECIFICARE ȘI A MEDIILOR DE DEZVOLTARE PENTRU REALIZAREA APLICAȚIILOR INFORMATICE ➤ C2.4 UTILIZAREA DE CRITERII ȘI METODE ADECVATE PENTRU EVALUAREA APLICAȚIILOR INFORMATICE ➤ C2.5 REALIZAREA UNOR PROIECTE INFORMATICE DEDICATE <p>C3. UTILIZAREA INSTRUMENTELOR INFORMATICE IN CONTEXT INTERDISCIPLINAR</p> <ul style="list-style-type: none"> ➤ C3.1 DESCRIEREA DE CONCEPTE, TEORII SI MODELE FOLOSITE IN DOMENIUL DE APLICARE. ➤ 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
Competențe transversale	<p>CT1. Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională</p> <p>CT3. Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare acunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare în limba română și într-o limbă de circulație internațională</p>

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

7.1. Obiectivul general al disciplinei	Formarea de competențe profesionale și transversale necesare obținerii calificării. Obiectivul principal al disciplinei constă în familiarizarea studenților cu fundamentele teoretice și aplicative ale tehnicilor avansate de programare a calculatoarelor.
7.2. Obiectivele specifice	<p>La finalul activităților, studentul va fi capabil să:</p> <ul style="list-style-type: none"> ➤ utilizeze un mediu vizual de programare; ➤ abordeze rezolvarea diverselor categorii de probleme prin tehnici de programare adecvate; ➤ folosească abilitatea de analiză a complexității algoritmilor; ➤ aplice reguli de muncă organizată și eficientă, să manifeste atitudini responsabile față de domeniul didactic-științific, să respecte principii și norme de etică profesională, să lucreze în echipă.

8. Conținuturi

8.1. Curs	Nr.ore	Metode de predare	Observații
1. Vedere de ansamblu asupra cursului. Recapituare analiza complexitatii algoritmilor, brute force, tehnica reducerii dimensiunii problemei, backtracking, divide et impera	4		
2. Algoritmi greedy. Analiza optimalitatii solutiei	2		

3. Programare dinamica	2	Expunere pe baza suportului de curs, studii de caz, conversație, dezbateri	
4. Metode euristice de rezolvare a problemelor. Branch and Bound	2		
5. Cautare combinatoriala. Probleme de satisfacere a restricțiilor (CSP – Constraint Satisfaction Problems). Backtracking si cautare locala	4		
6. Spatii de probleme. Probleme de cautare. Rezolvarea problemelor prin cautare: căutare pe nivel, căutare de cost uniform, căutare în adâncime (limitată), căutare în adâncime cu nivel iterativ, cautare bidirecțională. Eficienta cautarii	4		
7. Elemente avansate de programare vizuala (Microsoft Visual C#). Programare generică. Tratarea excepțiilor. Accesarea și prelucrarea datelor din baze de date. Elemente de grafică în C#	8		
8. Recapitulare. Discutarea subiectelor de examen.	2		
Bibliografie			
1. Knuth D.E., <i>The art of computer programming, Vol. I – Fundamental Algorithms</i> , ediția a treia, Addison Wesley Longman, 1997			
2. Knuth D.E., <i>The art of computer programming, Vol. III – Sorting and Searching</i> , ediția a doua, Addison Wesley Longman, 1998			
3. Livovschi L., Georgescu H., <i>Sinteza și analiza algoritmilor</i> , Universitatea din București, Fac. de Matematică, București, 1985			
4. Cormen, T.H., Leiserson, C.E., Rivest R.R., Stein C., <i>Introduction to algorithms</i> , MIT, 3rd ed, 2009			
5. Sedgewick, R., Wayne, K., <i>Algorithms</i> , 4th ed., Addison-Wesley, 2011			
6. Skiena, S., <i>The algorithm design manual</i> , 2nd ed., Springer Verlag, London, 2010			
7. Tim Roughgarden, <i>Algorithms: design and analysis, Part I and II</i> , Stanford University, Coursera 2015			
8. Russell, S., Norvig, P., <i>Artificial Intelligence: A Modern Approach</i> , 3rd ed., Pearson Education Inc., Prentice Hall, New Jersey, 2014			
8.2. Laborator	Nr. ore	Metode de predare	Observații
1. Recapitulare metode avansate de programare, analiza complexitatii algoritmilor, recursivitate, brute force, tehnica reducerii dimensiunii problemei	2	Expunere, studii de caz, conversație, dezbateri, aplicații pe calculator	
2. Recapitulare Divide et Impera, Backtracking	2		
3. Tehnica Greedy. Aplicații. Determinarea daca regula greedy aleasa conduce la o solutie optima	4		
4. Tehnica Programarii dinamice. Aplicații	2		
5. Tehnica Branch and Bound. Aplicații.	4		
6. Probleme de căutare	6		
7. Elemente avansate de programare vizuala în C#. Funcții și clase generice	2		
8. Accesarea și prelucrarea în C# a datelor din baze de date MSSQL	2		
9. Grafică în C#	2		
10. Evaluarea laborator	2		
Bibliografie			

1. Knuth D.E., *The art of computer programming, Vol. I – Fundamental Algorithms*, ediția a treia, Addison Wesley Longman, 1997
2. Knuth D.E., *The art of computer programming, Vol. III – Sorting and Searching*, ediția a doua, Addison Wesley Longman, 1998
3. Livovschi L., Georgescu H., *Sinteza și analiza algoritmilor*, Universitatea din București, Fac. de Matematică, București, 1985
4. Cormen, T.H., Leiserson, C.E., Rivest R.R., Stein C., *Introduction to algorithms*, MIT, 3rd ed, 2009
5. Sedgewick, R., Wayne, K., *Algorithms*, 4th ed., Addison-Wesley, 2011
6. Skiena, S., *The algorithm design manual*, 2nd ed., Springer Verlag, London, 2010
7. Tim Roughgarden, *Algorithms: design and analysis, Part I and II*, Stanford University, Coursera 2015
8. *C# Programming Guide*, <https://msdn.microsoft.com/en-us/library/67ef8sbd.aspx>
9. *Tutorial C#*, <http://www.tutorialspoint.com/csharp/index.htm>

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

Aprofundarea teoretică și practică a disciplinei este o condiție de bază pentru dezvoltarea și întreținerea aplicațiilor informatice destinate rezolvării problemelor din context real. Cursul și laboratorul sunt astfel concepute încât, prin competențele formate, să răspundă cerințelor pieței muncii.

10. Evaluare

Tip activitate	10.1. Criterii de evaluare	10.2. Metode de evaluare	10.3. Pondere din nota finală
10.4. Curs	Completitudinea și corectitudinea cunoștințelor, capacitatea de a opera cu cunoștințele asimilate, capacitatea de analiză, de interpretare personală, gradul de asimilare a limbajului de specialitate și capacitatea de comunicare.	Examen scris	40%
10.5. Seminar/laborator	Aplicarea practică a elementelor prezentate la orele de curs și de laborator, prin realizarea de aplicații folosind tehnici avansate de programare.	Evaluarea temei de casă	60%

10.6. Standard minim de performanță

Asimilarea limbajului de specialitate privind tehnici avansate de programare.
 Abilitatea de aplicare practică a cunoștințelor privind tehnicile avansate de programare. Realizarea și întreținerea unor aplicații informatice pentru rezolvarea unor probleme reale de complexitate medie.

Data completării	Semnătura titularului de	Semnătura titularului de laborator
22 sept. 2024	curs	Lector dr. Daniela Tudorică
	Lector dr. Daniela Tudorică	

Data avizării în
departament

24 sept. 2024

Director de departament
Lector dr. Anca Baci

Decan
Prof. univ. dr. Mihaela Suditu