

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Automatică și Calculatoare
1.3 Departamentul	Calculatoare
1.4 Domeniul de studii	Calculatoare și Tehnologia Informației
1.5 Ciclul de studii	Master
1.6 Programul de studii / Calificarea	Complemente de Știința Calculatoarelor/ Master
1.7 Forma de învățământ	IF – învățământ cu frecvență
1.8 Codul disciplinei	11.

2. Date despre disciplină

2.1 Denumirea disciplinei	Algoritmi fundamentali				
2.2 Titularii de curs	Conf.dr.Ing. Camelia Lemnaru – Camelia.Lemnaru@cs.utcluj.ro				
2.3 Titularul/Titularii activităților de seminar/laborator/proiect	Conf.dr.Ing. Camelia Lemnaru – Camelia.Lemnaru@cs.utcluj.ro				
2.4 Anul de studiu	II	2.5 Semestrul	1	2.6 Tipul de evaluare (E – examen, C – colocviu, V – verificare)	E
2.7 Regimul disciplinei	DA – de aprofundare, DS – de sinteza, DC – complementară				DS
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

3. Timpul total estimat

3.1 Număr de ore pe săptămână	5	din care:	Curs	2	Seminar	1	Laborator	2	Proiect	
3.2 Număr de ore pe semestru	70	din care:	Curs	28	Seminar	14	Laborator	28	Proiect	
3.3 Distribuția fondului de timp (ore pe semestru) pentru:										
(a) Studiul după manual, suport de curs, bibliografie și notițe										15
(b) Documentare suplimentară în bibliotecă, pe platforme electronice de specialitate și pe teren										10
(c) Pregătire seminarii / laboratoare, teme, referate, portofolii și eseuri										10
(d) Tutoriat										16
(e) Examinări										4
(f) Alte activități:										
3.4 Total ore studiu individual (suma (3.3(a))...3.3(f))										55
3.5 Total ore pe semestru (3.2+3.4)										125
3.6 Numărul de credite										5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Structuri de date, Programare
4.2 de competențe	Competențele acumulate la disciplinele de mai sus

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Tabla, proiector, calculator
5.2. de desfășurare a seminarului / laboratorului / proiectului	Calculatoare, software specific

6. Competențele specifice acumulate

6.1 Competențe profesionale	<p>C2 - Elaborarea de tehnici, tehnologii, metode și metodologii specifice sistemelor informatice</p> <ul style="list-style-type: none"> • C2.1 - Recunoașterea de tehnici, metode, metodologii și tehnologii utilizate în sistemele informatice • C2.2 - Fixarea condițiilor de utilizare pentru diferite platforme de calcul din domeniul sistemelor informatice • C2.3 - Dezvoltarea de aplicații bazate pe noi tehnici, metode și metodologii destinate sistemelor informatice
-----------------------------	---

	<ul style="list-style-type: none"> • C2.4 - Evaluarea necesarului de tehnologii, resurse, echipamente și integrarea și adaptarea acestora în sisteme informatice • C2.5 - Cercetarea și dezvoltarea de tehnici, metode și metodologii noi specifice sistemelor informatice C3 - Proiectarea inovativă a sistemelor informatice dedicate <ul style="list-style-type: none"> • C3.1 - Identificarea și descrierea tehnicilor, metodelor, metodologiilor și tehnologiilor necesare în proiectarea sistemelor informatice • C3.2 - Utilizarea de concepte, principii, tehnici, metodologii și tehnologii de proiectare a sistemelor informatice • C3.3 - Crearea și utilizarea de soluții noi adecvate, în realizarea de proiecte de sisteme informatice • C3.4 - Evaluarea efectelor alternativelor de rezolvare în creșterea performanțelor sistemelor informatice • C3.5 - Elaborarea de soluții eficiente în proiectarea sistemelor informatice prin selectarea alternativelor specifice domeniului
6.2 Competențe transversale	N/A

7. Obiectivele disciplinei

7.1 Obiectivul general al disciplinei	Asimilarea algoritmilor reprezentativi pentru structurile de date fundamentale
7.2 Obiectivele specifice	Transcrierea din pseudo cod într-un limbaj de programare, rularea, testarea, evaluarea secvenței, analiza de complexitate, comparare soluții alternative

8. Conținuturi

8.1 Curs	Nr.ore	Metode de predare	Observații
Fundamente Matematice: Notație Asimptotică, Recurente	2	Clasice + Multimedia (Teams + Moodle)	
Clase de Complexitate	2		
Sortare și Ordini Statistice	2		
Sortare și Ordini Statistice (continuare)	2		
Structuri de Date Avansate : Tabele de Dispersie, Arbori	2		
Structuri de Date Avansate: Heapuri, Mulțimi Disjuncte	2		
Tehnici Avansate de Proiectare și Analiză : Programare Dinamica	2		
Tehnici Avansate de Proiectare și Analiză: Algoritmi Greedy	2		
Tehnici Avansate de Proiectare și Analiză: Analiză Amortizată	2		
Grafuri: Căutare în Graf, Arbore de Acoperire Minim	2		
Grafuri: Drumuri Minime	2		
Grafuri: Flux Maxim,	2		
Grafuri: Grafuri Bipartite	2		
Elemente de calcul de complexitate	2		
Bibliografie (<i>bibliografia minimală a disciplinei conținând cel puțin o lucrare bibliografică de referință a disciplinei, care există la dispoziția studenților într-un număr de exemplare corespunzător</i>)			
1. T.Cormen, C.Rleiserson, R.Rivest, <i>Introducere în Algoritmi</i> , Editura Agora, Ed.1(2001) sau 2 (2004).			
8.2 Aplicații (seminar/laborator/proiect)*	Nr.ore	Metode de predare	Observații
Implementarea eficientă și compararea algoritmilor de sortare	3		
Implementarea eficientă și compararea algoritmilor de sortare utilizând heap-uri	3		
Implementarea eficientă și compararea algoritmilor pe liste	3		
Implementarea eficientă și compararea algoritmilor pentru tabele de dispersie	3		
Implementarea eficientă de algoritmi pe arbori	3		
Implementarea eficientă de algoritmi pe arbori (continuare)	3		
Implementarea structurilor de date îmbogățite	3		
Implementarea structurilor de date avansate(pe multimi disjuncte)	3		
Implementarea eficientă de algoritmi pe grafuri	3		
Implementarea eficientă de algoritmi pe grafuri (continuare)	3		

Implementarea eficientă de algoritmi pe grafuri (continuare)	3		
Implementarea eficientă de algoritmi pe grafuri (continuare)	3		
Aproximarea problemelor dificile	3		
Evaluare finală	3		
Bibliografie (<i>bibliografia minimală pentru aplicații conținând cel puțin o lucrare bibliografică de referință a disciplinei care există la dispoziția studenților într-un număr de exemplare corespunzător</i>)			
1. T.Cormen, C.Rleiserson, R.Rivest, <i>Introducere in Algoritmi</i> , Editura Agora, Ed.1(2001) sau 2 (2004).			

*Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.

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

Orice problema care trebuie rezolvată printr-o aplicație software specifică necesită implementarea unor algoritmi. Cunoașterea algoritmilor reprezentativi, abilitatea de evaluare a performanței, analiza comparativă și luarea deciziei într-un anumit context sunt esențiale.

10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	Examen Partial	Examen scris	20%
	Examen Final	Examen scris	50%
Seminar	Verificare continuă	Evaluarea temelor practice	30%
Laborator			
Proiect			
Standard minim de performanță: Nota aplicației ≥ 5 , Media ≥ 5			

Data completării:	Titulari	Titlu Prenume NUME	Semnătura
	Curs	Conf.dr.ing. Camelia Lemnaru	
	Aplicații	Conf.dr.ing. Camelia Lemnaru	

Data avizării în Consiliul Departamentului Calculatoare	Director Departament Prof.dr.ing. Rodica Potolea
Data aprobării în Consiliul Facultății de Automatică și Calculatoare	Decan Prof.dr.ing. Liviu Miclea