

FIȘA DISCIPLINEI
Anul universitar 2015-2016

Decan,
Prof. dr. ing. Corneliu Lazăr

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică „Gheorghe Asachi” din Iași
1.2 Facultatea	Automatică și Calculatoare
1.3 Departamentul	Calculatoare
1.4 Domeniul de studii	Calculatoare și tehnologia informației
1.5 Ciclu de studii ¹	Licență
1.6 Programul de studii	Tehnologia informației

2. Date despre disciplină

2.1 Denumirea disciplinei	Proiectarea algoritmilor						
2.2 Titularul activităților de curs	prof. dr. Mitică Craus						
2.3 Titularul activităților de aplicații	prof. dr. Mitică Craus, ș.l. dr. ing. Cristian Buțincu, ș.l. dr. ing. Adrian Alexandrescu						
2.4 Anul de studii ²	2	2.5 Semestrul ³	4	2.6 Tipul de evaluare ⁴	examen	2.7 Tipul disciplinei ⁵	DID

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

3.1 Număr de ore pe săptămână	5	din care 3.2 curs	3	3.3a sem.		3.3b laborator	2	3.3c proiect	
3.4 Total ore din planul de învățământ ⁶	70	din care 3.5 curs	42	3.6a sem.		3.6b laborator	28	3.6c proiect	
Distribuția fondului de timp ⁷									Nr. ore
Studiul după manual, suport de curs, bibliografie și notițe									48
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren									14
Pregătire seminarii/laboratoare/proiecte, teme, referate și portofolii									14
Tutoriat ⁸									2
Examinări ⁹									2
Alte activități:									
3.7 Total ore studiu individual ¹⁰	80								
3.8 Total ore pe semestru ¹¹	150								
3.9 Numărul de credite	6								

4. Precondiții (acolo unde este cazul)

4.1 de curriculum ¹²	• Algebra și analiza matematică, Bazele programării calculatoarelor, Matematică discretă, Structuri de date
4.2 de competențe	

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului ¹³	• Tablă, videoproiector
5.2 de desfășurare a seminarului / laboratorului / proiectului ¹⁴	• Sală de laborator dotată cu rețea de calculatoare; acces la internet; medii de programare de nivel înalt

6. Competențele specifice acumulate¹⁵

Număr de credite alocate disciplinei ¹⁶ :			6	Repartizare credite pe competențe ¹⁷
Competențe profesionale	CP1	Operarea cu fundamente științifice, ingineresti și ale informaticii		1,5
	CP2	Proiectarea componentelor hardware, software și de comunicații		1
	CP3	Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor		1
	CP4	Proiectarea și integrarea sistemelor informatice utilizând tehnologii și medii de programare		1,5
	CP5	Întreținerea și exploatarea sistemelor hardware, software și de comunicații		
	CP6	Utilizarea sistemelor inteligente		0,7
Competențe transversale	CT1	Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesiei		0,2
	CT2	Identificarea, descrierea și derularea proceselor din managementul proiectelor, cu preluarea diferitelor roluri în echipă și descrierea clară și concisă, verbal și în scris, în limba română și într-o limbă de circulație internațională, a rezultatelor din domeniul de activitate		
	CT3	Demonstrarea spiritului de inițiativă și acțiune pentru actualizarea cunoștințelor profesionale, economice și de cultură organizațională		0,1

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

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> Formarea unui stil științific de analiză și modelare a unei probleme; Construirea de algoritmi corecți și eficienți; Adaptarea algoritmilor generici la aplicații.
7.2 Obiective specifice	<ul style="list-style-type: none"> Cunoașterea metodelor de proiectare a algoritmilor: modelul algoritmic, implementarea, eficiența metodei; Cunoașterea problemelor generice care pot fi rezolvate printr-o paradigmă de proiectare specificată ; Abilitatea de a aplica algoritmul adecvat unei probleme ; Capacitatea de a proiecta, implementa și testa algoritmi; Abilități de programare.

8. Conținuturi

8.1 Curs ¹⁸	Metode de predare ¹⁹	Observații
<p>Limbaj algoritmic, algoritmi și probleme, evaluarea algoritmilor. (2 ore)</p> <p>Metode de proiectare a algoritmilor orientate pe problemă. (1 oră)</p> <p>Problema căutării: (6 ore)</p> <ul style="list-style-type: none"> Căutarea în liste liniare; Căutarea în arbori binari de căutare oarecare; Căutarea în arbori de căutare echilibrați (arbori AVL, bicolori, 2-3, B și B+); Tabel hash; Arbori digitali (tries). <p>Problema sortării: (6 ore)</p> <ul style="list-style-type: none"> Sortarea bazată pe comparații (sortarea prin interschimbare, prin inserție și prin selecție); Sortarea prin metoda numărării; Sortarea prin metoda distribuirii; Sortarea topologică. <p>Metode generale de proiectare a algoritmilor. (1oră)</p> <p>Metoda <i>divide-et-impera</i> (<i>divide-and-conquer</i>): (5ore)</p> <ul style="list-style-type: none"> Descriere; Model de algoritm <i>divide-et-impera</i>; Evaluarea timpului de execuție a unui algoritm <i>divide-et-impera</i>; Studii de caz: sortarea prin interclasare, sortarea bitonică, sortarea rapidă, transformata Fourier discretă, înmulțirea numerelor întregi mari, înmulțirea matricelor pătratice. <p>Metoda <i>greedy</i>; (6 ore)</p> <ul style="list-style-type: none"> Descriere; Modelul matematic Prototip de algoritm <i>greedy</i>; Analiza paradigmei; Studii de caz: arbori binari ponderați pe frontieră, compresiile de date prin metoda arborilor Huffman, interclasarea optimă, problema rucsacului (variante continuă), drumuri minime între un vârf al unui graf și celelalte vârfuri (algoritmul lui Dijkstra). <p>Metoda programării dinamice: (6 ore)</p> <ul style="list-style-type: none"> Descrierea paradigmei; Modelul matematic; Implementarea paradigmei; Analiza complexității; Studii de caz: drumuri minime între oricare două vârfuri ale unui graf (algoritmul Floyd-Warshall), înmulțirea optimă a unei secvențe de matrice, generarea unui arbore binar de căutare optimal, problema rucsacului (variante discretă). <p>Metodele <i>backtracking</i> și <i>branch-and-bound</i>: (3 ore))</p> <ul style="list-style-type: none"> Descriere; Organizarea spațiului soluțiilor candidat; <p>Metodele <i>backtracking</i>: (3 ore)</p> <ul style="list-style-type: none"> Modele de algoritmi <i>backtracking</i> (recursiv, nerecursiv); Studii de caz: problema celor n regine, problema submulțimilor de sumă dată. <p>Metoda <i>branch-and-bound</i>: (3 ore)</p> <ul style="list-style-type: none"> Prototip de algoritm <i>branch-and-bound</i>; Implementarea structurii de așteptare; <i>Branch-and-bound</i> pentru probleme de optim; Studii de caz: perspicu. 	<ul style="list-style-type: none"> Utilizarea video-proiectorului la prelegeri; Intrebări adresate studenților după prezentarea unor noțiuni; Intrebări adresate studenților în timpul și după prezentarea unor algoritmi; Scrierea pe tablă a unor explicații și exemple suplimentare; implicarea studenților în acest proces. 	<p>Recomandarea, pentru studiul individual, a unor capitole din bibliografia indicată, în vederea aprofundării sau extinderii cunoștințelor acumulate la curs.</p>

Bibliografie curs:		
1. D. Lucanu, M. Craus, (2008), <i>Proiectarea algoritmilor</i> , Editura Polirom		
2. T. Cormen, C. Leiserson, R. Rivest, (2000), <i>Introducere în algoritmi</i> , Computer Libris Agora, Cluj		
3. C.Croitu, (1992), <i>Tehnici de bază în optimizarea combinatorie</i> , Editura Universității “Al. I. Cuza”, Iași		
8.2a Seminar	Metode de predare ²⁰	Observații
8.2b Laborator	Metode de predare ²¹	Observații
1. Recapitularea unor noțiuni C/C++ și structuri de date. (2 ore) 2. Complexitatea algoritmilor: analiza unor algoritmi elementari (2 ore) 3. Căutare secvențială și binară, căutarea pe arbori binari de căutare: programe C/C++ (2 ore) 4. Cautarea pe arbori B: programe C/C++ de inserare și căutare a unei chei (2 ore) 5. Algoritmi de sortare: programe C/C++ pentru algoritmi Buble_Sort, Insertion_Sort și Shell_Sort (2 ore) 6. Algoritmi de sortare: programe C/C++ pentru algoritmul Radix_Sort și algoritmul de sortare topologică (2 ore) 7. Metoda <i>divide-et-impera</i> : implementarea în C/C++ algoritmilor Merge_Sort, Quick_Sort (2 ore) 8. Metoda <i>divide-et-impera</i> : implementarea în C/C++ a algoritmului de înmulțire a două numere întregi mari și a algoritmului lui Strassen (2 ore) 9. Metoda <i>greedy</i> : implementarea în C/C++ a algoritmilor de interclasare optimă și comprimare de date prin metoda arborilor Huffman; aplicații (2 ore) 10. Metoda <i>greedy</i> : implementarea în C/C++ a algoritmului pentru rezolvarea problemei rucsacului (variante continuă) și a algoritmului lui Dijkstra (2 ore) 11. Programare dinamică: implementarea în C/C++ a algoritmului pentru rezolvarea problemei înmulțirii optime a unei secvențe de matrice și a algoritmului Floyd-Warshall (2 ore) 12. Programare dinamică: implementarea în C/C++ a algoritmului de rezolvare a problemei rucsacului, varianta discretă (2 ore) 13. <i>Backtracking</i> : implementarea în C/C++ a algoritmilor pentru rezolvarea problemei celor <i>n</i> regine pe tabla de șah; aplicații (2 ore) 14. Evaluare temelor de casă (2 ore)	- Utilizarea video-proiectorului pentru prezentarea temei; - Scrierea pe tablă a unor exemple și implicarea studenților în acest proces. - Asistarea studenților la îndeplinirea sarcinilor propuse.	
8.2c Proiect	Metode de predare ²²	Observații
Bibliografie aplicații (seminar / laborator / proiect):		
1. Tutoriale C/C++		
2. D. Lucanu, M. Craus, (2008), <i>Proiectarea algoritmilor</i> , Editura Polirom		

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului²³

- Actualizarea continuă a conținuturilor și metodelor de predare în funcție de rezultatele cercetării în domeniu pe plan național și mondial, cerințele mediului academic, economic și social.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	<ul style="list-style-type: none"> • Cunoștințe teoretice însușite (cantitatea, corectitudinea, acuratețea) • Rezolvarea temelor propuse și calitatea soluțiilor 	Teste pe parcurs ²⁴ :	%
		Teme de casă: Temă de casă: studiul unei probleme pentru care se cere un algoritm, implementarea acestuia și o documentație	20%
		Evaluare finală: Proba 1: a) categoria de sarcini: test de cunoștințe; b) ponderea în nota examenului: 50% Proba 2: a) categoria de sarcini: rezolvare de probleme; b) ponderea în nota examenului: 50%	50% (minim 5)
10.5a Seminar	<ul style="list-style-type: none"> • Frecvența/relevanța intervențiilor sau răspunsurilor 	<ul style="list-style-type: none"> • Evidența intervențiilor, portofoliu de lucrări (referate, sinteze științifice) 	%
10.5b Laborator	<ul style="list-style-type: none"> • Rezolvarea temelor propuse 	<ul style="list-style-type: none"> • Evaluare continuă pe parcursul orelor 	30% (minim 5)

	<ul style="list-style-type: none"> și calitatea soluțiilor • Evaluarea și interpretarea rezultatelor • Cunoașterea instrumentelor specifice • Prezența și activitatea desfășurată 	de laborator: <ul style="list-style-type: none"> - Răspuns oral - Demonstrație practică 	
10.5c Proiect	<ul style="list-style-type: none"> • Calitatea proiectului realizat, corectitudinea documentației proiectului, justificarea soluțiilor alese 	<ul style="list-style-type: none"> • Autoevaluarea, prezentarea și/sau susținerea proiectului • Evaluarea critică a unui proiect 	% (minim 5)
10.5d Alte activități ²⁵	<ul style="list-style-type: none"> • 	<ul style="list-style-type: none"> • 	% (minim 5)
10.6 Standard minim de performanță ²⁶			
<ul style="list-style-type: none"> • 			

Data completării,

22 Sept. 2014

Titular de curs,

Prof.dr. Mitică Craus

Titulari de laborator,

Prof.dr. Mitică Craus

Ș.I. dr. ing. Cristian Buțincu

Ș.I. dr. ing. Adrian Alexandrescu

Data avizării în departament:

25 Sept. 2014

Director de departament,

Prof. Petru Cașcaval

¹ Licență / Master

² 1-4 pentru licență, 1-2 pentru master

³ 1-8 pentru licență, 1-3 pentru master

⁴ Examen, colocviu sau VP A/R – din planul de învățământ

⁵ DF - disciplină fundamentală, DID - disciplină în domeniu, DS – disciplină de specialitate sau DC - disciplină complementară - din planul de învățământ

⁶ Este egal cu 14 săptămâni x numărul de ore de la punctul 3.1 (similar pentru 3.5, 3.6abc)

⁷ Liniiile de mai jos se referă la studiul individual; totalul se completează la punctul 3.7.

⁸ Între 7 și 14 ore

⁹ Între 2 și 6 ore

¹⁰ Suma valorilor de pe liniile anterioare, care se referă la studiul individual.

¹¹ Suma dintre numărul de ore de activitate didactică directă (3.4) și numărul de ore de studiu individual (3.7); trebuie să fie egală cu numărul de credite alocate disciplinei (punctul 3.9) x 24 de ore pe credit.

¹² Se menționează disciplinele obligatoriu a fi promovate anterior sau echivalente

¹³ Tablă, vidoproiector, flipchart, materiale didactice specifice etc.

¹⁴ Tehnică de calcul, pachete software, standuri experimentale, etc.

¹⁵ Competențele din Grilele G1 și G1bis ale programului de studii, adaptate la specificul disciplinei, pentru care se repartizează credite (www.rncis.ro sau site-ul facultății)

¹⁶ Din planul de învățământ

¹⁷ Creditele alocate disciplinei se distribuie pe competențe profesionale și transversale în funcție de specificul disciplinei

¹⁸ Titluri de capitole și paragrafe

¹⁹ Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții (pentru fiecare capitol, dacă este cazul)

²⁰ Discuții, dezbateri, prezentare și/sau analiză de lucrări, rezolvare de exerciții și probleme

²¹ Demonstrație practică, exercițiu, experiment

²² Studiu de caz, demonstrație, exercițiu, analiza erorilor etc.

²³ Legătura cu alte discipline, utilitatea disciplinei pe piața muncii

²⁴ Se va preciza numărul de teste și săptămânile în care vor fi susținute.

²⁵ Cercuri științifice, concursuri profesionale etc.

²⁶ Se particularizează la specificul disciplinei standardul minim de performanță din grila de competențe a programului de studii.