FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	**UNIVERSITATEA DIN ORADEA**
1.2 Facultatea	**FACULTATEA DE INGINERIE ELECTRICĂ ŞI TEHNOLOGIA INFORMAŢIEI**
1.3 Departamentul	**CALCULATOARE ŞI TEHNOLOGIA INFORMAŢIEI**
1.4 Domeniul de studii	**CALCULATOARE ŞI TEHNOLOGIA INFORMAŢIEI**
1.5 Ciclul de studii	**LICENŢĂ**
1.6 Programul de studii/Calificarea	**TEHNOLOGIA INFORMAŢIEI/INGINER**

2. Date despre disciplină

2.1 Denumirea disciplinei			**STRUCTURI DE DATE ŞI ALGORITMI**				
2.2 Titularul activităţilor de curs			prof.univ.dr.ing. Zmaranda Doina				
2.3 Titularul activităţilor de laborator/proiect			conf.dr.ing. Coman Simina				
2.4 Anul de studiu	**II**	2.5 Semestrul	**3**	2.6 Tipul de evaluare	**Ex**	2.7 Regimul disciplinei	**(I)**

(I) Impusă; (O) Opțională; (F) Facultativă

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 laborator /proiect	**2**
3.4 Total ore din planul de învăţământ	**56**	din care: 3.5 curs	**28**	3.6 laborator /proiect	**28**
Distribuţia fondului de timp ore					ore
Studiul după manual, suport de curs, bibliografie şi notiţe					28
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren					8
Pregătire seminarii/laboratoare, teme, referate, portofolii şi eseuri					25
Tutoriat					2
Examinări					6
Alte activităţi.....					

3.7 Total ore studiu individual	**69**
3.9 Total ore pe semestru	**125**
3.10 Numărul de credite	**5**

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	
4.2 de competenţe	Competențe de programare în limbajul C/C++

5. Condiţii (acolo unde este cazul)

5.1. de desfăşurare a cursului	- cursul se desfășoară față în față (sală de curs dotată cu calculator şi videoproiector); prezentare bazată pe slide-uri
5.2. de desfăşurare a laboratorului /proiectului	- laboratorul se desfășoară față în față' - sală de laborator dotată cu calculatoare conectate în rețea și software adecvat - prezenta obligatorie la toate laboratoarele 100%; - recuperarea laboratoarelor se face conform Regulamentului privind activitatea profesională a studenților în baza Sistemului European de Credite Transferabile (ECTS); se pot recupera pe parcursul semestrului maxim 30% din totalul activităţilor

6.1. Competenţele specifice acumulate	
Competenţe profesionale	**CP1.** Operarea cu fundamente ştiinţifice, inginereşti şi ale informaticii **CP2.** Proiectarea componentelor hardware, software şi de comunicaţii **CP3.** Soluţionarea problemelor folosind instrumentele ştiinţei şi ingineriei calculatoarelor
Competenţe transversale	

6.2. Rezultatele aşteptate ale învăţării	
Cunoștințe	Studentul cunoaște și identifică concepte și metode elementare privitoare la utilizarea de algoritmi și structuri de date precum și a tehnicilor de programare aferente. Studentul identifică medii de programare (framework-uri) și cunoaște modalitatea de utilizare a acestora pentru realizarea de implementări în cadrul unor aplicații concrete
Aptitudini	Studentul alege şi utilizează concepte proprii specifice dezvoltării algoritmilor și utilizării tehnicilor de programare precum și diferitelor structuri de date specifice; specifică cerințe, analizează, elaborează, dezvoltă și testează aceste concepte şi tehnici de programare în cadrul unor aplicații dezvoltate utilizând un limbaj de programare de uz general
Responsabilitate și autonomie	Studentul arată spirit de inițiativă și acțiune pentru actualizarea cunoștințelor profesionale; studentul are o comportare onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesiei

7. Obiectivele disciplinei (reieşind din grila competenţelor specifice acumulate)

7.1 Obiectivul general al disciplinei	▪ Obiectivul cursului este acela de a familiariza studentii cu varietatea structurilor de date utilizate în programare precum şi cu aplicaţiile lor cele mai reprezentative. Astfel, prin structura cursului şi laboratorului s-a avut în vedere crearea unor deprinderi practice de programare prin abordarea unor soluţii nu numai funcţional corecte dar şi eficiente, prin cunoaşterea şi utilizarea unor structuri de date şi algoritmi specifici în rezolvarea unor aplicaţii specifice. Cursul are un caracter puternic aplicativ, conţinînd un număr mare de exemple de algoritmi în format sursă, fără a restrînge însă generalitatea conceptelor prezentate.
7.2 Obiectivele specifice	▪ Cursul îşi propune prezentarea diferitelor structuri de date împreună cu algoritmii de prelucrare aferenţi, precum şi a modalităţilor în care se pot utiliza în scopul implementării diferitelor tipuri de aplicaţii : structura de arbore (arbori generalizați, arbori binari, arbori echilibrați AVL, arbori B) şi respectiv structura de graf (grafuri neorientate/orientate, grafuri ponderate). ▪ Laboratorul, realizat utilizând limbajul C++ pe platforma .NET/Visual Studio, familiarizează studenţii cu aspecte practice privind rezolvarea diferitelor tipuri de probleme prin implementarea şi adaptarea unor algoritmi şi structuri de date specifice

8. Conţinuturi

8.1.Curs	Metode de predare	Observaţii
Introducere în structura de date arbore. Concepte. Tipuri de arbori.	Expunere liberă, cu prezentarea cursului pe slide-uri, față în față discuții libere;	2
Structura de arbore generalizat. Traversarea arborilor: preordine, inordine, postordine. Implementarea arborilor generalizaţi		2
Arbori binari. Caracteristici. Implementarea structurii de arbore binar		2
Arbori binari ordonati. Arbori binari de înălţime minimă. Implementarea arborilor binari ordonaţi		4
Arbori echilibraţi AVL. Caracteristici. Implementare		2
Arbori-B. Caracteristici. Implementare		2
Structura de date graf - introducere. Concepte. Tipuri de grafuri: grafuri neorientate, grafuri orientate, grafuri ponderate		2
Implementarea structurii de graf utilizând matrici de adiacenţă		2
Implementarea structurii de graf utilizând structuri (liste) de adiacenţă		2
Traversarea grafurilor. Traversarea grafurilor prin căutare în adîncime. Traversarea grafurilor prin căutare prin cuprindere		2
Grafuri ponderate. Algoritmi pentru determinarea arborelui de acoperire minim în grafuri. Algoritmul lui Prim. Algoritmul căutării bazate pe prioritate (Kruskal).		2
Algoritmi pentru determinarea drumurilor minime în grafuri. Algoritmul lui Dijkstra. Algoritmul lui Floyd		2
Inchiderea tranzitivă şi algoritmul lui Warshall		2

Bibliografie

1. http://www.algolist.net/Data_structures
2. http://oopweb.com/Algorithms/Files/Algorithms.html
3. https://www.tutorialspoint.com/data_structures_algorithms/index.htm
4. https://www.geeksforgeeks.org/top-algorithms-and-data-structures-for-competitive-programming/
5. Mark Weiss, Data Structures & Algorithm Analysis in C++, 4th Edition, Publisher : Pearson, ISBN-10 : 013284737X, ISBN-13 : 978-0132847377, 2013
6. Dietel&Dietel, C++ How to program, 8th Edition, Pearson Publisher, ISBN-13 : 978-0132662369, ISBN-1 : 9780132662369, 2011
7. **D. Zmaranda** - *Algoritmi şi tehnici de programare*, Editura Universităţii din Oradea, ISBN 973-613-062-2, 264 pg., 2001, versiune electronică actualizată 2020, https://uoradea-my.sharepoint.com/personal/rodica_zmaranda_didactic_uoradea_ro/_layouts/15/onedrive.aspx?sortField=LinkFilename&isAscending=true&id=%2Fpersonal%2Frodica%5Fzmaranda%5Fdidactic%5Fuoradea%5Fro%2FDocuments%2FSDD%2FCurs%5FStructuri%5Fde%5Fdate%2Epdf&parent=%2Fpersonal%2Frodica%5Fzmaranda%5Fdidactic%5Fuoradea%5Fro%2FDocuments%2FSDD
8. V. Creţu, *Structuri de date şi algoritmi – vol. 1: Structuri de date fundamentale*, Editura Orizonturi Universitare Timisoara, ISBN 973-9400-74-4, 2000

8.2. Laborator	Metode de predare	Observaţii
Structura de arbore. Arbori generalizaţi	Studenţii primesc temele pentru laborator cu cel puţin o săptămână înainte, şi le studiază. La începutul laboratorului se discută modalităţile de rezolvare a aplicaţiilor propuse folosind videoproiectorul. Pe urmă, studenţii realizează partea practică a lucrării (problemele propuse) sub îndrumarea cadrului didactic.	2
Structura de arbore. Arbori binari		2
Arbori binari ordonaţi ABO. Tehnici de căutare a unui nod, traversare şi creare a ABO.		4
Arbori binari ordonaţi ABO. Tehnici de suprimare a nodurilor din ABO.		2
Arbori echilibraţi AVL. Tehnici de inserţie şi suprimare a nodurilor în arbori echilibraţi AVL.		2
Arbori-B. Tehnici de inserţie şi suprimare a nodurilor în arbori-B.		2
Structura de date graf. Implementarea grafurilor prin matrici de adiacenţă.Traversarea grafurilor.		2
Structura de date graf. Implementarea grafurilor prin structuri de adiacenţă. Traversarea grafurilor.		2
Determinarea arborelui de acoperire minim al unui graf ponderat. Algoritmul lui Prim.		2
Determinarea arborelui de acoperire minim al unui graf ponderat. Algoritmul de căutare cu prioritate (Kruskal).		2
Tehnici de determinare a drumurilor minime în grafuri. Algoritmul lui Dijkstra şi algoritmul lui Floyd.		2
Incheierea situației la laborator, verificări şi recuperări		4

Bibliografie

1. **D. Zmaranda**, Rusu Claudia - Algoritmi şi tehnici de programare – îndrumător de laborator, volumul II, Editura Universităţii din Oradea, ISBN 973-613-302-8, 130 pg., 2003, versiune electronică actualizată 2020, https://uoradea-my.sharepoint.com/personal/rodica_zmaranda_didactic_uoradea_ro/_layouts/15/onedrive.aspx?isAscending=true&id=%2Fpersonal%2Frodica%5Fzmaranda%5Fdidactic%5Fuoradea%5Fro%2FDocuments%2FSDD%2FLAB%5FStructuri%5Fde%5Fdate&sortField=LinkFilename&view=0

2. **Zmaranda Doina**, Bonaciu Marius, Coman Simina - Algoritmi si tehnici de programare, Lucrari practice de laborator, Editie revizuita, Editura Universitatii din Oradea, ISBN 978-606-10-1895-6 , 2017

9. Coroborarea conţinuturilor disciplinei cu aşteptările reprezentanţilor comunităţii epistemice, asociaţiilor profesionale şi angajatori reprezentativi din domeniul aferent programului

- Conţinutul disciplinei se regăseşte în curricula specializării de Calculatoare şi din alte centre universitare care au acreditate aceste specializări (Universitatea „Politehnica”din Timişoara, Politehnica Bucureşti) iar cunoaşterea structurilor de date şi a algoritmilor prezentaţi în cadrul acestei discipline reprezintă o cerinţă fundamentală în scopul aprofundării deprinderilor şi abilităţilor de programare necesare

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	- pentru nota 5 sunt necesare răspunsuri corecte care să însumeze în total 5 de puncte (4 puncte + 1 punct din oficiu) - pentru nota 10, sunt necesare răspunsuri corecte la toate întrebările din cadrul subiectului propus (10 puncte)	**Examen scris -** evaluarea se face față în față Studenţii primesc spre rezolvare câte un subiect cu întrebări care testează însuşirea conceptelor teoretice prezentate în cadrul cursului. Fiecare întrebare are un punctaj; în total întrebările însumează un punctaj de 9 de puncte; se acordă 1 punct din oficiu	40 %
10.5 Laborator	- pentru nota 5, realizarea unei implementări funcţionale în proporţie de 50% a problemelor propuse la laborator - pentru nota 10, cunoaşterea amănunţită a modalităţii de implementare practică a tuturor lucrărilor de laborator	**Aplicaţie practică -** evaluarea se face față în față La fiecare laborator studenţii sunt evaluaţi pe baza activității (răspunsuri la întrebări, propuneri de implementare, etc.), evaluări care se concretizează la finalul laboratorului într-o notă pe activitatea de laborator din cursul semestrului. De asemenea, în ultima oră de laborator studenţii finalizează şi predau toate aplicațiile practice propuse în cadrul laboratorului. Media dintre nota primita pentru aplicaţiile practice şi nota de la activitatea de la laborator va reprezenta nota finală la laborator	60%
Calcul **Nota disciplină**: 0.60 Nota finală laborator + 0.40 Nota examen scris Conditii de participare la examenul final: Nota finală laborator ≥ 5 Conditii de promovare: **Nota disciplină** ≥ 5			
10.7Standard minim de performanţă			
• cunoaşterea structurilor de date utilizate în programare şi familiarizarea cu aplicaţiile cele mai reprezentative ale acestora • cunoaşterea în detaliu a modalităţii de implementare a structurii de date arbore în toate variantele sale: arbori generalizaţi, arbori binari, arbori AVL, arbori-B şi utilizarea acestora împreună cu algoritmii specifici de prelucrare în cadrul implementării unor programme • cunoaşterea tehnicilor fundamentale de implementare a structurii de graf: implementarea utilizând matrici de adiacenţă şi implementarea utilizând structuri de adiacenţă • familiarizarea cu principalele tipuri de algoritmi de prelucrare a grafurilor şi a modalităţior de utilizare a structurii de date graf precum şi algoritmilor specifici de prelucrare în rezolvarea unor probleme date			

Data completării
02.09.2026

Semnătura titularului de curs,
Prof. dr. ing. Zmaranda Doina
dzmaranda@uoradea.ro

Semnătura titularului de laborator
conf.dr.ing. Coman Simina Maria
scoman@uoradea.ro

Data avizării în departament
17.09.2026

Semnătura directorului de departament,
Conf.univ. dr. Elisa Valentina Moisi
emoisi@uoradea.ro

Data avizării în consiliul facultății
18.09.2026

Semnătură Decan,
Conf. univ. dr. ing. Eugen Gergely
egergely@uoradea.ro