Anexa 6

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	**Programarea calculatoarelor şi limbaje de programare II**						
2.2 Titularul activităţilor de curs	s.l.dr.ing. Simina COMAN						
2.3 Titularul activităţilor de laborator/proiect	s.l.dr.ing. Simina COMAN						
2.4 Anul de studiu	**I**	2.5 Semestrul	**2**	2.6 Tipul de evaluare	**Ex**	2.7 Regimul disciplinei	**DF**

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					23
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren					15
Pregătire seminarii/laboratoare, teme, referate, portofolii şi eseuri					23
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	Competenţe de programare în limbajul C

5. Condiţii (acolo unde este cazul)

5.1. de desfăşurare a cursului	- sală de curs dotată cu calculator şi videoproiector – prezentare bazată pe slide-uri - prezenţă minim 50% din cursuri
5.2. de desfăşurare a laboratorului /proiectului	- sală de laborator dotată cu calculatoare, C/C++ (Visual Studio/DevC++/MinGW) - prezenta obligatorie la toate laboratoarele; - se pot recupera pe parcursul semestrului maxim 4 lucrări (30 %); - frecvenţa la orele de laborator sub 70% conduce la refacerea disciplinei

6. Competenţele specifice acumulate	
Competenţe profesionale	**C1.** Defineste arhitectura software **C3.** Defineste cerinte tehnice
Competenţe transversale	

6.b. Rezultatele aşteptate ale învăţării

Cunoştiinţe	Studentul/absolventul descrie, identifică, sumarizează concepte și metode elementare privitoare la limbaje de programare, medii de programare.
Aptitudini	Studentul/absolventul specifică cerințe, analizează, elaborează, dezvoltă și testează programe în limbaje de programare C++.
Responsabilitate și autonomie	Studentul/absolventul 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	▪ Continuând elementele de programare începute în semestrul anterior, cursul îşi propune familiarizarea studenţilor cu o serie de tehnici şi concepte de programare avansate care permit proiectarea şi dezvoltarea unor programe cu un grad ridicat de complexitate ▪ Cursul are un caracter puternic aplicativ, în scopul aprofundării deprinderilor practice de programare ale studenţilor, conţinând 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 unor tehnici şi concepte de programare avansate împreună cu metode şi algoritmi specifici, care permit proiectarea şi implementarea unor programe complexe, în scopul rezolvării diferitelor tipuri de aplicaţii: manipulare avansată a tablourilor, fişierelor, şirurilor de caractere, împreună cu o serie de algoritmi cunoscuți în domeniu. Un capitol separat abordează, prin exemple, problema

	recursivității. Sunt prezentate, de asemenea, noţiuni fundamentale de evaluare a performanţelor algoritmilor, exemplificate prin evaluări comparative precum şi de proiectare şi implementare a programelor complexe ▪ Laboratorul, realizat utilizând limbajul C++, familiarizează studenţii cu aspecte practice privind rezolvarea diferitelor tipuri de probleme prin implementarea şi adaptarea unor algoritmi şi tipuri de date specifice

8. Conţinuturi

8.1.Curs	Metode de predare	Observaţii
CAP. 1. INTRODUCERE 1.1. Tipuri şi structuri de date 1.1.1. Tipuri de date statice 1.1.2. Tipuri de date dinamice 1.2. Evaluarea performanţelor algoritmilor – concepte utilizate, notația O(n)	Expunere liberă, cu prezentarea cursului pe videoproiector şi pe tablă	2 h
CAP. 2.Șiruri de caractere 2.1. Generalităţi. 2.2. Funcții 2.3 Tehnici de căutare în șiruri de caractere 2.3.1Căutarea Directă 2.3.2 Căutarea Boyer – Moore	Expunere liberă, cu prezentarea cursului pe videoproiector şi pe tablă	2h
CAP. 3 ALGORITMI DE CĂUTARE ŞI SORTARE PENTRU STRUCTURI DE DATE STATICE 3.1.Algoritmi de căutare în tablouri 3.1.1. Tehnica fanionului 3.1.2. Căutarea binară 3.1.3 Căutarea binară performantă 3.1.4 Căutarea prin interpolare 3.2. Algoritmi pentru sortarea tablourilor 3.2.1. Metode directe de sortare 3.2.1.1. Tehnica sortării prin inserţie 3.2.1.2. Tehnica sortării prin selecţie 3.2.1.3. Tehnica sortării prin interschimbare 3.2.2. Metode avansate de sortare 3.2.2.1. Tehnica sortării prin inserţie cu diminuarea incrementului (shellsort) 3.2.2.2. Tehnica sortării arborilor prin metodaansamblelor (heapsort) 3.2.2.3. Tehnica sortării prin partiţionare (quicksort) 3.2.3. Sortarea fișierelor secvențiale (sortarea externă) 3.2.3.1. Tehnica sortării prin interclasare 3.2.3.2.Tehnica sortării prin interclasare naturală	Expunere liberă, cu prezentarea cursului pe videoproiector şi pe tablă	2h 2h 2h 2h
CAP. 4. ALGORITMI RECURSIVI 4.1. Generalităţi. 4.2. Algoritmi recursivi. Exemple 4.2.1. Algoritmi de divizare 4.2.2. Algoritmi recursivi pentru determinarea tuturor soluţiilor unor probleme 4.2.3. Algoritmi cu revenire (backtracking) 4.2.4. Algoritmi pentru determinarea optimului (knapsack)	Expunere liberă, cu prezentarea cursului pe videoproiector şi pe tablă	2h 2h 2h
CAP.5. STRUCTURA DE DATE LISTĂ 5.1 Tehnici de implementare a listelor 5.1.1. Implementarea listelor cu ajutorul tipului tablou 5.1.2. Implementarea listelor cu ajutorul tipului pointer 5.1.2.1. Crearea listelor înlănţuite. Inserţia unui nod într-o listă înlănţuită 5.1.2.2. Suprimarea nodurilor dintr-o listă înlănţuită 5.1.2.3. Traversarea unei liste înlănţuite 5.1.3. Comparaţie între metodele de implementare a listelor	Expunere liberă, cu prezentarea cursului pe videoproiector şi pe tablă	2h 2h 2h

bazate pe tipul tablou respectiv pe tipul pointer 5.2. Variante ale structurii de listă 5.2.1. Liste ordonate. Utilizarea tehnicii fanionului în structura de listă. Cautarea în listă cu reordonare 5.2.2. Liste dublu înlănţuite 5.2.3. Stive 5.2.4. Cozi		2h
CAP. 6. TEHNICA DISPERSIEI 6.1. Principiul tehnicii dispersiei 6.2. Determinarea funcţiei de dispersie. Tratarea situaţiei de coliziune	Expunere liberă, cu prezentarea cursului pe videoproiector şi pe tablă	2h

Bibliografie
1. I. Horton, P.V. Weert: *Beginning C++*, Editura Springer, 2023;
2. M.McGrath, C++ Programming, Editura In Easy Steps, 2022;
3. P.J.Deitel, H.M. Deitel, *C: How to program*, Pearson Education International, ISBN 0-13-239300-X, Fifth Edition, 2007
4. D. Knuth, *Arta programarii calculatoarelor*, volumul 3 - Sortare si cautare, Editura Teora, 2004
5. D. Zmaranda - *Algoritmi şi tehnici de programare*, Editura Universităţii din Oradea, ISBN 973-613-062-2, 264 pg., 2001, versiune electronică actualizată 2014, https://uoradea-my.sharepoint.com/personal/rodica_zmaranda_didactic_uoradea_ro/Documents/PCLPIII/PCLP_III.pdf https://uoradea-my.sharepoint.com/personal/rodica_zmaranda_didactic_uoradea_ro/Documents/SDD/Structuri_de_date.pdf
6.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
1. Determinarea timpului de execuţie a unui program 2. Tehnici de căutare în tablouri 3. Tipul de date sir.Functii. Tehnici de cautare in siruri de caractere 4. Tehnici de sortare directe ale tablourilor 5. Tehnici de sortare avansate ale tablourilor 6. Sortarea fisierelor secventiale 7. Recursivitate – algoritmi recursivi 8. Recursivitate - backtracking 9. Structura de date listă 10. Liste ordonate. Utilizarea tehnicii fanionului în structura de listă. Liste dublu înlănțuite 11. Stive și cozi 12. Tehnica dispersiei 13. Predarea lucrărilor, încheierea situației la laborator 14. Recuperări	Studenţii primesc temele pentru laborator cu cel puţin o săptămână înainte, şi le studiază (problemele de la sfârşitul laboratorului). La începutul laboratorului se discută modalităţile de rezolvare a aplicaţiilor propuse. Pe urmă, studenţii realizează partea practică a lucrării (problemele propuse) sub îndrumarea cadrului didactic.	2 h 2 h 2 h 2 h 2 h 2 h 2 h 2 h 2 h 2 h 2 h 2 h 2 h 2 h

Bibliografie
1.B. StrouStrup, *Programming Principles and Practices using C++*, Editura Pearson Education, 2023
2. Doina Zmaranda, Marius Bonaciu, Coman Simina - - *Algoritmi şi tehnici de programare* – îndrumător de laborator, volumul I , Editura Universităţii din Oradea, Editie revizuita, ISBN: 978-606-10-1895-6, 90 pg., versiune electronica, 2017
3. D. Zmaranda, Bonaciu Marius - *Algoritmi şi tehnici de programare* – îndrumător de laborator, volumul I ,

Editura Universităţii din Oradea, ISBN 973-613-302-8, 100 pg., 2003, versiune electronică actualizată 2014, https://uoradea-my.sharepoint.com/personal/rodica_zmaranda_didactic_uoradea_ro/Documents/PCLPIII/Laborator_PCLPIII.pdf

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 tipurilor de date şi a algoritmilor prezentaţi în cadrul acestei discipline reprezintă o cerinţă fundamentală în scopul formării deprinderilor şi abilităţilor de bază 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 la minim 6 întrebări - pentru nota 10, este necesar răspunsul corect la toate subiectele	**Examen scris** Studenţii primesc spre rezolvare fiecare câte un formular cu 18 întrebări. Întrebările din cadrul formularului de examen sunt sub formă de grilă sau întrebare - răspuns	50 %
10.5 Laborator	- Pe parcursul semestrului se dau teste pentru a verifica pregătirea studenților. - Implementarea problemelor propuse	**Aplicaţie practică** 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, pe parcursul semestrului se dau teste de laborator pentru a verifica cunoștințele. Aceste teste reprezintă 50% din nota finală de la laborator. Nota finală în cadrul laboratorului se obține făcând media între nota de la activitate laborator și nota teste.	50%

10.7 Standard minim de performanţă

- cunoaşterea modalităţii de evaluare analitică a performanţelor unui algoritm, evaluarea comparativă a performanţelor unor algoritmi simpli
- înţelegerea tehnicilor de programare utilizate în cadrul metodelor de căutare în tablouri precum şi a metodelor sortare directă şi avansată a tablourilor şi fişierelor şi aplicarea metodelor de căutare şi sortare în diverse categorii de programe
- înţelegerea mecanismului recursivităţii, familiarizarea cu principalele tipuri de algoritmi recursivi şi aplicarea diverselor tipuri de algoritmi recursivi în cadrul unor aplicaţii specifice; manipularea structurilor cu autoreferire (liste)
- cunoaşterea avansată a modalităţii de manipulare a şirurilor şi a algoritmilor specifici de căutare în şiruri

Data completării
1.09.2026

Semnătura titularului de curs,

s.l. dr. ing. Coman Simina
scoman@uoradea.ro

Semnătura titularului de seminar/laborator/proiect,
s.l.dr.ing Coman Simina
scoman@uoradea.ro

Data avizării în departament
17.09.2026

Semnătura directorului de departament,
Conf. dr. ing. Elisa MOISI
emoisi@uoradea.ro

Data avizării în Consiliul Facultății IETI
18.09.2026

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

** - Se va specifica : Nume, Prenume, Grad didactic şi date de contact (e-mail, pagina web, etc).
.....