

STRUCTURI DE DATE ȘI ALGORITMI

1. Date despre disciplină/modul

Facultatea	Calculatoare, Informatică și Microelectronică				
Departamentul	Informatică și Ingineria Sistemelor				
Ciclul de studii	Studii superioare de licență, ciclul I				
Programul de studii	0613.5 Informatica Aplicată				
Anul de studii	Semestrul	Tip de evaluare	Categoria formativă	Categoria de opționalitate	Credite ECTS
I (învățământ cu frecvență);	2	E,PA	F – unitate de curs de fundamentală	O - unitate de curs obligatorie	5

2. Timpul total estimat

Total ore în planul de învățământ	Din care				
	Ore auditoriale		Lucrul individual		
	Curs	Laborator/seminar	Proiect de an	Studiul materialului teoretic	Pregătire aplicații
150	30	30/15	30	15	30

3. Precondiții de acces la disciplină/modul

Conform planului de învățământ	Programarea calculatoarelor, Matematici speciale
Conform competențelor	Cunoștințe și abilități de concepere și elaborare a algoritmilor și programelor în C pentru rezolvarea problemelor la calculator

4. Condiții de desfășurare a procesului educațional pentru

Curs	Pentru prezentarea materialului teoretic în sala de curs este nevoie de tablă, proiector și calculator.
Laborator/seminar	Studentii vor perfecta rapoarte conform condițiilor formulate în indicațiile metodice. Termenul de susținere a lucrării de laborator – o săptămână după finalizarea acesteia. Pentru prezentarea cu întârziere a lucrării, aceasta se depunțează cu 1pct./săptămână de întârziere.

5. Competențe specifice acumulate

Competențe profesionale	<p>CPL 2. Proiectarea și dezvoltarea aplicațiilor (B.1)</p> <p>K1 Programe/module software adecvate. K2 Componente hardware, instrumente și arhitecturi hardware. K3 Proiectarea funcțională și tehnică. K4 Tehnologiile de ultimă oră. K5 Limbaje de programare. K6 Baze de date (DBMS). K7 Sisteme de operare și platforme software. K8 Mediul de dezvoltare integrat (IDE - integrated development environment). K9 Dezvoltarea rapidă a aplicațiilor. K10 Problemele legate de drepturile de proprietate intelectuală (IPR). K11 Tehnologia de modelare tehnică și limbaje. K12 Limbajele de definire a interfeței (IDL). K13 Probleme de securitate.</p> <p>CPL 3. Integrarea componentelor (B.2)</p> <p>K1 Componente software.</p>
-------------------------	---

	<p>K2 Impactul integrării unui sistem asupra organizației sau a sistemului existent.</p> <p>K3 Tehnici de interfațare între module, sisteme și componente.</p> <p>K4 Tehnici de testare a integrării.</p> <p>K5 Instrumentele de dezvoltare (ex. mediul de dezvoltare, gestionare, control al modificărilor și accesul la codul sursă).</p> <p>K6 Bune practici de design.</p> <p>CPL 4. Testarea aplicațiilor (B.3)</p> <p>K1 Tehnicile, infrastructura și instrumentele necesare utilizate în procesul de testare.</p> <p>K2 Ciclul de viață al unui proces de testare.</p> <p>K3 tipurile de teste (funcțional, de integrare, performanță, utilizabilitate, sarcină etc.).</p> <p>K4 Standardele naționale și internaționale care definesc criteriile de calitate pentru testare.</p> <p>K5 specificul tehnologiilor legate de web, cloud, instrumente mobile și de probleme de mediu.</p> <p>CPL 5. Implementarea soluțiilor (B.4)</p> <p>K1 Tehnici de analiză a performanței.</p> <p>K2 Tehnicile legate de gestionarea problemelor (funcționare, performanță, compatibilitate).</p> <p>K3 Software-ul de ambalare/packaging și metode și tehnici de distribuție/desfășurare.</p> <p>K4 Impactul implementării/ desfășurării asupra arhitecturii existente.</p> <p>K5 Tehnologiile și standardele care se utilizează în timpul implementării/ /desfășurării.</p>
Competențe transversale	CT1. Aplicarea, în contextul respectării legislației, a drepturilor de proprietate intelectuală (inclusiv transfer tehnologic), a metodologiei de certificare a produselor, a principiilor, normelor și valorilor codului de etică profesională în cadrul propriei strategii de muncă riguroasă, eficientă și responsabilă.
	CT2. Identificarea rolurilor și responsabilităților într-o echipă și aplicarea de tehnici de relaționare și muncă eficientă în cadrul echipei.
	CT3. Identificarea oportunităților de formare continuă și valorificarea eficientă a resurselor și tehnicilor de învățare pentru propria dezvoltare.

6. Obiectivele disciplinei/modulului

Obiectivul general	Înșușirea profundă a structurilor de date și fișiere, metodelor de sortare a datelor, aplicațiilor dinamice și a algoritmilor de căutare a soluțiilor optime.
Obiectivele specifice	Să înțeleagă și să descrie algoritmi de prelucrare a structurilor elaborate. Să elaboreze funcțiile necesare pentru prelucrarea fișierelor. Să aplice corect algoritmi și procedeele de operare în rezolvarea problemelor cu aplicații dinamice.

7. Conținutul disciplinei/modulului

Tematica activităților didactice	Numărul de ore	
	învățământ cu frecvență	învățământ cu frecvență redusă
Tematica cursurilor		
T1 Sortarea datelor. Algoritmi de sortare și analiza performanțelor.	4	
T2 Structuri de date <i>struct</i> , <i>union</i> , cu câmpuri de <i>biți</i> . Declararea, inițializarea, citirea și afișarea structurilor.	4	
T3 Tipul de date FILE. Fișiere de tip logic și fizic. Funcții predefinite pentru date de tip FILE: <i>fopen()</i> , <i>fread()</i> , <i>fwrite()</i> , <i>fprintf()</i> , <i>fscanf()</i> , <i>fseek()</i> , <i>feof()</i> . Operații cu fișiere: creare, afișare, adăugare, corectare, sortare. Acces secvențial și direct la fișiere.	6	

T4 Structuri dinamice de date. Alocare de memorie. Funcții predefinite. Aplicații dinamice: stiva, șir de așteptare, listă liniară, lista înlănțuită, listă bidirecțională. Arbori. Arbori binari de căutare. Proceduri de operare cu aplicații dinamice: creare, parcurgere, afișare, modificare.	6	
T5 Algoritmi și metode de căutare optimală a soluțiilor. Clasificarea algoritmilor. Avantaje și dezavantaje ale algoritmilor din perspectiva: universalității, capacității de memorie utilizate și a vitezei de execuție.	2	
T6 Algoritmii care garantează soluția optimală: <i>Backtracking</i> , <i>Programarea dinamică</i> , <i>Divide et Impera</i> , <i>Branch and Bound</i> .	4	
T7 Algoritmii <i>Greedy</i> , <i>euristici</i> , <i>probabilistici</i> , <i>genetici</i> . Analiza algoritmilor.	4	
Total curs:	30	
Tematica lucrărilor practice/seminarelor		
LP1 Sortarea datelor. Algoritmi de sortare și analiza performanțelor	2	
LP2 Structuri de date <i>struct</i> , <i>union</i> , cu câmpuri de <i>biți</i> . Declararea, inițializarea, citirea și afișarea structurilor.	2	
LP3 Tipul de date FILE. Fișiere de tip logic și fizic. Funcții predefinite pentru date de tip FILE. Operații cu fișiere: creare, afișare, adăugare, corectare, sortare. Acces secvențial și direct la fișiere.	3	
LP4, LP5 Structuri dinamice de date. Alocare de memorie. Funcții predefinite. Aplicații dinamice: stiva, șir de așteptare, listă liniară, lista înlănțuită, listă bidirecțională. Arbori. Arbori binari de căutare.	3	
LP6 Algoritmi și metode de căutare optimală a soluțiilor. Algoritmii care garantează soluția optimală: <i>Backtracking</i> , <i>Programarea dinamică</i> , <i>Divide et Impera</i> , <i>Branch and Bound</i> .	3	
LP7 Algoritmii <i>Greedy</i> , <i>euristici</i> , <i>probabilistici</i> , <i>genetici</i> . Analiza algoritmilor.	2	
Total lucrări practice/seminare:	15	
Tematica lucrărilor de laborator		
LL1 Sortarea datelor. Algoritmi de sortare și analiza performanțelor.	4	
LL2 Structuri de date <i>struct</i> , <i>union</i> , cu câmpuri de <i>biți</i> . Declararea, inițializarea, citirea și afișarea structurilor.	4	
LL3 Tipul de date FILE. Fișiere de tip logic și fizic. Funcții predefinite pentru date de tip FILE. Operații cu fișiere: creare, afișare, adăugare, corectare, sortare. Acces secvențial și direct la fișiere.	6	
LL4, LL5 Structuri dinamice de date. Alocare de memorie. Funcții predefinite. Aplicații dinamice: stiva, șir de așteptare, listă liniară, lista înlănțuită, listă bidirecțională. Arbori. Arbori binari de căutare.	6	
LL6 Algoritmi și metode de căutare optimală a soluțiilor. Algoritmii care garantează soluția optimală: <i>Backtracking</i> , <i>Programarea dinamică</i> , <i>Divide et Impera</i> , <i>Branch and Bound</i> .	6	
LL7 Algoritmii <i>Greedy</i> , <i>euristici</i> , <i>probabilistici</i> , <i>genetici</i> . Analiza algoritmilor.	4	
Total lucrări de laborator:	30	

8. Referințe bibliografice

Principale	<ol style="list-style-type: none"> 1. Florian Moraru. Structuri de Date. Disponibil on-line: http://andrei.clubicisco.ro/cursuri/f/f-sym/1sd/sd2010A4.pdf 2. Jack Straub. C Programming: Data Structures and Algorithms. 167 p. Disponibil on-line: https://faculty.washington.edu/jstraub/dsa/Master_2_7a.pdf 3. Logofătu Doina. Bazele programării în C : aplicații. Doina Logofătu. Iași : Polirom, 2006. 406 p. ISBN 973-46-0219-5. 4. Ștefănescu Diana. Programarea în limbajele C/C++. Noțiuni de bază. 400p. București: Matrix Rom, 2002. ISBN / ISSN 973-685-475-2. 5. Tudor Liviana. Bazele programării în C. 240 p. ISBN / ISSN 978-973-755-644-8. Editura: Matrixrom. București, 2010.
-------------------	--

Suplimentare	<p>6. Buzurniuc, Șt. Inițiere în limbajul C. Evrica. Chișinău, 2004.</p> <p>7. Cechez, Em., Șerban, M. Programarea în limbajul C/C++. Editura: POLIROM. București, 2005.</p> <p>8. Claude Delannoy Programmer en langage C. Ucors et exercices corriges. EYROLLES, Paris, 2002.</p> <p>9. Ritchie, D., Kernighan, B. W.. C Programming Language. Publisher: Prentice Hall; 2nd edition 1988. 217 p. ISBN / ISSN: 9780131103627. Disponibil on-line: http://www2.cs.uregina.ca/~hilder/cs833/Other%20Reference%20Materials/The%20C%20Programming%20Language.pdf</p> <p>10. Negrescu Liviu. Limbajul C. Vol I, Vol II, Editura albastră. Cluj-Napoca, 1999.</p> <p>11. Pătruț, B. Aplicații în C și C++. Bogdan Pătruț. București: Teora, 1998. 167 p. ISBN 973-601-760-5.</p> <p>12. Sedgewick R.. Algorithmes en language C. DUNOD, Paris, 2001.</p>
---------------------	---

9. Utilizarea IA generativă

Permisivitatea de utilizare	<p>Utilizarea IA generative în cadrul temelor și proiectelor este permisă, cu condiția ca studenții să respecte următoarele reguli:</p> <ul style="list-style-type: none"> IA generativă poate fi utilizată pentru generarea de idei, structuri de text sau cod, dar toate materialele generate trebuie să fie revizuite și ajustate de către student pentru a se asigura că acestea corespund cerințelor academice. Orice utilizare a IA generative trebuie să fie declarată în secțiunea de apendice a fiecărei lucrări, folosind fraza: "În timpul pregătirii acestei lucrări, autorul a utilizat [NUME INSTRUMENT / SERVICIU] în scopul [MOTIV]. După utilizarea acestui instrument/serviciu, autorul a revizuit și editat conținutul după cum a fost necesar și își asumă întreaga responsabilitate pentru conținutul lucrării."
Restricții de utilizare	<p>Studenții nu trebuie să considere IA generativă ca o sursă de încredere pentru informații, deoarece nu oferă referințe clare sau surse documentate.</p> <ul style="list-style-type: none"> Nu este permisă citarea directă a conținutului generat de IA în lucrările academice ca și cum ar fi sursă primară. Activitățile în care este interzis utilizarea IA generativă sunt specificare de profesor și sunt de regulă evaluări intermediare și finale sau care nu presupun activități de dezvoltare a competențelor profesionale.

10. Evaluare

Periodică		Curentă	Studiu individual	Proiect/teză	Examen
EP 1	EP 2				
10%	10%	10%	---	40%	60%
<p>Standard minim de performanță</p> <p>Prezența la lecții; activitatea și calitatea pregătirii la / pentru prelegeri și lucrări de laborator; Obținerea notei minime de „5” la fiecare dintre atestări și lucrări de laborator;</p> <p>Obținerea notei minime de „5” la proiectul de an (se aplică după caz);</p> <p>Demonstrarea în lucrarea de examinare finală a cunoașterii tehnicilor de programare aplicate asupra structurilor de date prin implementarea algoritmilor studiați la rezolvarea problemelor în limbajul de programare C.</p>					