

FIȘA DISCIPLINEI

Structuri de date și algoritmi

Anul universitar 2026 - 2027

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Babeș-Bolyai
1.2. Facultatea	Matematică și Informatică
1.3. Departamentul	Departamentul de Informatică
1.4. Domeniul de studii	Informatică
1.5. Ciclu de studii	Licență
1.6. Programul de studii / Calificarea	Informatică (în engleză)
1.7. Forma de învățământ	Cu frecvență

2. Date despre disciplină

2.1. Denumirea disciplinei	Structuri de date și algoritmi			Codul disciplinei	MLE5022
2.2. Titularul activităților de curs	Lect. Dr. Oneț-Marian Zsuzsanna				
2.3. Titularul activităților de seminar	Lect. Dr. Oneț-Marian Zsuzsanna				
2.4. Anul de studiu	1	2.5. Semestrul	2	2.6. Tipul de evaluare	Examen
2.7. Regimul disciplinei	Obligatoriu	2.8. Tipul disciplinei		Disciplină fundamentală (DF)	

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

3.1. Număr de ore pe săptămână	4	din care: 3.2. curs	2	3.3. seminar/ laborator/ proiect	1S + 1L
3.4. Total ore din planul de învățământ	56	din care: 3.5. curs	28	3.6 seminar/laborator	28
Distribuția fondului de timp pentru studiul individual (SI) și activități de autoinstruire (AI)					ore
Studiul după manual, suport de curs, bibliografie și notițe (AI)					17
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					6
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					30
Tutoriat (consiliere profesională)					6
Examinări					10
Alte activități					
3.7. Total ore studiu individual (SI) și activități de autoinstruire (AI)				69	
3.8. Total ore pe semestru				125	
3.9. Numărul de credite				5	

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Fundamentele programării
4.2. de competențe	Competențe medii de programare

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Sală cu videoproiector
5.2. de desfășurare a seminarului/ laboratorului	Pentru seminar este nevoie de o sală cu tablă

6.1. Competențele dobândite în urma absolvirii programului de studii (se preiau din planul de învățământ)¹

¹ Se vor prelua din Planul de învățământ al programului de studii acele competențe profesionale și/sau transversale la dezvoltarea cărora contribuie disciplina pentru care se elaborează fișa disciplinei. Pentru fiecare competență se va prelua întregul enunț, inclusiv codul competenței, cu formularea care apare în planul de

Competențe profesionale	
Codul competenței	Competență
CP7	proiectează sistemul informatic
CP19	crează modele de date
CP1	crează softuri
Competențe transversale	
Codul competenței	Competență
CT3	Gândește analitic
CT2	Soluționează probleme

6.2. Rezultatele învățării specifice programului de studii (se preiau din planul de învățământ)²

Rezultatele învățării vizate prin disciplină		
Codul competenței	Cunoștințe și înțelegere (Knowledge and understanding)	Abilități academice specifice (Specific academic skills)
CP1	1. Studentul/absolventul identifică, explică și argumentează concepte fundamentale de structuri de date, algoritmi și paradigme de programare, precum și a arhitecturii calculatoarelor.	1. Studentul/absolventul elaborează, dezvoltă și demonstrează soluții software complexe utilizând algoritmi eficienți și paradigme diverse de programare.

7. Rezultatele învățării specifice disciplinei

Cunoștințe și înțelegere (Knowledge and understanding)
1. Studentul/absolventul identifică, explică și argumentează concepte fundamentale de structuri de date, tipuri abstracte de date și algoritmi
Abilități academice specifice (Specific academic skills)
1. Studentul/absolventul elaborează, dezvoltă și demonstrează soluții software complexe utilizând algoritmi eficienți și structuri de date potrivite.

8. Conținuturi

8.1 Curs	Metode de predare - învățare	Observații ³
1. Introducere. Structuri de date. Tipuri abstracte de date. <ul style="list-style-type: none"> Tipuri abstracte de date și structuri de date Convenții de pseudocod Complexități 	<ul style="list-style-type: none"> Expunere Descriere Conversația Demonstrație didactică	
2. Vector. Iteratori <ul style="list-style-type: none"> Vector dinamic Analiza complexității amortizate Interfața unui iterator 		
3. Tipuri abstracte de date		

învățământ, fără modificări. Dacă nu se preia nici o competență din oricare din cele două categorii, se șterge linia din tabel aferentă acelei categorii.

² Se menționează rezultatele învățării specifice programului de studiu la dezvoltarea cărora contribuie disciplina pentru care se elaborează fișa. Enunțurile, preluate fără modificări din Planul de învățământ în funcție de tipul disciplinei (DF/DS/DC) se trec în dreptul competenței asociate.

³ De exemplu aspecte organizatorice, recomandări pentru studenți, aspecte specifice legate de curs/seminar cum ar fi invitarea unor practicieni în domeniu etc.

<ul style="list-style-type: none"> • TAD Mulțime: descriere, domeniu, interfață și reprezentări posibile • TAD Dicționar: descriere, domeniu, interfață și reprezentări posibile • TAD Matrice: descriere, domeniu, interfață și reprezentări posibile • TAD MultiDicționar: descriere, domeniu, interfață și reprezentări posibile 		
4. Tipuri abstracte de date II <ul style="list-style-type: none"> • TAD Stivă: descriere, domeniu, interfață și reprezentări posibile • TAD Coadă: descriere, domeniu, interfață și reprezentări posibile • TAD Coadă cu Priorități: descriere, domeniu, interfață și reprezentări posibile • TAD Deque: descriere, domeniu, interfață și reprezentări posibile • TAD Lista: descriere, domeniu, interfață și reprezentări posibile 		
5. Liste înlănțuite <ul style="list-style-type: none"> • Lista simplu înlănțuită: reprezentare și operații • Listă dublu înlănțuită: reprezentare și operații • Iterator pentru liste înlănțuite 		
6. Liste înlănțuite II <ul style="list-style-type: none"> • Liste înlănțuite ordonate: reprezentare și operații • Liste înlănțuite circulare: reprezentare și operații • Liste înlănțuite pe tablou: reprezentare și operații 		
7. Ansamblu binar <ul style="list-style-type: none"> • Definiție, reprezentări, operații specifice • HeapSort 		
8. Tabela de dispersie <ul style="list-style-type: none"> • Tabela de adresare directă • Tabela de dispersie: descriere, proprietăți • Rezolvarea coliziunilor prin liste independente 		
9. Tabela de dispersie II <ul style="list-style-type: none"> • Rezolvarea coliziunilor prin liste întrepătrunse • Rezolvarea coliziunilor prin adresare deschisă 		
10. Tabela de dispersie III <ul style="list-style-type: none"> • Dispersia perfectă • Linked hash table • Containere reprezentate pe tabele de dispersie 		
11. Arbori. Arbori binari <ul style="list-style-type: none"> • Concepte legate de arbori 		

<ul style="list-style-type: none"> • Aplicații ale arborilor • Descrierea și proprietățile arborilor binari • Domeniul și interfața TAD Arbore binar • Reprezentări posibile pentru TAD Arbore Binar • Traversarea arborelui binar: algoritmi recursivi/non-recursivi 		
12. Arbori binari de căutare <ul style="list-style-type: none"> • Descriere, proprietăți • Reprezentare • Operații: algoritmi recursivi și non-recursivi • Containere reprezentare pe arbori binari de căutare 		
13. Arbore binar de căutare echilibrată <ul style="list-style-type: none"> • Arbori AVL 		
Aplicații ale structurilor de date studiate în diferite limbaje de programare		
Bibliografie <ol style="list-style-type: none"> 1. T. Cormen, C. Leiserson, R. Rivest, C. Stein: Introduction to algorithms, Third Edition, The MIT Press, 2009 2. S. Skiena: The algorithms design manual, Second edition, Springer, 2008 3. N. Karumanchi: Data structures and algorithms made easy, CareerMonk Publications, 2016 4. M. A. Weiss: Data structures and algorithm analysis in Java, Third Edition, Pearson, 2012 5. R. Sedgwick: Algorithms, Addison-Wesley Publishing, 1984 		
8.2 Laborator	Metode de predare - învățare	Observații
Lab 1 Exemplu de rezolvare a unei teme de laborator (Demo)	<ul style="list-style-type: none"> • Expunere • Exemple • Conversație 	Laboratorul este structurat sub forma a 2 ore din 2 în 2 săptămâni. Problemele de laborator atribuite la un laborator trebuie să fie prezentate în următorul laborator (cu excepția primei teme de laborator). Fiecare laborator se concentrează pe o anumită structură de date. Studenții vor primi un container (TAD) care trebuie să fie implementat folosind structura de date dată.
Lab 2. Discuții despre Demo. Exemplu de operație extra A1 - Vector dinamic		
Lab 3. A2 - Lista înlănțuită cu alocare dinamică		
Lab 4. A3 - Lista înlănțuită pe tablou		
Lab 5. A4 - Tabela de dispersie		
Lab 6. A5 - Arbore binar de căutare		
Lab 7. Predare problema de la Lab 6.		
Bibliografie: <ol style="list-style-type: none"> 1. T. Cormen, C. Leiserson, R. Rivest, C. Stein: Introduction to algorithms, Third Edition, The MIT Press, 2009 2. S. Skiena: The algorithms design manual, Second edition, Springer, 2008 3. N. Karumanchi: Data structures and algorithms made easy, CareerMonk Publications, 2016 4. M. A. Weiss: Data structures and algorithm analysis in Java, Third Edition, Pearson, 2012 5. R. Sedgwick: Algorithms, Addison-Wesley Publishing, 1984 		
8.3 Seminar	Metode de predare - învățare	Observații
1. TAD Colecție cu element generic. Reprezentări și implementări pe vector. Iterator pentru TAD Colecție.	<ul style="list-style-type: none"> • Expunere • Conversația • Exemple • Dezbateri 	Seminarul este structurat sub forma de 2 ore din 2 în 2 săptămâni
2. Complexități		
3. Bucket Sort, Sortare lexicografică și Radix Sort. Interclasarea a două liste înlănțuite sortate.		

4. MultiDicționar ordonat – reprezentări și implementare pe o listă simplu înlănțuită		
5. Evaluarea unei expresii aritmetice. Probleme rezolvate cu ansamblu binar.		
6. Tabela de dispersie		
7. Arbori binari		
Bibliografie:		
1. T. Cormen, C. Leiserson, R. Rivest, C. Stein: Introduction to algorithms, Third Edition, The MIT Press, 2009		
2. S. Skiena: The algorithms design manual, Second edition, Springer, 2008		
3. N. Karumanchi: Data structures and algorithms made easy, CareerMonk Publications, 2016		
4. M. A. Weiss: Data structures and algorithm analysis in Java, Third Edition, Pearson, 2012		
5. R. Sedgewick: Algorithms, Addison-Wesley Publishing, 1984		

9. Evaluare



















Tip activitate	9.1 Criterii de evaluare ⁴	9.2 Metode de evaluare ⁵	9.3 Pondere din nota finală
9.4 Curs	<ul style="list-style-type: none"> Corectitudinea și completitudinea cunoștințelor asimilate. Cunoștințe de aplicare a conceptelor 	Evaluare scrisă (în sesiune): examen scris	60%
9.5 Laborator	<ul style="list-style-type: none"> Implementarea în C++ a conceptelor și algoritmilor prezentați la cursuri Documentarea lucrărilor de laborator Respectarea termenelor de prezentare la laborator 	Corectitudinea implementării și a documentației (reprezentare, specificații, algoritmi, complexități).	40%
9.6 Seminar	<ul style="list-style-type: none"> Activitatea de seminar 	Participare activă la discuții în timpul seminarului	Maximum 0.5 puncte bonus adăugate la nota finală
9.7 Standard minim de promovare			
<ul style="list-style-type: none"> Cunoașterea conceptelor de bază. Fiecare student trebuie să demonstreze că a dobândit un nivel acceptabil de cunoștințe și înțelegere a domeniului, că este capabil să exprime cunoștințele dobândite într-o formă coerentă și că are capacitatea de a utiliza aceste cunoștințe pentru rezolvarea problemelor. Pentru a participa la examenul scris, un student trebuie să aibă cel puțin 6 prezențe la laborator și 5 prezențe la seminar. Pentru a promova cu succes examenul, un student trebuie să aibă cel puțin nota 5 la examenul scris și cel puțin nota 5 ca medie finală. 			

10. Etichete ODD (Obiective de Dezvoltare Durabilă / Sustainable Development Goals)⁶

⁴ Criteriile de evaluare trebuie să reflecte direct rezultatele învățării vizate la nivel de program de studii, respectiv la nivel de disciplină. Mai concret, se evaluează achizițiile de învățare menționate în rezultatele anticipate ale învățării.

⁵ Se recomandă stabilirea atât a metodelor de evaluare finală, cât și a strategiei de evaluare pe parcurs.

⁶ Selectați o singură etichetă, cea care, în conformitate cu [Procedura de aplicare a etichetelor ODD în procesul academic](#), se potrivește cel mai bine disciplinei. Dacă disciplina tratează tema dezvoltării durabile la modul general (de ex. prin prezentarea/introducerea cadrului general al dezvoltării durabile etc.) atunci se poate alocă eticheta

	Eticheta generală pentru Dezvoltare durabilă							
								
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
								Nu se aplică nici o etichetă
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	X

Data completării:

22.05.2026

Semnătura titularului de curs

Lect. PhD. ONEȚ-MARIAN Zsuzsanna

Semnătura titularului de seminar

Lect. PhD. ONEȚ-MARIAN Zsuzsanna

Data avizării în departament:

...

Semnătura directorului de departament

Conf. PhD. STERCA Adrian

generală de Dezvoltare Durabilă. Dacă niciuna dintre etichete nu descrie disciplina, selectați ultima opțiune: „Nu se aplică nici o etichetă”.