

FIȘA DISCIPLINEI

Gândire algoritmică

Anul universitar 2025-2026

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Babeș-Bolyai Cluj-Napoca
1.2. Facultatea	Facultatea de Matematică și Informatică
1.3. Departamentul	Departamentul de Informatică
1.4. Domeniul de studii	Informatică
1.5. Ciclu de studii	Master
1.6. Programul de studii / Calificarea	Știința datelor în industrie și societate
1.7. Forma de învățământ	Cu frecvență

2. Date despre disciplină

2.1. Denumirea disciplinei	Gândire Algoritmică			Codul disciplinei	MME8181
2.2. Titularul activităților de curs	Conf. Dr. Camelia Șerban				
2.3. Titularul activităților de seminar	Conf. Dr. Camelia Șerban				
2.4 Anul de studiu	1	2.5 Semestrul	2	2.6. Tipul de evaluare	E
				2.7 Regimul disciplinei	Obligativu

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

3.1 Număr de ore pe săptămână	3	Din care: 3.2 curs	2	3.3 seminar/laborator	1sem
3.4 Total ore din planul de învățământ	42	Din care: 3.5 curs	28	3.6 seminar/laborator	14
Distribuția fondului de timp pentru studiul individual (SI) și activități de autoinstruire (AI)					ore
Studiul după manual, suport de curs, bibliografie și notițe (AI)					35
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					32
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri (mai mare sau egal cu nr. total ore prevăzut în calendarul disciplinei pentru temele de control)					54
Tutoriat (consiliere profesională)					4
Examinări					8
Alte activități [de ex.: comunicare bidirecțională cu titularul de disciplină / tutorele]					-
3.7. Total ore studiu individual (SI) și activități de autoinstruire (AI)				133	
3.8. Total ore pe semestru				175	
3.9. Numărul de credite				7	

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	-
4.2. de competențe	-

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Videoproiector
5.2. de desfășurare a seminarului/ laboratorului	Calculatoare, medii specifice de dezvoltare și implementare

6.1. Competențele specifice acumulate¹

¹ Se poate opta pentru competențe sau pentru rezultatele învățării, respectiv pentru ambele. În cazul în care se alege o singură variantă, se va șterge tabelul aferent celeilalte opțiuni, iar opțiunea păstrată va fi numerotată cu 6.

Competențe profesionale/esențiale	C1.1 Descrierea adecvată a paradigmelor de programare și a mecanismelor de limbaj specifice, precum și identificarea diferenței dintre aspectele de ordin semantic și sintactic. C1.3 Elaborarea codurilor sursă adecvate și testarea unitară a unor componente într-un limbaj de programare cunoscut, pe baza unor specificații de proiectare date C1.4 Testarea unor aplicații pe baza unor planuri de test C1.5 Dezvoltarea de unități de program și elaborarea documentațiilor aferente
Competențe transversale	CT1 Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar și dezvoltarea capacităților empatică de comunicare inter-personală, de relaționare și colaborare cu grupuri diverse

6.2. Rezultatele învățării

Cunoștințe	Studentul are cunoștințe necesare pentru utilizarea calculatoarelor, dezvoltarea programelor și aplicațiilor software, procesarea informațiilor. Studentul are cunoștințe legate de programare, matematică, inginerie și tehnologie și are abilitățile necesare pentru a le folosi în crearea de sisteme informatice complexe. Studentul are cunoștințele necesare pentru proiectarea, analiza și administrarea bazelor de date.
Aptitudini	Studentul are abilitatea de a dezvolta, proiecta și crea noi aplicații, sisteme sau produse folosind bunele practici din domeniu. Studentul are aptitudinile necesare pentru conceperea programelor de calculator și analiza sistemelor software. Studentul are abilitatea de a aplica reguli generale unor probleme specifice și de a produce soluții relevante. Studentul are aptitudinile necesare pentru înțelegerea și folosirea conceptelor programării orientate obiect la dezvoltarea unor aplicații software de complexitate medie. Studentul are abilitatea de a înțelege și folosi șabloanele de proiectare pentru dezvoltarea aplicațiilor. Studentul are deprinderile necesare pentru a dezvolta aplicații cu interfețe grafice folosind șabloane arhitecturale potrivite pentru specificul aplicațiilor cu interacțiune utilizator.
Responsabilități și autonomie	Studentul este capabil să identifice probleme complexe și să examineze probleme conexe pentru a dezvolta opțiuni de rezolvare și implementa soluții. Studentul este capabil să combine informații diverse pentru a formula soluții și genera idei de dezvoltare pentru noi produse și aplicații. Studentul este capabil să aplice șabloane arhitecturale, șabloane de proiectare și bunele practici în domeniu pentru a proiecta aplicații software de complexitate mare. Studentul a dobândit capacitatea de a evalua diferite arhitecturi și soluții posibile pentru o problemă. Studentul are capacitatea de a alege și folosi paradigme de programare pentru realizarea de aplicații software. Studentul are capacitatea de a observa și obține informații din diverse surse.

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	Dezvoltare gândirii algoritmice, a conceptelor, metodelor și tehnicilor implicate în rezolvarea unor probleme.
7.2 Obiectivele specifice	Studentul va înțelege modul în care gândirea algoritmică poate fi utilizată de cercetători pentru a organiza datele structurate și nestructurate în abordarea problemelor dintr-un anumit domeniu.

8. Conținuturi

8.1 Curs		
1. Introducere in gandirea algoritmica	Metode de predare Expunerea interactivă Prezentarea Explicarea Exemple practice Discuții pe studii de caz	Observații
2. Functii		
3. Testare. Tratarea Exceptiilor		
4. Tipuri structurate.		
5. Cautare si Sortare		
6. Programarea modulara		
7. Tipuri definite de utilizator		
8. Lambda		
9. Introducere in biblioteci necesare in Stiinta Datelor: Python: Pandas data-frames; Matplotlib plotting		
10-11 Statstica pentru Data Science		
12-14 Python nivel intermediar pentru stiinta datelor		
Bibliografie		
1. Kleinberg and Tardos – <i>Algorithm Design</i> . Pearson Educational, 2014		
2. (http://www.cs.princeton.edu/~wayne/kleinberg-tardos/)		
3. <i>The Python language reference</i> . (https://docs.python.org/3/reference/index.html)		
4. <i>The Python standard library</i> . (https://docs.python.org/3/library/index.html)		
5. <i>The Python tutorial</i> . (https://docs.python.org/3/tutorial/index.html)		
6. Kent Beck - <i>Test Driven Development: By Example</i> . Addison-Wesley Longman, 2002.		
8.2 Seminar / laborator		
Aplicatii simple	Metode de predare Expunerea interactivă Explicarea Conversația Demonstrația didactică	Observații
Aplicatii simple		
Aplicatii simple		
Programarea modulara		
Lambda		
Introducere in bibliotecile Python pt Stiinta Datelor		
Introducere in Statistical Thinking		
Bibliografie		
1. Kleinberg and Tardos – <i>Algorithm Design</i> . Pearson Educational, 2014		
2. (http://www.cs.princeton.edu/~wayne/kleinberg-tardos/)		
3. <i>The Python language reference</i> . (https://docs.python.org/3/reference/index.html)		
4. <i>The Python standard library</i> . (https://docs.python.org/3/library/index.html)		
5. <i>The Python tutorial</i> . (https://docs.python.org/3/tutorial/index.html)		
6. Kent Beck - <i>Test Driven Development: By Example</i> . Addison-Wesley Longman, 2002.		

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

- Acest curs există în programul de studiu al tuturor universităților importante din România și străinătate
- Conținutul acestui curs este considerat important de către companiile de IT

10. Evaluare

Tip activitate	Tip activitate	Tip activitate	Tip activitate
10.4 Curs	10.4 Curs	10.4 Curs	10.4 Curs
10.5 Seminar/laborator	10.5 Seminar/laborator	10.5 Seminar/laborator	10.5 Seminar/laborator
10.6 Standard minim de performanță	10.6 Standard minim de performanță	10.6 Standard minim de performanță	10.6 Standard minim de performanță
	Fiecare student trebuie să obțină minim nota 5	Fiecare student trebuie să obțină minim nota 5	Fiecare student trebuie să obțină minim nota 5
Tip activitate			
• Minimum 5 la fiecare proba.			

11. Etichete ODD (Obiective de Dezvoltare Durabilă / Sustainable Development Goals)²

Nu se aplică.

Data completării:
15.04.2025

Semnătura titularului de curs

Conf. Dr. Camelia Șerban

Semnătura titularului de seminar

Conf. Dr. Camelia Șerban

Data avizării în departament:

Semnătura directorului de departament

...

² Păstrați doar etichetele care, în conformitate cu [Procedura de aplicare a etichetelor ODD în procesul academic](#), se potrivesc disciplinei și ștergeți-le pe celelalte, inclusiv eticheta generală pentru *Dezvoltare durabilă* - dacă nu se aplică. Dacă nicio etichetă nu descrie disciplina, ștergeți-le pe toate și scrieți "*Nu se aplică*".

