

FIȘA DISCIPLINEI

Algoritmi și programare

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 Matematică și Informatică al Liniei Maghiare
1.4. Domeniul de studii	Informatică
1.5. Ciclul de studii	Licență
1.6. Programul de studii / Calificarea	Matematică
1.7. Forma de învățământ	Cu frecvență

2. Date despre disciplină

2.1. Denumirea disciplinei		Algoritmi și programare				Codul disciplinei		MLM5108			
2.2. Titularul activităților de curs			VASS Balázs								
2.3. Titularul activităților de seminar			VASS Balázs								
2.4. Anul de studiu		1	2.5. Semestrul		1	2.6. Tipul de evaluare		C	2.7. Regimul disciplinei		Obligativu

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	1/1
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)					2
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					2
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					14
Tutoriat (consiliere profesională)					1
Examinări					
Alte activități					
3.7. Total ore studiu individual (SI) și activități de autoinstruire (AI)				19	
3.8. Total ore pe semestru				75	
3.9. Numărul de credite				3	

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Nu e cazul
4.2. de competențe	Înțelegerea enunțurilor problemelor, noțiunile de bază ale matematicii

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Sală de curs cu tablă și videoproiector
5.2. de desfășurare a seminarului/ laboratorului	- Sală de cu tablă și videoproiector (pentru seminar) - Sală de laborator, cu C/C++ instalat pe calculatoare (pentru laborator)

6. Competențele specifice acumulate

Competențe profesionale/esențiale	• C1.1 Identificarea notiunilor, descrierea teoriilor și utilizarea limbajului specific. • C1.4 Recunoașterea principalelor clase/tipuri de probleme matematice și selectarea metodelor și a tehnicilor adecvate pentru rezolvarea lor. • C2.1 Identificarea notiunilor de baza utilizate în descrierea unor fenomene și procese. • C2.2 Interpretarea rezultatelor prelucrării datelor. • C2.4 Analiza comparativă a rezultatelor obținute prin rezolvarea problemelor cu datele preexistente. • C2.5 Elaborarea și prezentarea unor proiecte și/sau lucrări vizând rezultatele obținute prin prelucrarea datelor. • C3.1 Identificarea notiunilor de baza folosite în construcția și specificarea algoritmilor. • C3.2 Interpretarea datelor și explicarea etapelor care intervin în probleme rezolvabile prin algoritmi. • C3.3 Aplicarea tehnicilor și metodelor specifice pentru proiectarea unor algoritmi. • C3.4 stabilirea avantajelor și limitelor unui algoritm dat, utilizând noțiuni și termeni specifici. • C3.5 Elaborarea unor proiecte vizând rezolvarea problemelor prin algoritmi.
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 empatice de comunicare inter-personală, de relaționare și colaborare cu grupuri diverse. • CT3 Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare în limba română și într-o limbă de circulație internațională.

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	• Înțelegerea problemelor informatice, modelare • Dezvoltarea creativității • Dezvoltarea abilităților de rezolvare a problemelor
7.2 Obiectivele specifice	• Dezvoltarea gândirii disciplinate, logice și algoritmice • Învățarea și practicarea metodelor de programare • Înțelegerea principiilor de bază ale designului de software

8. Conținuturi

8.1 Curs	Metode didactice	Observații
1. Rezolvarea problemelor în mediul informatic, noțiunea de algoritm	Prelegere, exemple	[2], cap. 1.
2. Reprezentarea algoritmilor, programarea structurată	Prelegere, exemple	[2], cap. 2.
3. Rafinare în pași succesivi	Prelegere, exemple	[2], cap. 3.
4. Algoritmi cu vectori de date	Prelegere, exemple	[2], cap. 4.
5. Subalgoritmi. Programarea modulară	Prelegere, exemple	[2], cap. 5.
6. Algoritmi de sortare (1)	Prelegere, exemple	[2], cap. 6.

7. Algoritmi de sortare (2)	Prelegere, exemple	[2], cap. 6.
8. Recursivitate	Prelegere, exemple	[2], cap. 7.
9. Metoda backtracking	Prelegere, exemple	[2], cap. 8.
10. Metoda divide et impera (1)	Prelegere, exemple	[2], cap. 9.
11. Metoda divide et impera (2)	Prelegere, exemple	[2], cap. 9.
12. Metoda greedy	Prelegere, exemple	[2], cap. 10.
13. Programare dinamică	Prelegere, exemple	[4], [5]
14. Elemente din teoria complexității algoritmilor. Aplicații în criptografie	Prelegere, exemple	[4], [5]
Bibliografie [1] Cormen T., Leiserson C., Rivest R., Stein, C. – Új algoritmusok, Sclolar, Budapest, 2003. [2] Ionescu K. – Bevezetés az algoritmikába, Egyetemi Könyvkiadó, Kolozsvár, 2007 [3] Kása Z. – Algoritmusok tervezése, Stúdium Könyvkiadó, Kolozsvár, 1994. [4] Knuth D. E. – A számítógép-programozás művészete, I, II, III kötet, 1992. [5] Rónyai, L., Ivanyos, G., Szabó, R. – Algoritmusok, Typotex, Budapest, 1999. [6] Wirth N. – Algorithms + Data Structures = Programs, Prentice Hall Inc., 1976.		
8.2 Seminar/Laborator	Metode didactice	Observații
1. O scurtă prezentare a următoarelor concepte: arhitectura calculatorului, fișiere executabile, cod sursă, compilator. Mediului de dezvoltare C / C ++	Problematizarea, conversația	
2. Reprezentarea algoritmilor (diagrama, pseudocod). Bazele limbajelor de programare C/C++	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
3. Date de intrare și ieșire, variabile, structuri de control în C/C++	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
4. Vectori de date, prelucrarea secvențială a datelor	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
5. Programare cu funcții în C/C++	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
6. Algoritmi de sortare (1)	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
7. Algoritmi de sortare (2)	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
8. Programare recursivă	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
9. Metoda backtracking	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
10. Metoda divide et impera (1)	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
11. Metoda divide et impera (2)	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
12. Metoda greedy	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
13. Programare dinamică	Problematizarea, conversația, rezolvarea problemelor practice (programare)	
14. Elemente din teoria complexității algoritmilor. Aplicații în criptografie	Problematizarea, conversația, rezolvarea problemelor practice (programare)	

Bibliografie

- [1] Cormen T., Leiserson C., Rivest R., Stein, C. – Új algoritmusok, Scolar, Budapest, 2003.
- [2] Ionescu K. – Bevezetés az algoritmikába, Egyetemi Könyvkiadó, Kolozsvár, 2007
- [3] Kása Z. – Algoritmusok tervezése, Stúdium Könyvkiadó, Kolozsvár, 1994.
- [4] Knuth D. E. – A számítógép-programozás művészete, I, II, III kötet, 1992.
- [5] Rónyai, L., Ivanyos, G., Szabó, R. – Algoritmusok, Typotex, Budapest, 1999.
- [6] Wirth N. – Algorithms + Data Structures = Programs, Prentice Hall Inc., 1976.

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

- Conținutul cursului coincide cu cel tradițional al unui curs de introducere în teoria algoritmilor (și introducere în programare) predat la universitățile majore din învățământul universitar
- În contextul acestui curs, luăm în considerare posibilitățile de utilizare a calculatorului pentru a studia problemele matematice.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Cunoașterea noțiunilor și algoritmilor de bază	- Examen parțial la mijlocul semestrului (25%) - Examen parțial la sfârșitul semestrului (25%)	50%
10.5 Seminar/laborator	Corectitudinea implementării, stilul de programare, eficiența programelor dezvoltate	- Probleme practice (de programare în C/C++) în cadrul orelor de laborator (30%) - Teme pentru acasă (20%)	50%
10.6 Standard minim de performanță			
• Înțelegerea, cunoașterea implementarea algoritmilor elementari în C/C++ • Rezolvarea unor problemei mai simple în stil recursiv • Cunoașterea metodelor de programare backtracking, divide et impera, greedy			

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

Nu se aplică.

Data completării:

24. septembrie 2025.

Semnătura titularului de curs

conf. univ. dr. Balázs VASS

Semnătura titularului de seminar

conf. univ. dr. Balázs VASS

Data avizării în departament:

24. septembrie 2025.

Semnătura directorului de departament

conf. univ. dr. Szilárd-Károly ANDRÁS