



Grafuri și arbori

Consultații pentru concursul Mate-Info UBB 2026

stud. Rareș-Andrei Cotoi

March 7, 2026



Facultatea de
Matematică și Informatică
UBB Cluj-Napoca



Cuprins

1 Recapitulare - Need to know

► Recapitulare - Need to know

► Recapitulare - Nice to know

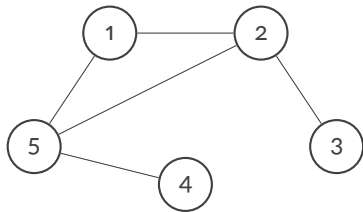
► Probleme



Graf neorientat

1 Recapitulare - Need to know

Un **graf neorientat** este o pereche $G = (V, U)$, unde V este o mulțime finită, nevidă, de **vârfuri** (noduri), iar U este o mulțime de submulțimi cu două elemente din V , numite **muchii**.



$$V = \{1, 2, 3, 4, 5\}$$

$$U = \{[1, 2], [1, 5], [2, 3], [2, 5], [4, 5]\}$$

Vârfurile 1 și 2 sunt **adiacente**.

Muchiile $[1, 2]$ și $[2, 3]$ sunt **incidente** (au extremitatea comună 2).

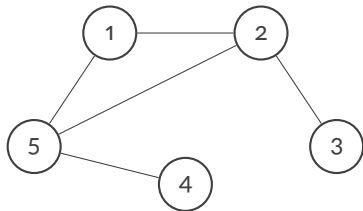
Vecinii lui 2: $\{1, 3, 5\}$.



Gradul unui vârf

1 Recapitulare - Need to know

Gradul unui vârf x , notat $d(x)$, este numărul de vârfuri adiacente cu x (echivalent, numărul de muchii incidente cu x).



$$d(1) = 2, \quad d(2) = 3, \quad d(3) = 1 \\ d(4) = 1, \quad d(5) = 3$$

Vârf izolat: $d(x) = 0$

Vârf terminal: $d(x) = 1$ (ex: 3, 4)

Grad maxim posibil: $n - 1$

Teoremă: $\sum_{x \in V} d(x) = 2m$, unde $m = |E|$.

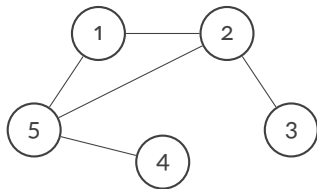
Consecință: Numărul de vârfuri de grad impar este un număr **par**.



Matricea de adiacență

1 Recapitulare - Need to know

Matricea de adiacență a unui graf neorientat $G = (V, U)$ cu n vârfuri este o matrice A de dimensiune $n \times n$, unde $A_{i,j} = \begin{cases} 1, & \text{dacă } [i,j] \in U \\ 0, & \text{altfel} \end{cases}$



$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

- Matricea este **simetrică** față de diagonala principală.
- Diagonala principală conține doar 0.
- $d(x)$ = suma elementelor de pe linia (sau coloana) x .



Graf parțial. Subgraf. Graf complementar

1 Recapitulare - Need to know

Fie $G = (V, U)$ un graf neorientat cu n vârfuri și m muchii.

Graf parțial $G_1 = (V, U_1)$, unde $U_1 \subseteq U$: se păstrează toate vârfurile, se elimină eventual muchii.

Subgraf $G_1 = (V_1, U_1)$, unde $V_1 \subseteq V$ și U_1 conține toate muchiile din U cu ambele extremități în V_1 : se elimină eventual vârfuri și muchiile incidente cu ele.

Graf complementar $\bar{G} = (V, \bar{U})$: muchia $[x, y] \in \bar{U} \Leftrightarrow [x, y] \notin U$.

Formule:

- Număr de **grafuri parțiale**: 2^m (orice submulțime a muchiilor)
- Număr de **subgrafuri**: $2^n - 1$ (orice submulțime nevidă a vârfurilor)
- Graful complementar este **unic**



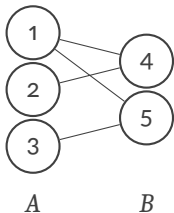
Graf complet. Graf bipartit

1 Recapitulare - Need to know

Graf complet K_n : oricare două vârfuri distincte sunt adiacente. $\Rightarrow C_n^2 = \frac{n(n-1)}{2}$ muchii, $d(x) = n - 1 \forall x \in V$.

Nr. grafuri neorientate distincte cu n vârfuri: $2^{\frac{n(n-1)}{2}}$ (fiecare pereche poate fi sau nu muchie).

Graf bipartit: $\exists A, B \subseteq V$ cu $A \cup B = V$, $A \cap B = \emptyset$, iar orice muchie are o extremitate în A și cealaltă în B .



$$A = \{1, 2, 3\}, B = \{4, 5\}$$

Bipartit complet: $\forall x \in A, y \in B, [x, y] \in U$.



Lanț. Ciclu

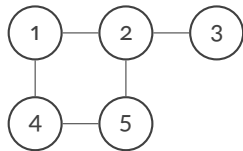
1 Recapitulare - Need to know

Lanț: succesiune de vârfuri $L = [x_1, x_2, \dots, x_k]$ în care oricare două consecutive sunt adiacente. Lungime: $k - 1$.

Lanț elementar: vârfuri distincte. **Lanț simplu:** muchii distincte.

Ciclu: lanț simplu cu $x_1 = x_k$. Lungimea minimă: 3.

Ciclu elementar: toate vârfurile distincte, exceptând $x_1 = x_k$.



$[1, 2, 3]$ – lanț elementar

$[1, 2, 5, 4, 1]$ – ciclu elementar

$[4, 1, 2, 3]$ – lanț elementar

Un graf fără cicluri: **aciclic**.



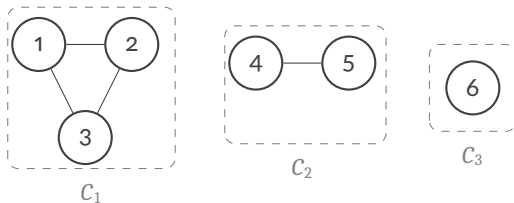
Conexitate. Componente conexe

1 Recapitulare - Need to know

Un graf neorientat este **conex** dacă pentru oricare două vârfuri x, y există cel puțin un lanț de la x la y .

O **componentă conexă** este un subgraf conex **maximal** (nu se mai poate adăuga un vârf fără a pierde conexitatea).

Un graf este conex $\Leftrightarrow$ are o singură componentă conexă.



Graful de mai sus are **3 componente conexe**: $C_1 = \{1, 2, 3\}$, $C_2 = \{4, 5\}$, $C_3 = \{6\}$.

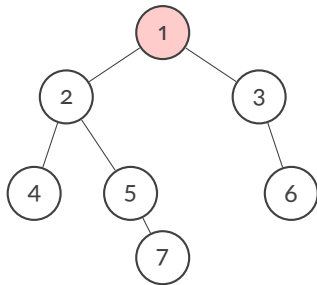


Parcurgerea în lăţime (BFS)

1 Recapitulare - Need to know

BFS (Breadth-First Search) explorează graful **nivel cu nivel**, pornind de la un vârf sursă, folosind o **coadă**.

La fiecare pas, se extrage un vârf din coadă şi se adaugă vecinii **nevizitaţi**.



BFS din vârful 1:

Nivel 0: $\{1\}$

Nivel 1: $\{2, 3\}$

Nivel 2: $\{4, 5, 6\}$

Nivel 3: $\{7\}$

Ordine vizitare: 1, 2, 3, 4, 5, 6, 7

Utilă pentru: distanţe minime, componente conexe.

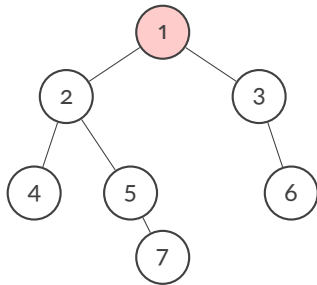


Parcurgerea în adâncime (DFS)

1 Recapitulare - Need to know

DFS (Depth-First Search) explorează graful cât mai **în adâncime**, pornind de la un vârf sursă, folosind o **stivă** (sau recursivitate).

Se avansează pe un vecin nevizitat; dacă nu mai există, se revine la vârful anterior.



DFS din vârful 1:

$1 \rightarrow 2 \rightarrow 4$ (4 nu are vecini nevizitați)
 $\rightarrow 5 \rightarrow 7$ (7 nu are vecini nevizitați)
 $\rightarrow 3 \rightarrow 6$ (6 nu are vecini nevizitați)

Ordine vizitare: 1, 2, 4, 5, 7, 3, 6

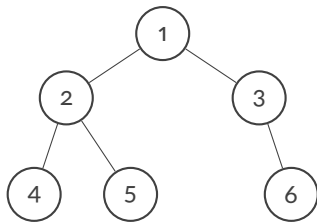
Utilă pentru: componente conexe, detecția ciclurilor.



Arbori

1 Recapitulare - Need to know

Un **arbore** este un graf **conex** și **aciclic**.



Proprietăți:

- Un arbore cu n vârfuri are exact $n - 1$ muchii.
- Este conex **minimal**: eliminarea oricărei muchii **deconectează** graful.
- Este aciclic **maximal**: adăugarea oricărei muchii **crează** un ciclu.
- Între oricare două vârfuri există un **unic** lanț.

O **pădure** este un graf aciclic (fiecare componentă conexă este un arbore).

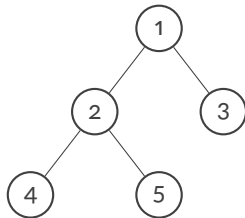
Un **arbore parțial** al unui graf G este un graf parțial al lui G care este arbore.



Arbori binari

1 Recapitulare - Need to know

Un **arbore binar** este un arbore cu rădăcină în care fiecare nod are **cel mult 2 copii**: copilul **stâng** și copilul **drept**.



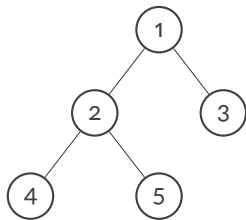
Traversări (vizitarea tuturor nodurilor):

- **Preordine** (RSD): rădăcină, stânga, dreapta
- **Inordine** (SRD): stânga, rădăcină, dreapta
- **Postordine** (SDR): stânga, dreapta, rădăcină



Traversări – exemplu

1 Recapitulare - Need to know



Preordine ($R \rightarrow S \rightarrow D$):

1, 2, 4, 5, 3

Inordine ($S \rightarrow R \rightarrow D$):

4, 2, 5, 1, 3

Postordine ($S \rightarrow D \rightarrow R$):

4, 5, 2, 3, 1



Cuprins

2 Recapitulare - Nice to know

► Recapitulare - Need to know

► Recapitulare - Nice to know

► Probleme



Graf eulerian

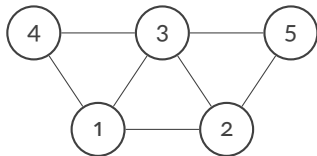
2 Recapitulare - Nice to know

Ciclu eulerian: ciclu care conține **toate muchiile** grafului (fiecare muchie apare exact o dată).

Graf eulerian: graf care conține un ciclu eulerian.

Teoremă: Un graf neorientat fără vârfuri izolate este eulerian $\Leftrightarrow$ este **conex** și **toate** vârfurile au grad **par**.

Lanț eulerian: lanț simplu care conține toate muchiile. Există $\Leftrightarrow$ graful este conex și are exact **2 vârfuri de grad impar** (acestea vor fi extremitățile lanțului).



$$d(1) = 3, d(2) = 3, d(3) = 4$$

$$d(4) = 2, d(5) = 2$$

Vârfuri de grad impar: 1, 2 (exact 2)

$\Rightarrow$ **nu** este eulerian, dar admite

lanț eulerian: $[1, 4, 3, 5, 2, 3, 1, 2]$



Graf hamiltonian

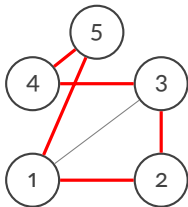
2 Recapitulare - Nice to know

Ciclu hamiltonian: ciclu elementar care conține **toate vârfurile** grafului.

Graf hamiltonian: graf care conține un ciclu hamiltonian.

Condiții suficiente (dacă nu sunt îndeplinite, **nu** înseamnă că graful nu e hamiltonian):

- Dacă $n \geq 3$ și $d(x) \geq \frac{n}{2}$ pentru $\forall x \in V$, atunci G este hamiltonian.
- **Teorema Ore:** Dacă $d(x) + d(y) \geq n$ pentru orice x, y neadiacente, atunci G este hamiltonian.



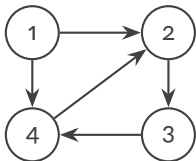
Ciclu hamiltonian (roșu):
[1, 2, 3, 4, 5, 1]



Graf orientat

2 Recapitulare - Nice to know

Un **graf orientat** (digraf) este o pereche $G = (V, U)$, unde U conține **perechi ordonate** (x, y) numite **arce**. Arcul $(x, y) \neq (y, x)$.



$$V = \{1, 2, 3, 4\}$$

$$U = \{(1, 2), (1, 4), (2, 3), (3, 4), (4, 2)\}$$

Grad interior $d^-(x)$: nr. arce care **intră** în x

Grad exterior $d^+(x)$: nr. arce care **ies** din x

$$d^+(1) = 2, d^-(1) = 0$$

$$d^+(2) = 1, d^-(2) = 2$$

Teoremă: $\sum_{x \in V} d^+(x) = \sum_{x \in V} d^-(x) = m$

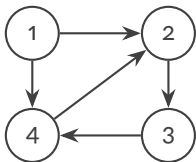


Graf orientat – matricea de adiacență

2 Recapitulare - Nice to know

Matricea de adiacență a unui graf orientat **nu este neapărat simetrică**.

$$A_{i,j} = \begin{cases} 1, & \text{dacă } (i,j) \in U \\ 0, & \text{altfel} \end{cases}$$



$$A = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

$d^+(x)$ = suma pe linia x

$d^-(x)$ = suma pe coloana x

Nr. grafuri orientate cu n noduri: $4^{\frac{n(n-1)}{2}}$ (fiecare pereche: 4 variante).

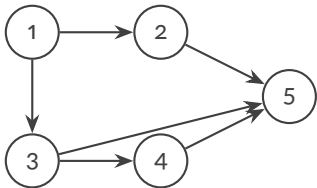


DAG. Sortare topologică

2 Recapitulare - Nice to know

DAG (Directed Acyclic Graph): graf orientat **fără cicluri**.

Sortare topologică: ordonare liniară a vârfurilor astfel încât, pentru orice arc (u, v) , u apare **înaintea** lui v . Posibilă **doar** în DAG-uri.



O sortare topologică validă:

1, 2, 3, 4, 5

Altă sortare validă:

1, 3, 4, 2, 5

Algoritmul: se folosește BFS (algoritmul lui Kahn): se pornește de la nodurile cu $d^- = 0$, se elimină, și se repetă.

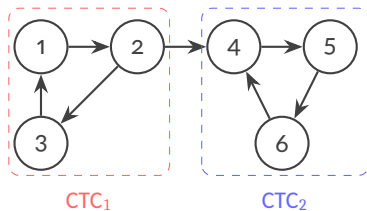


Tare conexitate

2 Recapitulare - Nice to know

Un graf orientat este **tare conex** dacă $\forall x, y \in V$ (distincte), există drum de la x la y și de la y la x .

O **componentă tare conexă** (CTC) este un subgraf tare conex **maximal**.



$$CTC_1 = \{1, 2, 3\}$$

$$CTC_2 = \{4, 5, 6\}$$

Graful **nu** este tare conex (nu există drum de la 4 la 1).

Algoritm: Kosaraju (2 parcurgeri DFS: una pe graful inițial, una pe graful **transpus**).



Cheat sheet – formule

2 Recapitulare - Nice to know

Grafuri neorientate:

- $\sum d(x) = 2m$; nr. vârfuri grad impar: par
- K_n : $m = \frac{n(n-1)}{2}$, $d(x) = n - 1$
- Nr. grafuri distincte, n vârfuri: $2^{\frac{n(n-1)}{2}}$
- Nr. grafuri parțiale: 2^m
- Nr. subgrafuri: $2^n - 1$
- Arbore, n vârfuri: $n - 1$ muchii
- Eulerian $\Leftrightarrow$ conex + $d(x)$ par $\forall x$
- Lanț eulerian $\Leftrightarrow$ conex + exact 2 cu d impar

Grafuri neorientate (cont.):

- Hamilton (suf.): $d(x) \geq \frac{n}{2}$, $\forall x$, $n \geq 3$
- Ore: $d(x) + d(y) \geq n$, $\forall x, y$ neadiac.
- Cicluri hamilton. în K_n : $\frac{(n-1)!}{2}$

Grafuri orientate:

- $\sum d^+(x) = \sum d^-(x) = m$
- Nr. grafuri orientate, n noduri: $4^{\frac{n(n-1)}{2}}$
- Tare conex $\Leftrightarrow \forall x, y: \exists$ drum $x \rightarrow y$ și $y \rightarrow x$
- Sortare topologică: doar în DAG



Cuprins

3 Probleme

► Recapitulare - Need to know

► Recapitulare - Nice to know

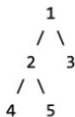
► Probleme



Problema 1

3 Probleme

10. Se consideră următorul arbore binar:



Care dintre următoarele șiruri de noduri corespund traversării arborelui în preordine?

- A. 1, 2, 4, 5, 3
- B. 4, 2, 5, 1, 3
- C. 1, 2, 3, 4, 5
- D. 4, 5, 2, 3, 1

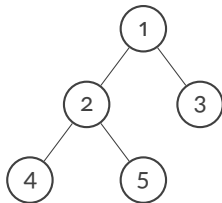
Sursa: Admitere UBB iulie 2024



Problema 1 – soluție

3 Probleme

Arborele binar:



Traversarea în **preordine** (RSD: rădăcină → stânga → dreapta):

$\underbrace{1}$, $\underbrace{2, 4, 5}$, $\underbrace{3}$
rădăcină subarbore stâng subarbore drept

Rezultat: 1, 2, 4, 5, 3 $\Rightarrow$ Răspunsul corect este **A**.



Problema 2

3 Probleme

7. Se consideră următorul arbore binar:



Care dintre următoarele șiruri de noduri corespund traversării arborelui în postordine?

A. 2, 3, 4, 5, 6, 8, 9

B. 4, 3, 2, 9, 8, 6, 5

C. 2, 4, 3, 6, 9, 8, 5

D. 9, 6, 8, 5, 3, 2, 4

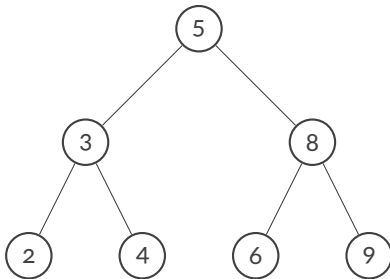
Sursa: Admitere UBB septembrie 2024



Problema 2 – soluție

3 Probleme

Arborele binar:



Traversarea în **postordine** (SDR: stânga → dreapta → rădăcină):

$\underbrace{2, 4, 3}$, $\underbrace{6, 9, 8}$, $\underbrace{5}$. Rezultat: 2, 4, 3, 6, 9, 8, 5 $\Rightarrow$ Răspunsul corect este **C**.
subarbore stâng subarbore drept rădăcină



Problema 3

3 Probleme

13. Construim un graf neorientat în modul următor: pentru fiecare număr natural n , astfel încât $2 \leq n \leq 20$, adăugăm un nod etichetat cu acel număr, iar între două noduri cu etichete x și y adăugăm o muchie, dacă y este divizor propriu al lui x .

Care dintre următoarele afirmații sunt adevărate?

- A. Graful construit are 5 componente conexe.
- B. Doar nodurile 12, 18 și 20 au gradul maxim.
- C. Pentru oricare 2 numere neprime x, y ($2 \leq x \leq y \leq n$), gradul lui x este mai mare sau egal decât gradul lui y .
- D. Există un singur nod cu gradul 1.

Sursa: Admitere UBB iulie 2025



Problema 3 – soluție

3 Probleme

Nodurile grafului: $\{2, 3, \dots, 20\}$. Muchia $[x, y]$ există dacă $y|x$ și $y \neq x$.

A. Graful are 5 componente conexe.

Componenta principală conține toate numerele compuse și factorii lor primi din $[2, 20]$: $\{2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 14, 15, 16, 18, 20\}$ (conectate prin relații de divizibilitate).

Numerele prime $\{11, 13, 17, 19\}$ sunt **izolate** (multiplii lor depășesc 20). Total: $1 + 4 = 5$ componente. **Adevărat.**

B. Doar nodurile 12, 18 și 20 au gradul maxim.

$d(2) = 9$ (2 este divizor pentru 4, 6, 8, 10, 12, 14, 16, 18, 20), deci nodul 2 are gradul maxim, nu 12, 18, 20 (care au $d = 4$). **Fals.**



Problema 3 – soluție

3 Probleme

C. Pentru oricare 2 numere neprime x, y ($2 \leq x \leq y \leq n$), $d(x) \geq d(y)$.

Contraexemplu: $x = 9$ (compus), $y = 10$ (compus), $9 \leq 10$.

$d(9) = 2$ (muchii: $[9, 3], [18, 9]$), $d(10) = 3$ (muchii: $[10, 2], [10, 5], [20, 10]$).

$d(9) < d(10)$, deci condiția nu este respectată. **Fals.**

D. Există un singur nod cu gradul 1.

Nodul 7 are $d(7) = 1$ (singura muchie: $[14, 7]$). Verificând toate celelalte noduri din componenta principală, toate au grad ≥ 2 , iar nodurile izolate au grad 0. **Adevărat.**

Răspunsul corect este **AD**.



Problema 4

3 Probleme

14. Precizați câte grafuri neorientate distincte cu 5 noduri, numerotate de la 1 la 5, se pot construi, astfel încât nodul 3 să aibă gradul 1. Două grafuri sunt distincte dacă matricele lor de adiacență sunt diferite.

A. 2^8

B. $5!$

C. $5 * C_5^4$

D. 2^5

Sursa: Admitere UBB septembrie 2025



Problema 4 – soluție

3 Probleme

Avem 5 noduri, $\{1, 2, 3, 4, 5\}$ și vrem ca nodul 3 să aibă gradul 1.

Pasul 1: Alegem cu care nod este conectat nodul 3. Posibilități: 1, 2, 4 sau 5 $\Rightarrow$ 4 variante.

Pasul 2: Celelalte 3 muchii posibile cu nodul 3 (i.e., $[3, x]$ pentru x nealese) **nu trebuie** să existe (pentru a menține $d(3) = 1$).

Pasul 3: Muchiile care **nu** implică nodul 3 pot fi orice combinație. Acestea sunt muchiile între nodurile $\{1, 2, 4, 5\}$, adică $C_4^2 = 6$ muchii posibile. Fiecare poate fi prezentă sau nu $\Rightarrow 2^6 = 64$ variante.

Total: $4 \times 2^6 = 4 \times 64 = 256 = 2^8 \Rightarrow$ Răspunsul corect este **A**.



Grafuri și arbori

Mulțumesc pentru atenție!