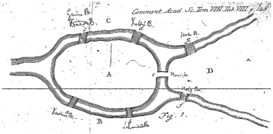
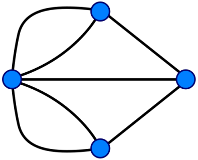




# Algoritmica grafurilor

# I. Introducere

Mihai Suciu
Facultatea de Matematică și Informatică (UBB)
Departamentul de Informatică

Februarie, 28, 2018

# Continut

# De ce?

Obiective:

- Obținerea unei imagini de ansamblu, cunoașterea și ințelegerea noțiunilor, modelelor generale de probleme și algoritmilor de rezolvare a acestora
- Cunoașterea conceptelor teoretice ale algoritmicii grafurilor și aplicarea acestora în modelarea și rezolvarea problemelor
- Analizarea unui gaf și a problemelor ce țin de grafuri: conectivitate, cel mai scurt drum, drum minim, flux de date, problema comis-voiajorului, etc.
- Cunoașterea implementării algoritmilor într-un limbaj de programare

# Conținut curs

1. Noțiuni de bază
2. Studiu aprofundat al reprezentării grafurilor. Drumuri în grafuri.
3. Algoritmul lui Bellman-Kalaba, algoritmul lui Ford, algoritmi matriceali, drum ciclic, drumuri Euleriene, drumuri Hamiltoniene.
4. Conectivitate și probleme de lanț minim. Parcurgeri de graf în lațime și adâncime.
5. Numere fundamentale în teoria grafurilor.
6. Arbori și păduri

# Conținut curs (II)

7. Cuplaje în grafuri
8. Probleme extremale
9. Probleme grele: ciclu Hamiltonian, problema comis-voiajorului. Probleme de numărare și enumerare.
10. Probleme grele: clique, vertex cover, colorare
11. Ciclu elementar Eulerian. Grafuri planare.
12. Rețele de transport.
13. Fluxuri în rețele de transport.
14. Probleme de cuplaj.

# Bibliografie


- Berge C., Graphes et hypergraphes, Dunod, Paris 1970.
- Berge C., Teoria grafurilor și aplicațiile ei, Ed. Tehnica, 1972
- T. Toadere, Grafe. Teorie, algoritmi și aplicații, Ed. Albastra, Cluj-N (ed. I, II, III), 2002 și 2009
- KÁSA ZOLTÁN, Combinatorică cu aplicații, Presa Universitara Clujeana, 2003.
- Cormen, Leiserson, Rivest, Introducere în algoritmi, Editura Computer Libris Agora, 2000.
- Rosu A., Teoria grafelor, algoritmi, aplicații. Ed. Militară, 1974.
- Ciurea E., Ciupala L., Algoritmi - algoritmii fluxurilor în rețele, Ed. Matrix Rom, 2006.
- CATARANCIUC S., IACOB M.E., TOADERE T., Probleme de teoria grafelor, Lito. Univ. Cluj-Napoca, 1994.
- KÁSA Z., TARTIA C., TAMBULEA L.: Culegere de probleme de teoria grafelor, Lito. Univ. Cluj-Napoca 1979.

# Bibliografie (II)


- TOMESCU I., Probleme de combinatorica si teoria grafurilor. Ed. Did. si Pedag. Bucuresti 1981.
- Easley David and Kleinberg Jon. 2010. Networks, Crowds, and Markets: Reasoning about a Highly Connected World. Cambridge University Press, New York, NY, USA.
- Mark Newman. 2010. Networks: An Introduction. Oxford University Press, Inc., New York, NY, USA
- Matthew O. Jackson. 2008. Social and Economic Networks. Princeton University Press, Princeton, NJ, USA.
- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2009. Introduction to Algorithms, Third Edition (3rd ed.). The MIT Press.
- Geir Agnarsson and Raymond Greenlaw. 2006. Graph Theory: Modeling, Applications, and Algorithms. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

# Organizare

- curs, seminar, laborator: Mihai Suciu (*mihai-suciu [at] cs.ubbcluj.ro*) (www.cs.ubbcluj.ro/~mihai-suciu)
- laborator:
  - Asist. drd. COROIU Adriana (http://www.cs.ubbcluj.ro/~adrianac/)
  - C.d.asociat dr. APATEAN Anca
- Pagina web a cursului
  www.cs.ubbcluj.ro/~mihai-suciu/graf/

# Structura

- Curs: 2 ore / săptămână
- Seminar: 1 oră / săptămână
- Laborator: 1 oră / săptămână

Orar:
http://www.cs.ubbcluj.ro/files/orar/2017-2/disc/MLR5025.html

# Evaluare și cerințe

- Colocviu (C) - examen scris
- Activitate Laborator (L) - activitate la laborator (3 teste susținute pe parcursul semestrului)
- Puncte bonus la laborator (B) **1p** (0.25p / laborator)
- nota finala:

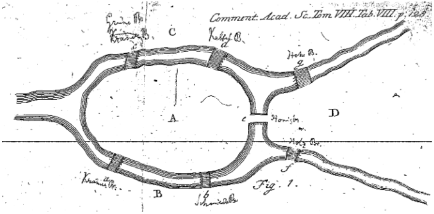
$$\mathbf{0.67} * C + \mathbf{0.33} * L + B = 11p$$

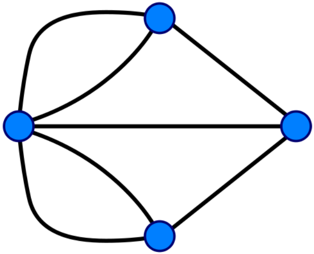
- **colocviu** - nota trebuie să fie **minim 5**!!
- **laborator** - media testelor trebuie sa fie **minim 5**!!

# Evaluare și cerințe (II)

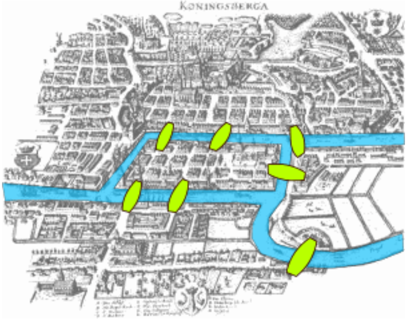
- Activitatea de seminar este OBLIGATORIE în proporție de minim 75% $\longrightarrow$ maxim **2** absențe
- Activitatea de laborator este OBLIGATORIE în proporție de minim 90% $\longrightarrow$ maxim **1** absențe.
- problemele primite la laborator trebuie rezolvate în **C/C++** (ca și IDE se recomandă Qt - `https://www.qt.io/download`)
- Este necesară participarea studenților la ambele ore de seminar / laborator pentru a fi luată în considerare prezența.
- Studenții cu mai mult de 2 absente **nemotivate** la seminar **sau** laborator nu vor fi primiți la examenul din sesiunea normală și **nici** la examenul din sesiunea de restanțe (acești studenți vor trebui să repete acest curs în anul universitar următor). Sunt exceptați de la această cerință cei scutiți medical care pot dovedi cu acte fiecare absență în parte.

# Partea II



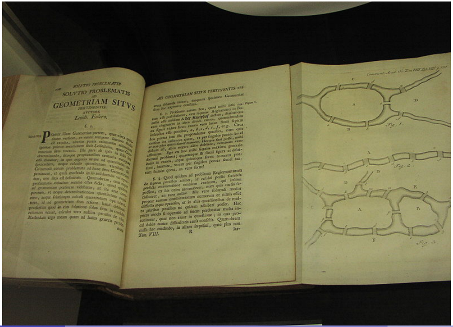



# Inceputuri

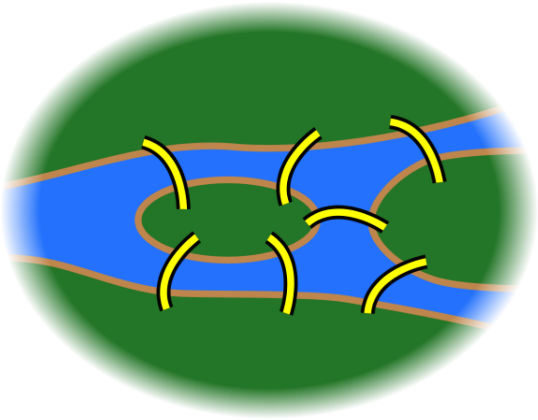


# Soluție

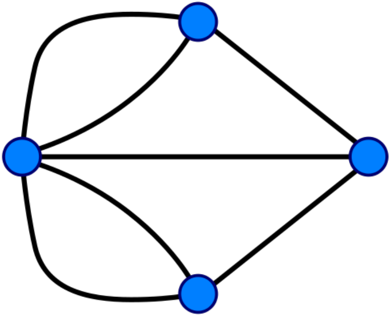
Leonhard Euler (1707-1783)



# Podurile din Königsberg

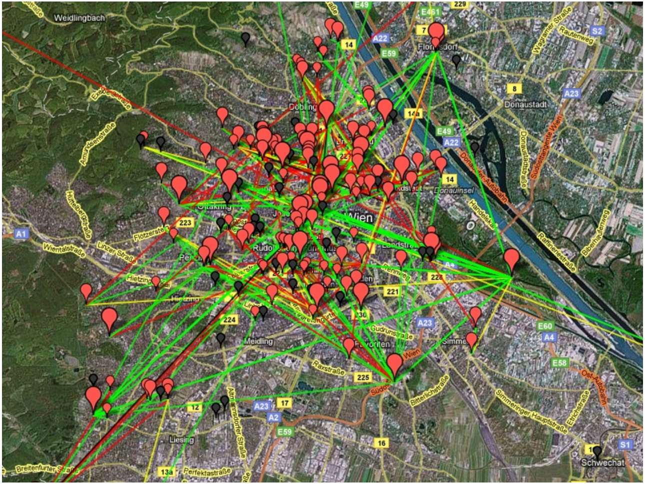


# Podurile din Königsberg (II)

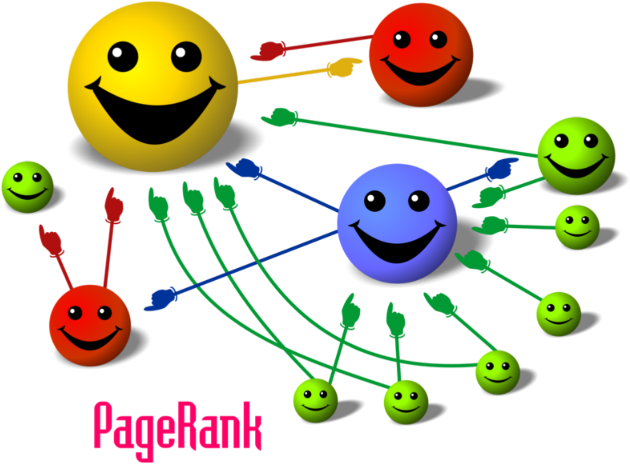


# Unde suntem acum?

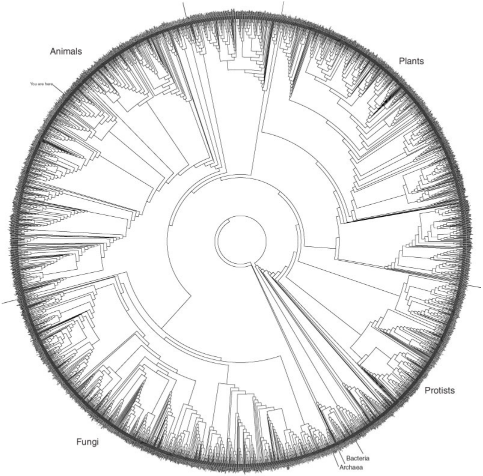
# Unde suntem acum? (II)



# Unde suntem acum? (III)



# Unde suntem acum? (IV)



# Definiții

## Multigraf neorientat

se numește multigraf orice sistem de forma $G = (V, E, g)$ unde
$V$ - mulțimea vârfurilor, $V \neq \varnothing$
$E$ - mulțimea muchiilor, $V \cap E = \varnothing$
$g : E \rightarrow V \bigotimes V$

Se mai poate scrie

$$G = (V(G), E(G), g(G))$$

Observații:

1. dacă mulțimile $V$ și $E$ sunt finite $\Rightarrow$ multigraful G este finit
2. $AxB = \{(a, b)|a \in A, b \in B\}$ - pereche ordonată de elemente
3. $A \bigotimes B = \{(a, b)|a \in A, b \in B \,\mathrm{sau}\, a \in B, b \in A\}$ - pereche neordonată

# Multigraf (n,m)

$n = |V|$ - **ordinul** multigrafului G

$m = |E|$ - **dimensiunea** multigrafului G

- dacă extremitățile unei muchii coincid, muchia se numește **buclă**

$$g(e) = \{a, a\}$$

- dacă

$$g(e_1) = g(e_2)$$

atunci muchiile $e_1$ și $e_2$ sunt paralele

# Muchii adiacente

- setul muchiilor ce leagă vârfurile $a$ și $b$ este

$$g^{-1}(a, b) = \{e \in E(G) | g(e) = \{a, b\}\}$$

**Muchii adiacente**

fie $x$ un vârf din $G$. $N_G(x)$ sau $N(X)$ este setul muchiilor adiacente lui x:

$$N_G(x) = \{y \in V(G), \exists e \in E(G), g(e) = \{x, y\}\}$$

sau

$$N_G(X) = \{y \in V(G), g^{-1}(x, y) \neq \varnothing\}$$

# Muchii incidente

## Muchii incidente

intr-un multigraf G, setul muchiilor incidente lui $x$ (care nu sunt bucle) este:

$$I_G(x) = \{e \in E(G), \exists y \in V(G), y \neq x, g(e) = \{x, y\}\}$$

## Bucle incidente

setul buclelor incidente nodului $x$ este:

$$L_G(x) = \{e \in E(G), g(e) = \{x, x\}\}$$

# Gradul unui vârf

## Gradul unui vârf

gradul unui vârf, notat $d(x)$, este numărul muchiilor incidente lui $x$:

$$d(x) = card(I_G(x)) + 2 * card(L_G(x)).$$

Dacă:

- $d(x) = 0$, $x$ este un vârf izolat
- $d(x) = 1$, $x$ este vârful unei muchii

# Graf simplu

## Graf simplu

un multigraf fără bucle și muchii paralele se numește **graf simplu**. În acest caz

$$|g^{-1}(a, b)| \leq 1, \forall a, b \in V.$$

Se poate scrie $\{a, b\}$ în loc de $g(e) = \{a, b\}$.
Graful se notează $G = (V, E)$.

Observații:

- pentru un graf simplu gradul unui nod este:

$$d(x) = |N_G(x)|$$

# Definiții

## graf regular

graf în care toate vârfurile au aceleași grad.
Graf $k$-regular, toate vârfurile au gradul $k$

$$d(x) = k, \forall x \in V(G).$$

## Graf complet

un graf pentru care toate perechile de vârfuri sunt adiacente se numește **graf complet**. Un graf complet de ordinul $n$ se notează $K_n$.

Exemple

# Definiții (II)

graful $G' = (V', E')$ este complementul grafului $G = (V, E)$, dacă $V' = V$ și $E' = \{\{a, b\}, \{a, b\} \notin E\}$.

- dacă G este un graf de ordinul $n$ atunci $E(G) \cup E(G') = E(K_n)$.

# Izomorfism

## Izomorfism multigraf

multigrafurile $G_1$ și $G_2$ sunt izomorfe dacă există bijecțiile $f : V_1 \rightarrow V_2$ și $g : \mathbb{N} \rightarrow \mathbb{N}$ astfel ca:

$$(i, j, k) \in E_1 \Leftrightarrow (f(i), f(j), g(k)) \in E_2.$$

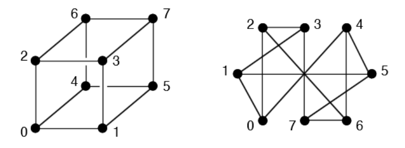
## Izomorfism graf simplu

două grafuri $G_1$ și $G_2$ sunt izomorfe dacă există funcția bijectivă $f : V_1 \rightarrow V_2$ astfel ca:

$$(a, b) \in E_1 \Leftrightarrow (f(a), f(b)) \in E_2.$$

# Izomorfism (II)

## Exemplu izomorfism




## Teoremă

relația de izomorfism în mulțimea grafurilor este o relație de echivalență.

# Exemple de grafuri

- graf nul
- graf linie
- graf ciclu
- graf complet
- graf bipartit complet

### hand-shaking

pentru un graf G avem

$$\sum_{u \in V(G)} d_G(u) = 2|E(G)|.$$

### corolar

într-un graf există întotdeauna un număr par de vârfuri ce au grad impar

### corolar

fiecare graf $k$-regular pe $n$ vârfuri are $\frac{kn}{2}$ muchii, în particular $K_n$ are $\frac{n(n-1)}{2}$ muchii.

# Subgraf

## Definiție

pentru multigraful $G_1 = (V_1, E_1, g_1)$ și $G = (V, E, g)$ spunem că $G_1$ este subgraf al lui $G$ dacă
$V_1 \subseteq V$
$E_1 \subseteq E$
$g_1(e) = g(e) \forall e \in E_1$
se notează $G_1 \subseteq G$

# Multigraf, graf orientat

## Multigraf orientat

se numește multigraf orientat orice sistem de forma $\vec{G} = (V, E, \eta)$ unde
$V$ este mulțimea vârfurilor, $V \neq \varnothing$
$E$ este mulțimea **arcelor**, $V \cap E = \varnothing$
$\eta : E \to VxV$

- $\eta(e) = \{u, v\}$ arcul este **incident spre exterior** vârfului u, arcul este **incident spre interior** vârfului v
- $\eta(e) = \{u, u\}$ - arc buclă
- $\eta(e_1) = \eta(e_2)$ - arce paralele
- un **graf orientat simplu** se definește în mod similar unui graf neorientat

# Subgrad interior, exterior

## Definiție

fie multigraful $G = (V, E, \eta)$ și vârful $x \in V$

- se numește subgradul interior, se notează $d^-(x)$, numărul arcelor incidente spre interior vârfului $x$:

$$d^-(x) = |\{e \in E | \eta(e) = \{y, x\}, \forall y \in V\}| = |N^{in}_{\vec{G}}(x)|$$

- se numește subgradul exterior, notat $d^+$, numărul arcelor incidente spre exterir nodului $x$:

$$d^+(x) = |\{e \in E | \eta(e) = \{x, y\}, \forall y \in V\}| = |N^{out}_{\vec{G}}(x)|$$

- $$d(x) = d^-(x) + d^+(x)$$

# Subgrad interior, exterior (II)

**Teoremă**

pentru un multigraf orientat avem:

$$\sum_{x \in V(\vec{G})} d^{-}(x) = \sum_{x \in V(\vec{G})} d^{+}(x) = |E(\vec{G})|$$

# Multigraf ponderat

## Multigraf ponderat

se numește multigraf ponderat orice sistem de forma $G = (V, E, g, W)$
$V$ - mulțimea vârfurilor, $V \neq \varnothing$
$E$ - mulțimea muchiilor, $V \cap E = \varnothing$
$g : E \rightarrow V \bigotimes V$
$W : E \rightarrow \mathbb{R}$ - ponderea muchiilor

## Multigraf ponderat orientat

se numește multigraf ponderat orientat orice sistem de forma
$\vec{G} = (V, E, \eta, W)$ unde
$V$ este mulțimea vârfurilor, $V \neq \varnothing$
$E$ este mulțimea **arcelor**, $V \cap E = \varnothing$
$\eta : E \rightarrow V \ x \ V$
$W : E \rightarrow \mathbb{R}$ - ponderea arcelor

# Drumuri în grafuri

## Drum

fiind dat un graf orientat $G = (V, E)$, prin **drum** în graful $G$ înțelegem o succesiune de arce cu proprietatea că extremitatea terminală a unui arc al drumului coincide cu extremitatea inițială a arcului urmator din drum.

- drum = o succesiune de vârfuri care sunt extremități ale arcelor ce compun drumul
- un drum $\mu$ este $\{e_1, e_2, ..., e_q\}$ fie $\{i_0, i_1, ..., i_q\}$ cu proprietatea că $e_j = (i_{j-1}, i_j) \in E$ pentru $j = 1, 2, ..., q$

# Drumuri în grafuri (II)

**Lungimea unui drum**

lungimea unui drum este numărul arcelor care compun drumul respectiv.

Un drum într-un graf este:

- **simplu** dacă nu folosește de două ori un același arc
- **compus** dacă nu este simplu
- **elementar** dacă nu contine (trece) de două ori un același vârf
- **circuit** dacă extremitatea inițială a drumului coincide cu cea finală
- **eulerian** dacă este simplu și trece prin toate arcele grafului
- **hamiltonian** dacă este elementar și trece prin toate vârfurile grafului

# Drumuri în grafuri neorientate

- corespunzător noțiunilor de drum si circuit în grafurile neorientate sunt noțiunile de **lanț** și **ciclu**

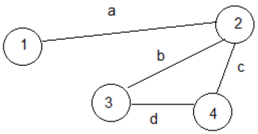
## Lanț

un **lanț** este o succesiune de muchii cu proprietatea că oricare muchie are o extremitate comună cu muchia precedentă și cealaltă extremitate este comună cu muchia următoare.

## Ciclu

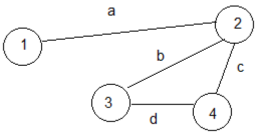
dacă extremitățile lanțului coincid, atunci lanțul se numește **ciclu**.

# Exemplu



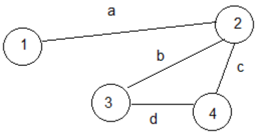

- Lanț:

# Exemplu



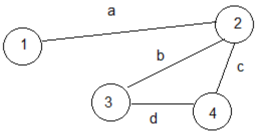

- Lanț: 1a2c4d3b2c4

# Exemplu



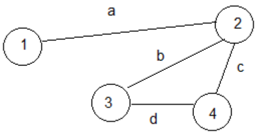

- Lanț: 1a2c4d3b2c4
- Lanț simplu:

# Exemplu



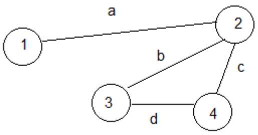

- Lanț: 1a2c4d3b2c4
- Lanț simplu: 2b3d4c2

# Exemplu



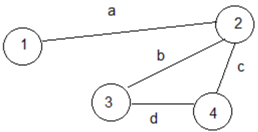

- Lanț: 1a2c4d3b2c4
- Lanț simplu: 2b3d4c2
- Lanț elementar:

# Exemplu



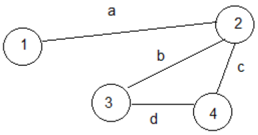

- Lanț: 1a2c4d3b2c4
- Lanț simplu: 2b3d4c2
- Lanț elementar: 1a2b3

# Exemplu




- Lanț: 1a2c4d3b2c4
- Lanț simplu: 2b3d4c2
- Lanț elementar: 1a2b3
- Ciclu simplu:

# Exemplu




- Lanț: 1a2c4d3b2c4
- Lanț simplu: 2b3d4c2
- Lanț elementar: 1a2b3
- Ciclu simplu: 2b3d4c2

# Graf tare conex, conex

### Graf tare conex

un graf orientat este **tare conex** dacă între oricare două vârfuri ale grafului există un drum.

- graf tare conex - prin oricare două vârfuri trece ce puțin un circuit

### Graf conex

un graf neorientat este **conex** dacă între oricare două vârfuri ale grafului există un lanț.

# Reprezentarea matriceală

Pentru exemplele date grafurile au fost reprezentate grafic, pentru un program scris această reprezentare nu este suficientă.

Un graf poate fi reprezentat folosind:

- **matricea de adiacență**

  $A = (a_{ij}), i, j = 1, 2, ..., n$ unde

$$a_{ij} = \begin{cases} 1, & (i,j) \in E \\ 0, & (i,j) \notin E \end{cases}$$

# Reprezentarea matriceală

Pentru exemplele date grafurile au fost reprezentate grafic, pentru un program scris această reprezentare nu este suficientă.

Un graf poate fi reprezentat folosind:

- **matricea de adiacență**
  $A = (a_{ij}), i, j = 1, 2, ..., n$ unde

$$a_{ij} = \begin{cases} 1, & (i,j) \in E \\ 0, & (i,j) \notin E \end{cases}$$

- **matricea de incidență** - se atașează grafurilor simple a căror mulțime de arce s-a ordonat, linia $i$ corespunde vârfului $i$ iar coloana $j$ corespunde arcului $e$, matricea este de tipul $nxm$. Elementele matricii:

$$b_{ij} = \begin{cases} 1, & \exists j \in V | e = (i,j), \\ -1, & \exists j \in V | e = (j,i), i \in V, e \in E \\ 0, & altfel \end{cases}$$

# Reprezentarea matriceală

Pentru exemplele date grafurile au fost reprezentate grafic, pentru un program scris această reprezentare nu este suficientă.

Un graf poate fi reprezentat folosind:

- **matricea de adiacență**
  $A = (a_{ij}), i, j = 1, 2, ..., n$ unde
  $$a_{ij} = \begin{cases} 1, & (i,j) \in E \\ 0, & (i,j) \notin E \end{cases}$$
- **matricea de incidență** - se atașează grafurilor simple a căror mulțime de arce s-a ordonat, linia $i$ corespunde vârfului $i$ iar coloana $j$ corespunde arcului $e$, matricea este de tipul $nxm$. Elementele matricii:
  $$b_{ij} = \begin{cases} 1, & \exists j \in V | e = (i,j), \\ -1, & \exists j \in V | e = (j,i), i \in V, e \in E \\ 0, & altfel \end{cases}$$
- **listă**

# Determinarea matricei distanțelor

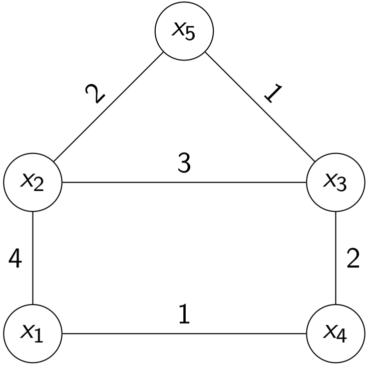
algoritmul Warshall

$\textsc{Warshall}(D_0)$

$D := D_0$
**for** $k := 1$ **to** $n$ **do**
  **for** $i := 1$ **to** $n$ **do**
    **for** $j := 1$ **to** n **do**
      $d_{ij} := \min(d_{ij}, d_{ik} + d_{kj})$
**return** $D$

# Exemplu

Fie graful ponderat



# Matricea distanțelor

$$D_0 = \begin{pmatrix} 0 & 2 & \infty & \infty & 1 \\ 2 & 0 & 4 & \infty & 3 \\ \infty & 4 & 0 & 1 & \infty \\ \infty & \infty & 1 & 0 & 2 \\ 1 & 3 & \infty & 2 & 0 \end{pmatrix} \qquad D = \begin{pmatrix} 0 & 2 & 4 & 3 & 1 \\ 2 & 0 & 4 & 5 & 3 \\ 4 & 4 & 0 & 1 & 3 \\ 3 & 5 & 1 & 0 & 2 \\ 1 & 3 & 3 & 2 & 0 \end{pmatrix}$$

# Problema puncte bonus

- primele cinci rezolvari corecte (CU EXPLICAȚII) primesc puncte bonus
- rezolvarile trebuie trimise la adresa *mihai-suciu@cs.ubbcluj.ro*

Problema: Într-o zi Hercule Poirot primește vizita bunului său prieten căpitanul Hastings care a fost rugat să investigheze un furt care a rămas nerezolvat de zece ani.
Acum zece ani contesei Vera Rossakoff i-a fost furat faimosul diamant ”diamantul Cullinan” din propriul castel. Contesa a dat o petrecere la castel cu câteva zile înainte de furt.
Căpitanul Hastings a intervievat persoanele care au participat la petrecere (Abe, Oliver, Li Chang, Darrell, Betsy, colonelul Appleby și Sonia), dar după zece ani nu și-au adus aminte foarte multe detalii.
Detaliile cazului, locația diamantului arată că făptașul cunoștea castelul. Hastings a întrebat suspecții de câte ori au fost la castel dar după zece ani nu și-au adus aminte.

Suspecți:

- Abe, Oliver, Li Chang, Darrell, Betsy, colonelul Appleby și Sonia

Hastings i-a mai întrebat cu cine s-au întâlnit la castel:

- Abe: Oliver, Li Chang, Betsy și colonelul Appleby.
- Oliver: Abe, Li Chang, Darrell, Betsy și Sonia.
- Li Chang; Abe, Oliver și Darrell.
- Darrell: Oliver, Li Chang și Betsy.
- Betsy: Abe, Oliver, Darrell și Sonia.
- colonelul Appleby: Abe și Sonia.
- Sonia: Oliver, Betsy și colonelul Appleby.

Poirot a desenat ceva pe hârtie și după câteva minute a știut cine a furat diamantul. Cine este hoțul?