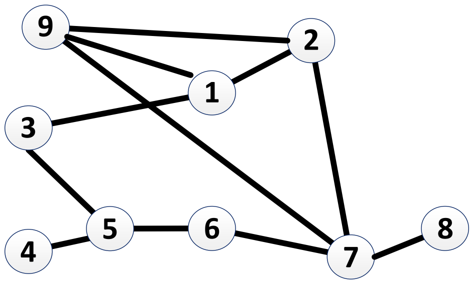
1. **(1p)** Desenați o hartă a următorului graf și indicați fețele și gradul fiecărei fețe.



2. (**1p**) Care dintre următoarele afirmații sunt adevărate? Justificați pentru fiecare afirmație de ce este adevărată sau falsă (complexitatea algoritmilor studiați se presupune cunoscută, nu trebuie demonstrată în justificare)

a) Orice graf neorientat fără puncte critice (de articulație) nu are muchii critice.

b) Fie G un graf orientat ponderat fără circuite (ponderile pot fi și negative) și un vârf t. Există un algoritm de complexitate O(n+m) care calculează distanța de la fiecare vârf al grafului la vârful t, unde n este numărul de vârfuri ale grafului și m numărul de arce.

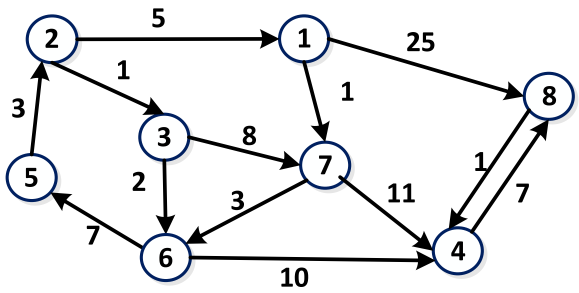
c) Fie M_1 si M_2 două hărți ale unui graf planar conex G. Hărțile M_1 și M_2 au același număr de fețe.

d) Fie G un graf neorientat conex, ponderat. Dacă în G sunt și muchii de același cost, atunci G are cel puțin doi arbori parțiali de cost minim.

3. (1p) a) Se consideră următoarea mulțime de intervale închise [6,12], [3,5], [2,7], [2,9], [8,10], [11,14]: definiți și desenați graful interval (graful intersecției intervalelor) asociat mulțimii de intervale dată.

b) Descrieți algoritmul Greedy de colorare a vârfurilor unui graf neorientat pentru o ordonare dată a vârfurilor și **exemplificați** (pas cu pas) acest algoritm pentru graful de la punctul a) alegând o ordonare a vârfurilor care să **asigure** faptul că algoritmul Greedy va folosi un număr minim de culori. Care este strategia de ordonare care asigură colorarea cu un număr minim de culori folosind algoritmul Greedy pentru un graf interval (în cazul general)?

4. (1,5p) a) Se consideră următorul graf orientat ponderat și vârful sursă s = 2. Descrieți algoritmul lui Dijkstra și ilustrați (cu explicații) pașii algoritmului lui Dijkstra până când este vizitat (selectat) vârful 4 și sunt relaxate arcele care ies din acesta (inclusiv acest pas).



b) Dați exemplu de un graf orientat ponderat cu 7 vârfuri și cel puțin 8 arce pentru care algoritmul lui Dijkstra pentru sursa s = 1 nu calculează corect distanța de la vârful 1 la vârful 4. Justificați.

5. (2p) a) Definiți noțiunile de flux, tăietură (s-t tăietură), capacitate a unei tăieturi și tăietură minimă.

b) Se consideră o rețea de transport N = (G=(V, E), s, t, c) cu n vârfuri și m arce (vârful sursă *s*, destinația *t* și funcția de capacitate *c*) și *f* **un flux maxim în** N. Se presupun construite în memorie matricele C și F cu semnificația:

- C este matricea de capacități: pentru i, j ∈{1,…,n}

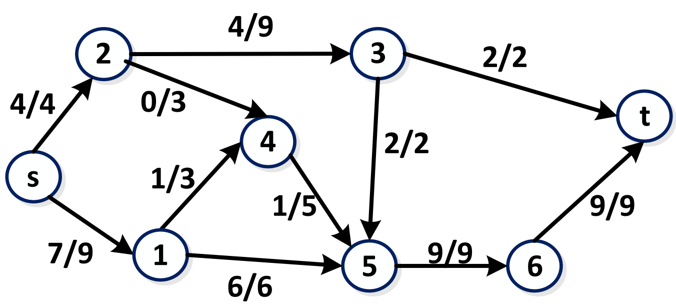
$$C[i][j]=\begin{cases} c(ij), & \text{daca ij} \in \text{E} \\ 0, & \text{altfel} \end{cases}$$

- F este o matrice în care este memorat fluxul maxim f:

$$F[i][j]=\begin{cases} f(ij), & \text{daca ij} \in \text{E} \\ 0, & \text{altfel} \end{cases}$$

Scrieți pseudocodul unui algoritm O(n+m) care, având în memorie n, m, listele de adiacență ale lui G și matricele C și F, determină o tăietură (s-t tăietură) minimă în N. Justificați complexitatea algoritmului propus. (Nu este necesară citirea grafului, se pot folosi în pseudocod instrucțiuni de genul for j in lista de adiacenta a lui i, for ij in E etc)

c) Ilustrați pașii algoritmului descris la b) pentru rețeaua din figura următoare, unde pe un arc e sunt trecute valorile f(e)/c(e) reprezentând flux/capacitate; fluxul f este flux maxim:



d) Presupunem că toate capacitățile arcelor rețelei N sunt pare. Arătați că nu există un flux maxim f cu proprietatea că f(ij) este impar pentru orice ij∈E.

6. (1,5p) Fie G un graf orientat ponderat cu n vârfuri și m arce și p un număr natural mai mic sau egal cu n. Să se determine acele perechi de vârfuri u, v cu proprietatea că există un drum de cost minim de la u la v (de cost egal cu distanța d(u, v)) care are ca vârfuri interne (diferite de extremitățile u și v) doar vârfuri din mulțimea {1,2,…, p}; pentru fiecare astfel de pereche să se afișeze și un drum cu proprietatea cerută.

Descrieți un algoritm de complexitate $O(n^3)$ care rezolvă această problemă și scrieți pseudocodul acestui algoritm (fără citirea datelor; se pot folosi în pseudocod instrucțiuni de genul: fie C matricea costurilor lui G, for j in lista de adiacenta a lui i, for ij in E etc). Justificați corectitudinea și complexitatea algoritmului descris (justificare completă a corectitudinii, fără a presupune cunoscută corectitudinea unor algoritmi clasici sau a unor rezultate demonstrate la curs).

7. **(1p)**. a) Dați exemplu de un graf neorientat conex cu 7 vârfuri care este eulerian, dar nu este hamiltonian (justificați).

b) Dați exemplu de un graf neorientat conex cu 7 vârfuri care este hamiltonian, dar nu este eulerian.