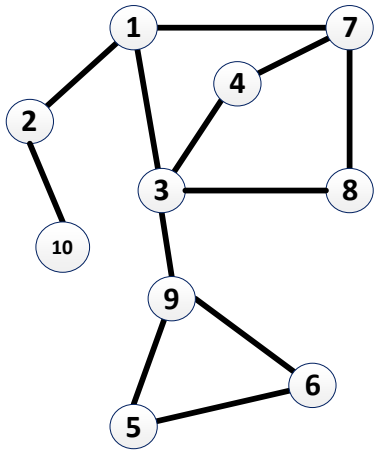
Nume: **25.01.2023**
Grupa:

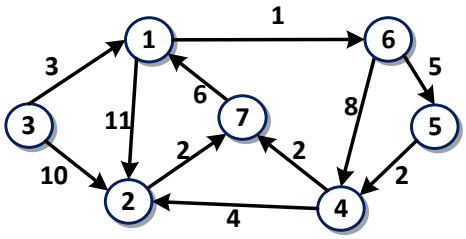
EXAMEN LA ALGORITMI FUNDAMENTALI

Pentru graful din imaginea din stânga rezolvaţi cerinţele 1-4 şi justificaţi răspunsurile; <u>vecinii unui vârf se consideră în ordine lexicografică</u>



1) (0,5p) Care sunt muchiile critice (definiţi şi noţiunea)?
2) (0,5p) Exemplificaţi (cu explicaţii) cum funcţionează parcurgerea în adâncime df(4), ilustrând si arborele df asociat
3) (0,75p) Indicaţi un subgraf bipartit (indus) al acestui graf cu număr maxim de noduri şi indicaţi o bipartiţie (sau bicolorare) a acestuia, explicând şi cum aţi construit-o.
4) (0,75p) Puneţi ponderi pe muchii astfel încât graful să aibă exact 2 arbori parţiali de cost minim de cost 50 şi indicaţi aceşti arbori.

Pentru graful din imaginea din stânga rezolvaţi cerinţele 5 şi 6:

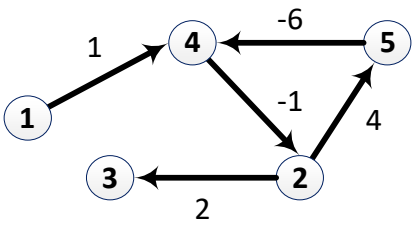


5) (0,5p) Exemplificaţi paşii algoritmului lui Dijkstra (cu explicaţii) pornind din vârful 3.

6) (0,5p) Exemplificaţi paşii algoritmului lui Kruskal (cu explicaţii) pentru graful neorientat asociat (obţinut ignorând orientarea arcelor)

7) (0,5p) Fie G = (V, E, w) un graf orientat ponderat, cu ponderi numere **întregi** şi s un vârf în G. Considerăm algoritmul lui Bellman Ford descris în următorul pseudocod:

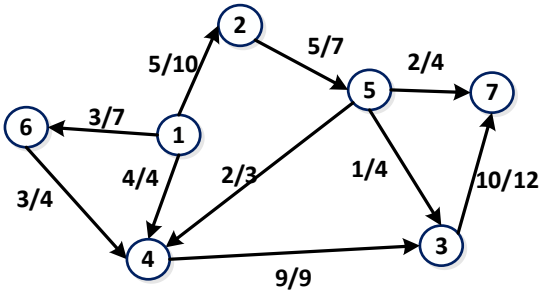
```
pentru fiecare u∈V executa
      d[u] = infinit
d[s] = 0
pentru i = 1, |V|-1 executa
      pentru fiecare uv ∈ E executa
            daca d[u]+w(u,v)<d[v]
      atunci
                  d[v] = d[u]+w(u,v)
```



Considerăm graful din figura din dreapta pseudocodului. La finalul execuţiei pseudocodului de mai sus pentru acest graf, s=1 şi arcele considerate în ordinea E={(1,4), (2,3), (4,2), (2,5), (5,4)} vectorul d are elementele [0, -9, -4, -11, -5]. Adăugaţi în pseudocod instrucţiunile necesare (cu explicaţii) pentru ca algoritmul să testeze existenţa unui circuit cu cost negativ în graf accesibil din s (=pentru care există un drum de la s la un vârf al său) şi ilustraţi-le pe graful dat ca exemplu (cu explicaţii).

CERINŢĂ - Minim 3p din primele 7 subiecte

8) (1p) În rețeaua de transport din figura alăturată pe un arc e sunt trecute valorile f(e)/c(e) reprezentând flux/capacitate. Sursa este vârful 1, iar destinația 7.

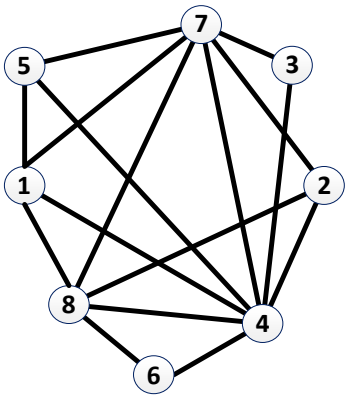


Ilustrați pașii algoritmului Ford-Fulkerson pentru această rețea pornind de la fluxul indicat și alegând la fiecare pas un s-t lanț f-nesaturat de lungime minimă (algoritmul Edmonds-Karp). Indicați o tăietură (s-t tăietură) minimă în rețea (se vor indica vârfurile din bipartiție, arcele directe, arcele inverse) și determinați capacitatea acestei tăieturi. Justificați răspunsurile.

9) (1,5p) Fie G un graf neorientat conex cu exact k noduri (vârfuri) de grad impar. Arătați că există [k/2] lanțuri simple (în care nu se repetă muchii) care nu au muchii în comun cu proprietatea că orice muchie din graf aparține unui astfel de lanț și [k/2] este minim cu această proprietate (nu există un număr mai mic de lanțuri simple fără muchii în comun cu proprietatea că orice muchie din graf aparține unui astfel de lanț).

b) Fie n>5. Care este numărul maxim de noduri (puncte) critice ale unui graf neorientat conex cu n noduri și exact k=4 noduri de grad impar? Justificați și construiți un astfel de graf cu număr maxim de noduri critice (neorientat, conex, cu n noduri dintre care 4 de grad impar)

10) (0,5p) Explicați cum se modifică algoritmul (relațiile de recurență) pentru calculul distanței de editare între două cuvinte dacă operația de inserare a unui caracter are costul c1 și operația de modificare a unui caracter în altul are costul c2, cu c1 și c2 numere naturale date (deci nu toate operațiile au costul 1, ca în cazul distanței de editare Levenshtein clasică)



11) (0,5p) Descrieți algoritmul de 6-colorare a vârfurilor unui graf neorientat conex planar și exemplificați acest algoritm pentru graful alăturat.

12) (1,5p)

(0,75 soluție corectă + 0,75 discuții complexitate + complexitate optimă)