Subiectul 1 (3 puncte)

Se dă un arbore ponderat T cu n>3 vârfuri și un vârf s.
Informațiile despre arbore se citesc din fișierul ***graf.in*** cu structura:

- pe prima linie este n
- pe următoarele n linii sunt listele de adiacență ale lui G; linia i începe cu gradul vârfului i și apoi conține vecinii vârfului i, fiecare vecin fiind urmat de costul muchiei:

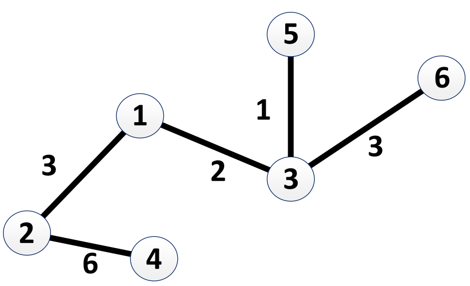
<grad> <vecin1> <cost_muchie_de_la_i_la_vecin1> <vecin2> <cost_muchie_de_la_i_la_vecin2> etc

- pe ultima linie este vârful s

(vârfurile sunt numerotate de la 1)
Pentru un lanț P în T definim capacitatea lanțului P ca fiind capacitatea minimă a unei muchii din P.

a) Să se afișeze pentru fiecare vârf v capacitatea unicului lanț elementar de la s la v (sub forma v: capacitate lanț) **Complexitate O(n)**
b) Să se afișeze care este capacitatea minimă a unui lanț cu o extremitate în s și să se afișeze un astfel de lanț

graf.in	***Iesire pe ecran***
6 **2** 2 3 3 2 **2** 1 3 4 6 **3** 1 2 5 1 6 3 **1** 2 6 **1** 3 1 **1** 3 3 1 Explicații: informațiile de pe linia 2 2 3 3 2 corespunzătoare vârfului 1 se citesc astfel: varful 1 are gradul 2; primul său vecin este 2 și costul muchiei de la el la 2 este 3: al doilea sau vecin este 3, iar costul muchiei de la el la 3 este 2	1: 0 2: 3 3: 2 4: 3 5: 1 6: 2 Capacitatea minima este 1 pentru lantul 1 3 5



Subiectul 2 (3 puncte)

Se citesc informații despre un graf **orientat** ponderat **fără circuite** G din fișierul `graf.in`. Fișierul are următoarea structură:

- pe prima linie sunt două numere reprezentând numărul de vârfuri n (n>4) și numărul de arce m ale grafului, **m>n**
- pe următoarele m linii sunt câte 3 numere întregi reprezentând extremitatea inițială, extremitatea finală și costul unui arc din graf (**costul poate fi și negativ**)
- pe următoarea linie (a (m+2)-a linie) din fișier este un număr natural k (0<k<n) reprezentând numărul de vârfuri sursă; vârfurile sursă din G vor fi 1, 2, ... , k
- pe ultima linie a fișierului sunt două vârfuri t_1 și t_2, reprezentând vârfurile destinație ale grafului (distincte de vârfurile sursă din G).

Notăm cu S ={1,…,k} mulțimea vârfurilor sursă din G și cu T={t_1,t_2} mulțimea vârfurilor destinație din G. Spunem că un vârf y este accesibil din x în G dacă există un drum de la x la y. Presupunem că există cel puțin un vârf destinație care este accesibil dintr-un vârf sursă.

a) Să se verifice dacă graful dat este fără circuite și să se afişeze un mesaj corespunzător.

b) Să se determine distanța între cele două mulțimi S și T:

$$d(S, T) = \min \{d(x, y) \mid x \in S, y \in T\}$$

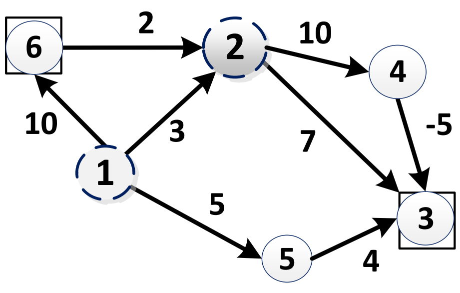
Să se determine în plus și o pereche de vârfuri (s,t) cu $s \in S$ și $t \in T$ cu

$$d(s,t) = d(S,T) = \min \{d(x, y) \mid x \in S, y \in T\}$$

și să se afişeze (pe ecran) un drum minim de la s la t. **Complexitate O(m)**

Exemplu

graf.in	**Iesire pe ecran**
`6 8`	`distanta intre multimi = 5`
`1 2 3`	`s=2 t=3`
`1 6 10`	`drum minim 2 4 3`
`6 2 2`	
`2 4 10`	
`4 3 -5`	
`5 3 4`	
`1 5 5`	
`2 3 7`	
`2`	
`3 6`	



Explicații

k=2 ⇒ S = {1, 2}

T = {3, 6}

d(1,3)=8, d(2,3)=5

d(1,6)=10, d(2,6)=∞

Cea mai mică este d(2,3)=5

Un drum minim de la 2 la 3 este 2 4 3

Subiectul 3 (3 puncte)

Propuneți un algoritm bazat pe algoritmul Ford-Fulkerson / Edmonds Karp pentru rezolvarea următoarei probleme.

Într-un restaurant sunt n mese numerotate 1,...,n sunt și m ospătari numerotați 1,..., m. Proprietarul restaurantului urmează să aibă un eveniment în restaurant și dorește să repartizeze fiecărui ospătar mesele de care trebuie să se ocupe. El întreabă pe fiecare ospătar la ce mese ar vrea să servească, altfel că pentru fiecare ospătar j știe lista meselor pe care le preferă.

Proprietarul ar vrea ca la fiecare masă să fie exact k ospătari și un ospătar să servească la cel mult p mese și ar vrea să respecte și preferințele ospătarilor legate de mese.

Scrieți un program care, dacă este posibilă o distribuție a ospătarilor la mese care să respecte dorințele proprietarului, cu **cel mult** o excepție și anume o masă la care să fie doar k-1 ospătari, să afișeze o astfel de distribuție sub forma prezentată în exemplul de mai jos (perechi masa ospătar). Altfel se va afișa mesajul “imposibil”

Datele despre restaurant și opțiunile ospătarilor se vor citi dintr-un fișier cu următoarea structură:

- pe prima linie sunt numerele naturale n, m, k, p
- pe următoarele m linii sunt numere naturale, pe linia a j-a din cele m fiind numărul de mese la care vrea să servească ospătarul j și care sunt indicii acestor mese

Complexitate $O(n^2m^2)$

restaurant.in	Iesire pe ecran (solutia nu este unica)
4 7 2 1 **4** 1 2 3 4 **1** 1 **2** 2 4 **2** 2 3 **2** 1 4 **3** 1 3 4 **1** 1 Explicații: Ospătarul 1 prefera 4 mese: 1,2,3,4, ospătarul 2 preferă masa 1, ospătarul 3 preferă 2 mese: 2 și 4 etc	1 2 1 7 2 3 2 4 3 1 3 6 4 5 (primul indice reprezintă masa, iar al doilea ospătarul asociat)