Subiectul 1

Se dă un graf neorientat conex G cu n>3 vârfuri, m muchii, m>n. Să se determine doi arbori parțiali T și T' ai lui G cu proprietățile:

- T este arbore de distante față de vârful 1: $d_T(1,v) = d_G(1,v)$ pentru orice vârf v din G
- În T' există cel puțin un vârf v cu $d_{T'}(1, v) \neq d_G(1, v)$.

Se vor afișa muchiile celor doi arbori parțiali determinați și, în plus, se vor afișa toate vârfurile v pentru care $d_{T'}(1,v) \neq d_G(1,v)$. **Complexitate O(m)**
Informațiile despre graf se citesc din fișierul graf.in cu structura:

- pe prima linie sunt n și m
- pe următoarele m linii sunt câte 2 numere naturale reprezentând extremitățile unei muchii

($d_G(x,y)$ = distanța de la x la y în G)

graf.in	Iesire pe ecran (solutia nu este unica)
5 7 1 2 1 3 2 3 2 4 3 4 3 5 4 5	T: 1 2 1 3 2 4 3 5 T': 1 2 2 4 4 5 3 4 v: 3 5

Subiectul 2

Se citesc informații despre un graf **orientat** ponderat 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 **pozitive** reprezentând extremitatea inițială, extremitatea finală și costul unui arc din graf
- 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.

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.

Să se determine distanța între cele două mulțimi:

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

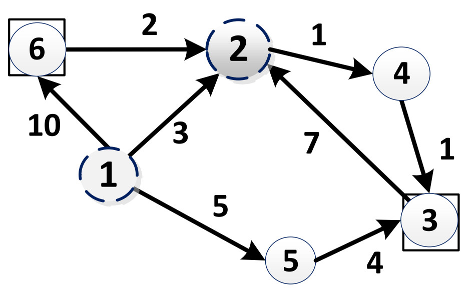
Să se determine în plus și o pereche de vârfuri (s,t) cu $s \in S$ și $t \in T$ cu

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

și să se afișeze (pe ecran) un drum minim de la s la t. **Complexitate O(mlog(n))**

Exemplu

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



Explicații

k=2 ⇒ S = {1, 2}

T = {3, 6}

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

d(1,6)=10, d(2,6)=∞

Cea mai mică este d(2,3)

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

Subiectul 3

Fişierul graf.in conţine următoarele informaţii despre un graf **bipartit** conex:

- pe prima linie sunt 2 numere naturale n şi m reprezentând numărul de vârfuri şi numărul de muchii
- pe următoarele m linii sunt perechi de numere x y (separate prin spaţiu) reprezentând extremităţile unei muchii

Se consideră graful G dat în fişierul graf.in. Notăm cu k numărul de vârfuri de grad impar din graf.

a) Folosind un algoritm de determinare a unui flux maxim într-o reţea de transport, determinaţi un cuplaj maxim în subgraful indus de mulţimea vârfurilor de grad impar din G.

b) Folosind punctul a) determinaţi dacă exista k/2 muchii care se pot elimina din G astfel încât să se obţină un graf cu următoarele proprietăţi:

- gradul fiecărui vârf din G' este egal cu cel din G sau cu unu mai mic.
- în G' în fiecare componentă conexă există câte un ciclu care conţine toate muchiile din componentă (o singura dată) **Complexitate O(nm^2)**

graf.in	Iesire pe ecran (solutia nu este unica)
8 9 1 5 1 6 1 7 2 5 3 5 3 7 3 4 8 7 8 4	1 6 2 5 3 7