

Subiectul 1

Se dă un graf neorientat conex cu $n > 3$ vârfuri, m muchii, $m > n$ și un vârf s .

Să se afișeze muchiile a doi arbori parțiali ai grafului, T_1 și T_2 , dintre care unul, T_1 , este arbore de distanțe față de s ($d_{T_1}(s, u) = d_G(s, u)$ pentru orice vârf u din G), iar celălalt, T_2 , nu este arbore de distanțe față de s . Se va afișa în plus un vârf u pentru care $d_{T_2}(s, u) \neq d_G(s, u)$.

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
- pe ultima linie este vârfurile s

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

<i>graf.in</i>	<i>iesire pe ecran (soluția nu este unică)</i>
4 5	T1:
1 2	1 2
1 3	1 3
2 3	2 4
2 4	T2:
3 4	1 2
1	2 3
	2 4
	$u = 3$

Subiectul 2

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

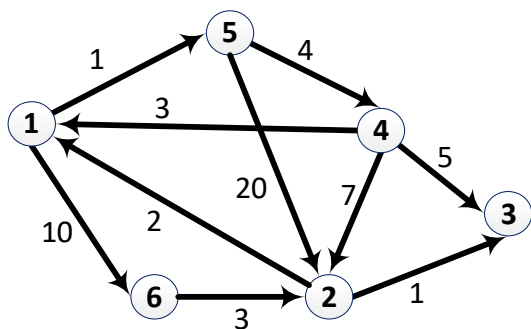
- Pe prima linie sunt două numere reprezentând numărul de vârfuri n ($n > 4$) și numărul de arce m ale grafului, $m > n$
- Pe următoarele m linii sunt câte 3 numere întregi **pozitive** reprezentând extremitatea inițială, extremitatea finală și costul unui arc din graf
- Pe penultima linie este un număr natural b
- Pe ultima linie este un număr s reprezentând un nod sursă în graf.

În punctul s se află un călător care are bugetul b .

a) Să se determine un cel mai depărtat nod v din graf la care călătorul poate ajunge din s printr-un drum (elementar) de cost cel mult b , cât să se încadreze în buget (acel vârf pentru care se obține $\max\{d(s,u) \mid d(s,u) \leq b, u \text{ vârf în } V\}$) și să se afișeze un drum de cost minim de la s la v . Dacă sunt mai multe astfel de noduri se va alege cel cu indicele cel mai mic.

b) Observând că un circuit este format totuși dintr-un drum și un arc, călătorul va mai roagă să determinați în plus dacă poate face un traseu de cost cel mult b care pornește din s și se termina tot în s fără a trece de mai multe ori prin același vârf, altfel spus să determinați dacă există un circuit elementar în G de cost mai mic sau egal cu b care conține s și, în caz afirmativ, să afișați un astfel de circuit. **Complexitate $O(m \log(n))$**

graf.in	Ieșire pe ecran
6 10 1 5 1 1 6 10 2 1 2 4 1 3 5 2 20 5 4 4 4 2 7 4 3 5 2 3 1 6 2 3 11 1	a) v=3 1 5 4 3 b) 1 5 4 1



$d(1, 2) = 12$
 $d(1, 3) = 10$
 $d(1, 4) = 5$
 $d(1, 5) = 5$
 $d(1, 6) = 1$
 $d(1, 7) = 10$
 $b = 11 \Rightarrow$ cele mai mari distanțe mai mici sau egale cu 11 sunt $d(1, 3)$ și $d(1, 7)$

Subiectul 3

Se dau n fabrici de monitoare numerotate $1...n$ și m depozite numerotate $n+1,..., n+m$. Pentru fiecare fabrică i se cunoaște $c(i)$ = câte monitoare au fost produse la momentul curent, iar pentru fiecare depozit j se cunoaște $c(j)$ = numărul de monitoare pe care le poate depozita la momentul curent. Fiecare fabrică are contracte cu anumite depozite. În contractul dintre fabrică i și depozitul j este trecută cantitatea maximă de monitoare care poate fi trimisă spre depozitare de la fabrică i la depozitul j , notată $w(i,j)$. Datele se vor citi din fișierul `fabrics.in` cu următoarea structură:

- pe prima linie sunt numerele naturale n și m
- pe a doua linie este un șir de n numere naturale reprezentând cantitatea de monitoare existente în fiecare dintre cele n fabrici
- pe a treia linie este un șir de m numere naturale reprezentând numărul de monitoare pe care le poate depozita fiecare dintre cele m depozite
- pe a patra linie este un număr k reprezentând numărul de contracte dintre fabrici și depozite
- pe următoarele k linii sunt triplete de numere naturale i j w (separate prin spațiu) cu semnificația: de la fabrică i la depozitul j se pot trimite maxim w monitoare.

Să se determine, dacă există, o modalitate de a depozita toate monitoarele existente în fabrici la momentul curent în depozite respectând condițiile din contracte și capacitatea de depozitare a fiecărui depozit. **Complexitate** $O((n+m)k^2)$

Rezultatul se va afișa sub forma prezentată în exemplul de mai jos.

Observație: Putem modela problema cu un graf bipartit fabrici-depozite (cu vârfuri corespunzătoare fabricilor și depozitelor și muchii reprezentând existența unui contract între fabrică și depozit). Dacă $c(i) = 1$ pentru fiecare fabrică i , $c(j)=1$ pentru fiecare depozit și $w(i, j)=1$ pentru orice contract, atunci problema se reduce la a determina un cuplaj de cardinal maxim în graful bipartit fabrici-depozite și a verifica dacă orice vârf fabrică este saturat.

Se acorda 1p dacă se rezolva doar problema pentru $c(i) = 1$ pentru fiecare fabrică i , $c(j)=1$ pentru fiecare depozit și $w(i, j)=1$ pentru orice contract

fabrics.in	lesire pe ecran (solutia nu este unica)
3 3	1 4 3
6 5 4	1 5 3
7 5 4	2 4 2
7	2 5 2
1 4 7	2 6 1
1 5 5	3 4 2
2 4 3	3 6 2
2 5 2	
2 6 3	
3 4 5	
3 6 2	

