Subjectul 1 (3 puncte)

Se dă un graf neorientat cu n>3 vârfuri și m muchii care nu este bipartit.

- a) Să se determine un ciclu elementar impar în graf (cu număr impar de muchii). Se vor afișa muchiile unui astfel de ciclu. **Complexitate O(n+m)**
- b) Să se determine dacă în grad mai există în graf un alt ciclu în afară de cel afișat la punctul
- a) (nu neapărat impar) și, în caz afirmativ, să se afișeze un astfel de ciclu (diferit de cel de la
- a)); altfel se va afișa un mesaj corespunzător. Complexitate O(n+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

graf.in	lesire pe ecran
5 6	a)
12	2 4
23	45
2 4	5 2
4 5	b)
25	12
15	2 4
	4 5
	5 1

Subjectul 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 pozitive reprezentând extremitatea inițială, extremitatea finală și costul unui arc din graf
- pe următoarea linie (a (m+2)-a linie) din fișier sunt un număr natural k (0<k<n) și un șir de k vârfuri reprezentând vârfurile sursă ale grafului s₁,...,s_k
- pe ultima linie a fișierului sunt două vârfuri t₁ și t₂, reprezentând vârfurile destinație ale grafului (distincte de vârfurile sursă de pe linia anterioară).

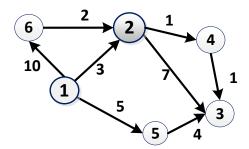
Notăm cu $S = \{s_1,...,s_k\}$ mulțimea vârfurilor sursă din G și cu $T = \{t_1, t_2\}$ mulțimea vârfurilor destinație din G. Spunem că un vârf y este accesibil din x în G dacă există un drum de la x la y.

- a) Să se verifice dacă graful dat este fără circuite și să se afișeze un mesaj corespunzător.
- b) Să se determine un drum de cost maxim care începe cu un vârf din S și se termină cu un vârf din T. Dacă nu există un astfel de drum se va afișa un mesaj corespunzător: niciun vârf destinație nu este accesibil dintr-un vârf sursă.

Complexitate O(m)

Exemplu

2. Action 1. Control 1	
Iesire pe ecran	
drum maxim 1 6 2	



$$S = \{1, 3\}$$

 $T = \{2, 5\}$

Subjectul 3 (3 puncte)

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

Pentru n proiecte, numerotate 1,..., n s-au înscris m studenți numerotați 1,...,m, fiecare student depunând o listă de optiuni cu proiectele la care vrea să participe.

- a) Dat un număr k de la tastatură, să de determine o listă de k asocieri proiect student prin care k studenți diferiți sunt asociați la k proiecte diferite **Complexitate O(km)**
- b) Să se determine, dacă există, o modalitatea de a asocia toți studenții la proiecte astfel încât un student să fie asociat la exact 2 proiecte, iar la un proiect să fie asociați exact 2 studenți și să se afișeze o astfel de modalitate sub forma prezentată în exemplul de mai jos. Altfel se va afișa mesajul "nu este posibil". **Complexitate O(nm)**

Datele despre proiecte și studenți se vor citi dintr-un fișier cu următoarea structură:

- pe prima linie sunt numerele naturale n și m
- pe următoarele linii sunt perechi de numere naturale i j cu $i \in \{1,..., n\}$ și $j \in \{1,..., m\}$ cu semnificația: studentul j s-a înscris la proiectul i.

graf.in	lesire pe ecran (solutia nu este unica)
4 4	a)
11	pentru k=2
12	asocieri proiect - student
13	11
2 1	2 2
2 2	b)
31	asocieri proiect-student
33	11
3 4	12
4 3	2 1
4 4	2 2
	33
(primul este indicele proiectului, al doilea al	3 4
studentului)	4 3
	4 4

