Subiectul 1

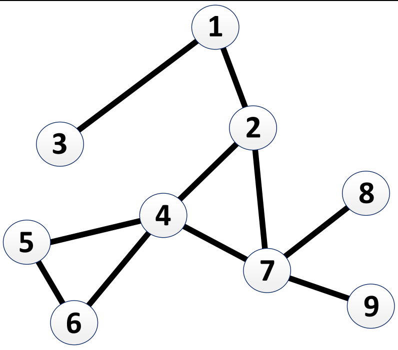
Se dă un graf neorientat conex cu n>3 vârfuri şi m>n muchii. Să se afişeze punctele critice în care sunt incidente muchii critice. Pentru fiecare astfel de punct se va afişa numărul de muchii critice care sunt incidente în el şi numărul de componente biconexe care îl conțin, fără a memora componentele biconexe ale grafului şi fără a memora muchiile critice.

Complexitate O(m)

Informațiile despre graf se citesc din fişierul graf.in cu structura:

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

graf.in	Iesire pe ecran (nu neaparat in aceasta ordine)
9 10 1 2 1 3 2 4 2 7 4 7 4 5 4 6 5 6 7 8 7 9	Puncte critice cerute: 1: incidente 2 muchii critice este in 2 componente biconexe 2: incidente 1 muchii critice este in 2 componente biconexe 7: incidente 2 muchii critice este in 3 componente biconexe



Subiectul 2

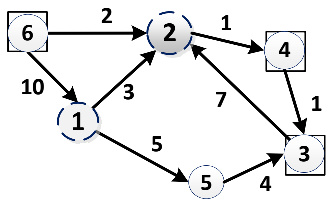
Se citesc informații despre un graf **orientat** ponderat G din fișierul `graf.in`. Fișierul are următoarea structură:

- Pe prima linie sunt două numere reprezentând numărul de vârfuri n (n>4) și numărul de arce m ale grafului, **m>n**
- Pe a doua linie din fișier sunt un număr natural k (0<k<n) și un șir de k vârfuri reprezentând vârfurile sursă ale grafului $s_1,\dots,s_k$
- Pe a treia linie a fișierului sunt trei vârfuri, reprezentând vârfurile destinație t_1, t_2, t_3 din G.
- Pe următoarele m linii sunt câte 3 numere întregi **pozitive** reprezentând extremitatea inițială, extremitatea finală și costul unui arc din graf

Notăm cu $S = \{s_1,\dots,s_k\}$ mulțimea vârfurilor sursă din G și cu $T=\{t_1, t_2, t_3\}$ 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. Să de determine pentru fiecare vârf destinație $t \in T$ un vârf sursă $s \in S$ cu proprietatea că t este accesibil din s și distanța de la s la t este minimă (s este o sursă din care se poate ajunge cel mai repede în t) și să se afișeze un drum minim de la s la t. Dacă nu există o astfel de sursă se va afișa un mesaj corespunzător. **Complexitate O(mlog(n))**

graf.in	Iesire pe ecran

`6 8`
`2 1 2`
`3 4 6`
`1 2 3`
`6 1 10`
`6 2 2`
`2 4 1`
`4 3 1`
`5 3 4`
`1 5 5`
`3 2 7`
	`t=3 s=2 drum minim 2 4 3`
	`t=4 s=2 drum minim 2 4`
	`t=6 nu exista s`



k=2 , S = {1, 2}

t_1=3, t_2= 4, t_3=6 $\Rightarrow$ T={3,4,6}

t=3: distanta(1,3)=5, distanta(2,3)=2

Cea mai mică este distanta(2,3) $\Rightarrow$ s=2, drum minim 2 4 3

t=4: distanta(1,4)=4, distanta(2,4)=1 $\Rightarrow$ s=2, drum minim 2 4

t=6: distanta(1,6)= ∞, distanta(2,6)=∞ $\Rightarrow$ nu există s

Subiectul 3

Fișierul graf.in conține următoarele informații despre un graf **bipartit** conex:

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

Se consideră graful G dat în fișierul 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 rețea de transport, determinați un cuplaj maxim în subgraful indus de mulțimea vârfurilor de grad impar din G.

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

- 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 conține 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