Varianta 1

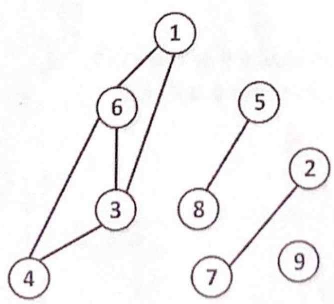
Subiectul 1 (2 puncte)

Se dau un graf neorientat G cu n>3 vârfuri și m muchii și un număr natural k mai mic decât numărul de componente conexe ale lui G.
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 line este numărul k

O mutare de tip *cut-paste* aplicată asupra muchiilor lui G constă într-o succesiune de două operații: eliminarea unei muchii (x,y) din G urmată de adăugarea unei muchii (u,v) între două vârfuri neadiacente din graful obținut. Vom nota această operație prin (x,y) => (u,v).
Dorim să transformăm G într-un graf cu exact k componente conexe folosind astfel de mutări (!cât mai puține).
Să se determine dacă există un șir de mutări de tip *cut-paste* pe care să le efectuăm astfel încât în final să obținem un graf cu exact k componente conexe. Dacă nu există un astfel de șir se va afișa mesajul „nu se poate". Dacă există, se vor afișa și un șir de mutări prin care putem face acest lucru de lungime minimă.
Complexitate O(n+m)

graf.in	***Iesire pe ecran*** soluția nu este unică
9 7 1 6 3 6 4 6 3 4 5 8 2 7 1 3 2	minim 2 mutari cut-paste (1,3) => (2,9) (3,4) => (2,5)



Varianta 1

Subiectul 2 (2 puncte)

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

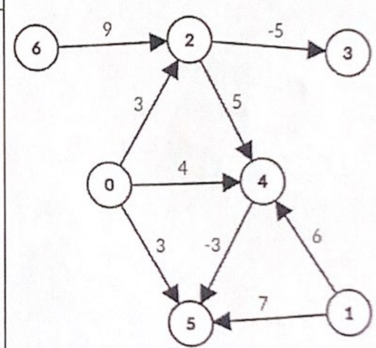
- pe prima linie sunt două numere reprezentând numărul de vârfuri n (n>4) și numărul de arce m ale grafului, **m>n**
- pe următoarele m linii sunt câte 3 numere întregi reprezentând extremitatea inițială, extremitatea finală și costul unui arc din graf (vârfurile sunt numerotate de la 0)
- pe următoarea linie (a (m+2)-a linie) din fișier este un număr natural k (0<k<10^6)

Determinați cel mai costisitor drum din graf (adică un drum de cost maxim) care trece doar prin arce cu cost mai mic sau egal cu k.

Complexitate O(m)

Exemplu

graf.in	Iesire pe ecran
7 9	Drumul este:
0 2 3	0 2 4
0 4 4	
0 5 3	
1 4 6	
1 5 7	
2 3 -5	
2 4 5	
4 5 -3	
6 2 9	
7	



Explicații: “Cel mai costisitor drum” este 6 2 4 dar el trece prin arcul 6 9 de cost 9 > 7. Cel mai costisitor drum care nu trece prin arce cu cost mai mare 7 este de la 0 la 4 via 2.

Varianta 1

Subiectul 3 (2 puncte)

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

La un examen participă n clase, numerotate 1,2,...,n. Pentru fiecare clasă i se cunoaște numărul de elevi e_i. Profesorul a pregătit m variante de subiecte numerotate 1, 2, ..., m, iar dintr-o variantă i a tipărit v_i exemplare. Profesorul ar vrea să împartă subiectele pe clase, astfel încât la o clasă cel mult x elevi să aibă același subiect. Mai mult, la fiecare clasă are o listă cu variantele pe care ar vrea să le dea.

Se dă un fișier cu informațiile de mai sus organizat astfel:

- pe prima linie sunt numerele naturale n și m
- pe a doua linie sunt valorile $e_1, e_2, \ldots, e_n$ separate prin spațiu (reprezentând numărul de elevi din fiecare clasă)
- pe a treia linie sunt valorile $v_1, v_2, \ldots, v_m$ separate prin spațiu (reprezentând numărul de exemplare din fiecare variantă)
- pe a patra linie este numărul x (reprezentând numărul maxim de elevi dintr-o clasă care pot avea aceeași variantă de subiect)
- pe următoarele n linii sunt numere naturale, pe linia a i-a din cele n linii fiind variantele pe care profesorul ar vrea să le dea la clasa i.

a) Scrieți un program care decide dacă se poate face o distribuire a variantelor pe clase care să respecte condițiile cerute de profesor; în caz afirmativ, afișați și o astfel de distribuire sub forma indicată în exemplu (pentru fiecare clasă se va afișa câte subiecte să distribuie din fiecare variantă)

b) În cazul în care răspunsul la a) este negativ, determinați dacă, în cazul în care valoarea lui x crește cu 1 (profesorul este dispus să accepte ca x+1 elevi dintr-o clasă să aibă același subiect), problema are soluție și, în caz afirmativ, care este aceasta – afișată în același format ca la a)

variante.in	Ieșire pe ecran (solutia nu este unica)
3 3 7 8 1 6 9 2 8 1 2 1 2 1 3	DA Clasa 1: 1 6 0 Clasa 2: 5 3 0 Clasa 3: 0 0 1 *(Explicații afișare: în clasa 1 : 1 elev va avea varianta 1, 6 elevi varianta 2 și 0 varianta 3)*
variante.in	**Ieșire pe ecran (solutia nu este unica)**
2 2 6 6 6 6 2 1 2 1 2	NU DA Clasa 1: 3 3 Clasa 2: 3 3

Explicații: În al doilea exemplu cu 2 lucrări nu pot distribui 6 lucrări la fiecare clasa dar cu 3 se poate.