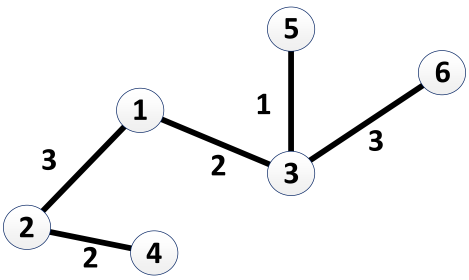
Subiectul 1 (3 puncte)

Se dă un graf neorientat ponderat G cu n>3 vârfuri, m muchii și un vârf s.
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 un vârf sursă s.

Pentru un lanț P în G definim ponderea lanțului P ca fiind produsul ponderilor muchiilor care îl compun.
Dacă G este arbore, să se afișeze pentru fiecare vârf v ponderea unicului lanț elementar de la s la v (sub forma v: pondere lanț de la s la v), altfel să se afișeze un arbore parțial al componentei care conține s. **Complexitate O(n+m)**

graf.in	***Iesire pe ecran***
6 5 1 2 3 1 3 2 2 4 2 3 5 1 3 6 3 1	Este arbore 1: 0 2: 3 3: 2 4: 6 5: 2 6: 6



Subiectul 2 (3 puncte)

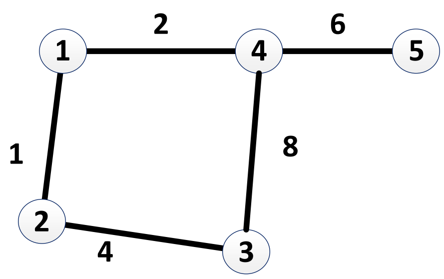
Se citesc informații despre un graf **neorientat** ponderat conex G din fișierul `graf.in`. Fișierul are următoarea structură:

- pe prima linie sunt două numere reprezentând numărul de vârfuri n (n>4) și numărul de muchii m ale grafului, **m>n**
- pe următoarele m linii sunt câte 3 numere pozitive reprezentând extremitatea inițială, extremitatea finală și costul unei muchii din graf
- pe ultima linie sunt un număr natural k (0<k<n) și un șir de k vârfuri reprezentând vârfurile sursă ale grafului $s_1,\ldots,s_k$

Să se afișeze muchiile unui graf parțial al lui G în care fiecare vârf v este conectat prin cel puțin un lanț de o sursă. Dacă există mai multe astfel de grafuri parțiale, să se determine unul cu costul total minim. Se vor afișa muchiile acestui graf parțial. De asemenea, să se afișeze și care este vârful sursă cel mai important **în acest graf parțial**, importanța unui vârf sursă fiind dată de numărul de vârfuri accesibile din acea sursă (care sunt conectate la acea sursă prin lanț). **Complexitate O(mlog(n))**.

Exemplu

graf.in	**Iesire pe ecran (nu conteaza ordinea în care sunt afisate muchiile)**
5 5 1 2 1 1 4 2 2 3 4 3 4 8 4 5 6 2 1 2	1 4 4 5 2 3 1



Două surse: 1 și 2

În graful parțial cu muchiile

1 4

4 5

2 3

din sursa 1 sunt accesibile încă două vârfuri (4 și 5), iar din 2 doar unul (3), deci 1 este mai important

Subiectul 3 (3 puncte)

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

Într-un restaurant sunt n mese numerotate 1,...,n sunt și m ospătari numerotați 1,..., m. Proprietarul restaurantului urmează să aibă un eveniment în restaurant și dorește să repartizeze fiecărui ospătar mesele de care trebuie să se ocupe. El întreabă pe fiecare ospătar la ce mese ar vrea să servească, altfel că pentru fiecare ospătar j știe lista meselor pe care le preferă. Proprietarul ar vrea ca la fiecare masă să fie exact k ospătari și un ospătar să servească la cel mult p mese și ar vrea să respecte și preferințele ospătarilor legate de mese.

Scrieți un program care, dacă este posibilă o distribuție a ospătarilor la mese care să respecte dorințele proprietarului, cu **cel mult** o excepție și anume o masă la care să fie doar k-1 ospătari, să afișeze o astfel de distribuție sub forma prezentată în exemplul de mai jos (perechi masa ospătar). Altfel se va afișa mesajul "imposibil"

Datele despre restaurant și opțiunile ospătarilor se vor citi dintr-un fișier cu următoarea structură:

- pe prima linie sunt numerele naturale n, m, k, p
- pe următoarele m linii sunt numere naturale, pe linia a j-a din cele m fiind numărul de mese la care vrea să servească ospătarul j și care sunt indicii acestor mese

Complexitate $O(n^2m^2)$

restaurant.in	Iesire pe ecran (solutia nu este unica)
4 7 2 1 **4** 1 2 3 4 **1** 1 **2** 2 4 **2** 2 3 **2** 1 4 **3** 1 3 4 **1** 1 Explicații: Ospătarul 1 prefera 4 mese: 1,2,3,4, ospătarul 2 preferă masa 1, ospătarul 3 preferă 2 mese: 2 și 4 etc	1 2 1 7 2 3 2 4 3 1 3 6 4 5 (primul indice reprezintă masa, iar al doilea ospătarul asociat)