Subiectul 1 (3 puncte)

Se dă un graf neorientat cu $n>3$ vârfuri și m muchii și un vârf s.
a) Adăugați la G un număr minim de muchii astfel încât să devină conex. Construiți în memorie și afișați pe ecran listele de adiacență ale grafului astfel obținut. **Complexitate O(n+m)**
b) Determinați excentricitatea ecc(s) a vârfului s în noul graf G_1 obținut la a):

$$ecc(s) = \max(d(s,v) | v \text{ vârf în } G_1\}$$

unde d(s,v) este distanța de la s la v.
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
- pe ultima linie este vârful s

graf.in	***Iesire pe ecran (solutia nu este unica)***
6 4 1 3 1 5 3 5 2 4 6	a) 1 2 2 6 b) 3 Explicații: d(6,1)=2, d(6,2)=1, d(6,3)=3, d(6,4)=2, d(6,5)=3, deci ecc(6)=3

Subiectul 2 (3 puncte)

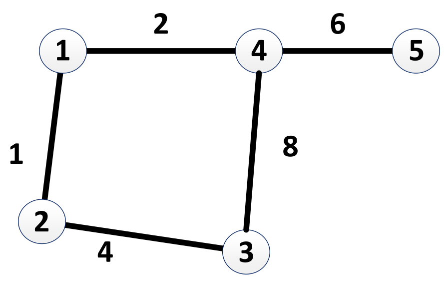
Se citesc informații despre un graf **neorientat** ponderat conex G din fișierul `graf.in`. Fișierul are următoarea structură:

- 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

a) Să se afișeze costul unui arbore parțial de cost minim în G. **Complexitate O(mlog(n))**.
b) Se citesc de la tastatură două muchii **noi** date tot prin extremitatea inițială, extremitatea finală și cost. Știind că **doar una** dintre aceste muchii se va adăuga la graful G, decideți pe care o adăugați astfel încât noul graf să aibă un arbore parțial de cost minim cu cost cât mai mic și afișați muchiile unui arbore parțial de cost minim în acest graf. **Complexitate O(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 Intrare de la tastatura 3 5 5 1 3 5	a) 13 b) adaugam 3 5 1 2 1 4 2 3 3 5



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)