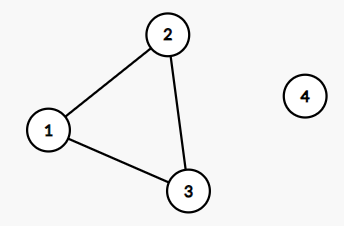
***CIURESCU IRINA - ALEXANDRA***

***GRUPA 242***

**TEMA 1 – Seminar AF**

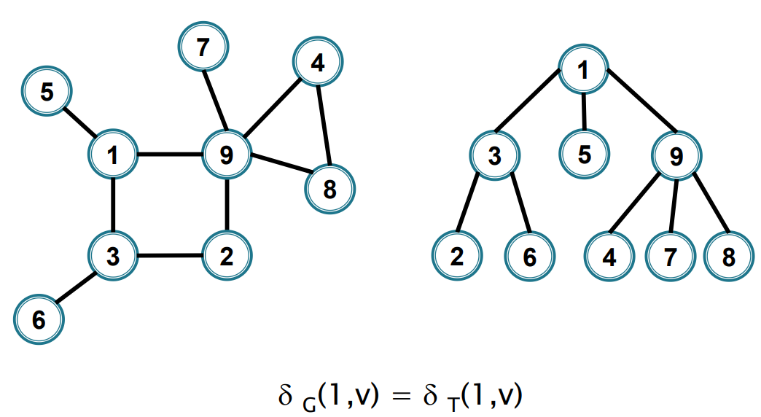
1. **Nu orice graf** cu n>=1 noduri și n-1 muchii este arbore deoarece:

* dacă graful are n noduri și exact n−1 muchii ce sunt dispuse de exemplu între n−1 noduri, iar un nod rămâne izolat, graful nu este conex și nici aciclic -> **graful nu este arbore**



1. Se pot elimina maxim **|E| – (n-1)** muchii deoarece:

* Graful partial G’ va avea **(n-1)** muchii și anume muchiile arborelui BF de rădăcină 1 (notat T) obținut în urma aplicării algoritmului BF(1) , deci G’ este conex.
* La finalul algoritmului BF(1), pentru orice vârf v avem d[v]=dG(1,v).
* În plus, arborele T păstrează distanțele din graf de la 1 la celelalte vârfuri v, deci dT(1,v)=dG(1,v).



1. **Nu este obligatoriu ca G1=G2** deoarece:

* Chiar dacă G1≠G2 se pot obține arbori BFS și DFS egali în urma parcurgerii în lățime și în adâncime a grafurilor G1 și G2 din vârful 1.
* Am plecat de la arborii egali BFS și DFS. Am adăugat inițial la G1 și G2 toate muchiile din cei 2 arbori. Apoi am adăugat câte o muchie la fiecare graf, muchie care nu influența ordinea de vizitare a nodurilor (muchii de întoarcere sau de traversare) cum ar fi muchiile (2,5) din G1 și (3,5) din G2.
* Am obținut astfel 2 grafuri cu număr egal de noduri și muchii, care nu sunt egale, dar care au arborii BFS și DFS identici.

|  |  |  |
| --- | --- | --- |
| **Graful G1** | **Arborele BFS** | **Arborele DFS** |
|  |  |  |
| **Graful G2** |
|  |

1. **L1 = L2** deoarece:

* Dacă T1 și T2 sunt doi arbori de cost minim, aceștia au aceeași sumă totală a costurilor muchiilor, deoarece sunt arbori de cost minim pentru același graf.
* Presupunem că L1 ≠ L2. Atunci unul dintre arbori ar avea un cost total mai mic decât celălalt, ceea ce contrazice premisa că T1 și T2 sunt arbori de cost minim pentru același graf.

1. **Algoritm**:

* Se determină arborele parțial de cost minim (notat T) pentru graful G=(V,E) (V=mulțimea punctelor de interes, E=distanțele dintre punctele de interes) cu algoritmul lui Prim, graful fiind complet.
* Aplicăm DF din nodul (x1,y1) pe arborele T.
* Fiecare muchie (x,y) va fi parcursă/afișată de 2 ori (la descoperirea nodului y cand am folosit muchia (x,y) ca muchie de arbore și la finalizarea nodului y când ne vom reintoarce in nodul tată x ).
* Cost drum = 2 \* cost APM.
* Complexitate O(n2)