

FIȘA DE LABORATOR 7

Grafuri și arbori

A. Grafuri neorientate. Aplicații

I. Adiacență, incidență, grad

- Din fișierul `graf.in` se citește de pe prima linie valorile n (numărul de noduri, $n < 100$) și m (numărul de muchii) ale unui graf neorientat $G = (X, U)$. De pe următoarele m linii se citesc cele m muchii ale grafului. Scrieți câte o funcție pentru:
 - citirea grafului și construirea matricei de adiacență;
 - afișarea matricei de adiacență a grafului;
 - afișarea gradele tuturor nodurilor (se va folosi o funcție care primește ca parametrii n numărul de noduri, a matricea de adiacență și x un nod din graf și care returnează gradul nodului x);
 - afișarea nodurilor de grad maxim;
 - afișarea nodurilor terminale;
 - afișarea nodurilor izolate;
 - afișarea nodurilor în ordinea creșcătoare a gradelor.
- Din fișierul `graf.in` se citesc de pe prima linie valorile n (numărul de noduri, $n < 100$) și m (numărul de muchii). De pe următoarele m linii se citesc muchiile grafului. Fără a construi matricea de adiacență a grafului, scrieți câte o funcție pentru:
 - citirea datelor și construirea matricei muchiilor;
 - afișarea matricei muchiilor;
 - afișarea gradele tuturor nodurilor (se va folosi o funcție care primește ca parametrii n numărul de noduri, a matricea muchiilor și x un nod din graf și care returnează gradul nodului x);
 - afișarea nodurilor de grad maxim;
 - afișarea nodurilor terminale;
 - afișarea nodurilor izolate;
 - afișarea nodurilor în ordinea creșcătoare a gradelor.
- Fișierul `matrice.txt` conține pe primul rând un număr natural n ($n < 100$), iar de pe următoarele n linii elementele unei matrice pătratice. Să se determine dacă matricea dată poate fi matricea de adiacență a unui graf neorientat, afișând pe ecran un mesaj corespunzător. În caz afirmativ să se determine și să se afișeze pe ecran:
 - numărul muchiilor grafului;
 - muchii grafului;
 - lista de adiacență a grafului.

II. Lanțuri și cicluri

- Fișierul `matrice.txt` conține pe primul rând un număr natural n ($n < 100$), iar pe următoarele n linii elementele unei matrice pătratice. De la tastatură se citește o secvență de k noduri (k citit de la tastatură).

- a) testare lanț - să se verifice dacă secvența de noduri dată este lanț (se va afișa pe ecran un mesaj corespunzător), în caz afirmativ se va determina dacă lanțul este elementar sau ne-elementar;
- b) testare ciclu - să se verifice dacă o secvență de noduri dată este ciclu (se va afișa pe ecran un mesaj corespunzător), în caz afirmativ se va determina dacă ciclul este elementar sau neelementar;
- c) generarea lanțurilor elementare între două noduri date - să se genereze și să se afișeze pe ecran toate lanțurile elementare între 2 vârfuri x și y , citite de la tastatură;
- d) generarea tuturor lanțurilor elementare - să se genereze și să se afișeze pe ecran toate lanțurile elementare din graf;
- e) matricea lanțurilor: algoritmul Roy-Warshall - să se afișeze pe ecran matricea lanțurilor între oricare două vârfuri.

III. Matricea costurilor. Determinarea matricei lanțurilor

5. De la tastatură se dă un graf neorientat prin: n numărul de noduri, m numărul de muchii, lista muchiilor și costul fiecărei muchii ($n < 100$). Să se afișeze pe ecran:
 - a) matricea costurilor și costul mediu pentru graful dat;
 - b) matricea costurilor minime, costurile lanțurilor minime: algoritmul Roy-Floyd.

IV. Tipuri de grafuri

6. Afișarea tuturor grafurilor neorientate cu n noduri. Se citește de la tastatură un număr natural nenul n ($n < 10$). Să se determine și să se afișeze toate grafurile neorientate care se pot forma cu n noduri. Pentru fiecare graf se va afișa pe ecran matricea sa de adiacență.
7. Verificare graf complet. Se citește de la tastatură n număr natural nenul ($n < 10$) și a o matrice pătratică simetrică față de diagonal principal. Să se determine dacă graful dat este graf complet, afișând pe ecran un mesaj corespunzător. Dacă graful dat nu este graf complet să se afișeze pe ecran muchiile care trebuie adăugate astfel încât acesta să devină graf complet, afișând muchiile pe linii distincte ale ecranului.
8. Verificare graf partial/subgraf. Se citește de la tastatură datele a două grafuri neorientate G_1 și G_2 , astfel:
 - n_1 numărul de noduri ($n_1 < 10$), m_1 numărul de muchii și lista muchiilor, reprezentând primul graf;
 - n_2 numărul de noduri ($n_2 < 10$), m_2 numărul de muchii și lista muchiilor, reprezentând al doilea graf;
 - a. să se verifice dacă graful G_2 graf dat este graf partial al grafului G_1 , afișând pe ecran un mesaj corespunzător;
 - b. să se verifice dacă graful G_2 graf dat este subgraf al grafului G_1 , afișând pe ecran un mesaj corespunzător; lista celor n_2 noduri ale grafului G_2 se citește de la tastatură.

V. Parcurgerea grafurilor

10. Fișierul `matrice.txt` conține pe primul rând un număr natural n ($n < 100$), iar de pe următoarele n linii elementele unei matrice pătratice.
- să se afișeze pe ecran nodurile grafului prin parcurgerea în lățime (BFS – Breadth First Search);
 - să se afișeze pe ecran nodurile grafului prin parcurgerea în adâncime (DFS – Depth First Search);

VI. Conexitate. Componente conexe

11. Fișierul `matrice.txt` conține pe primul rând un număr natural n ($n < 100$), reprezentând numărul de noduri ale unui graf neorientat, iar de pe următoarele n linii elementele unei matrice pătratice, reprezentând matricea de adiacență a unui graf neorientat. Să se scrie un program C++ care:
- să afișeze pe ecran mesajul **Graful este conex**, dacă graful dat este conex, sau mesajul **Graful nu este conex** în caz contrar, (utilizând BFS și DFS);
 - să afișeze pe ecran componentele conexe ale grafului dat; pentru fiecare componentă conexă se afișează nodurile componente, (utilizând BFS și DFS).

B. Grafuri orientate. Aplicații

I. Adiacență, incidență, grade

1. Din fișierul `graf.in` se citește de pe prima linie valorile n (numărul de vârfuri, $n < 100$) și m (numărul de arce) ale unui graf orientat $G = (X, U)$. De pe următoarele m linii se citesc cele m arce ale grafului. Scrieți câte o funcție pentru:
- citirea grafului și construirea matricei de adiacență;
 - afișarea matricei de adiacență a grafului;
 - afișarea gradelor exterioare ale tuturor vârfurilor (se va folosi o funcție care primește ca parametrii n numărul de vârfuri, a matricea de adiacență și x un vârf din graf și care returnează gradul exterior al vârfului x);
 - afișarea gradelor interioare ale tuturor vârfurilor (se va folosi o funcție care primește ca parametrii n numărul de vârfuri, a matricea de adiacență și x un vârf din graf și care returnează gradul interior al vârfului x);
 - afișarea vârfurilor care au gradul exterior maxim;
 - afișarea vârfurilor care au gradul interior minim;
 - afișarea vârfurilor terminale;
 - afișarea vârfurilor izolate.

II. Tipuri de grafuri

2. Verificare graf complet. Fișierul text `graf.txt` conține:
- pe primul rând n număr natural nenul ($n < 10$), reprezentând numărul de vârfuri ale grafului,

- pe al doilea rând m număr natural, reprezentând numărul de arce ale grafului și
 - pe următoarele m linii m perechi de numere reprezentând extremitățile arcelor grafului.
- Să se determine dacă graful dat este graf complet, afișând pe ecran un mesaj corespunzător.

3. Verificare graf turneu. Fișierul text `graf.txt` conține:

- pe primul rând n număr natural nenul ($n < 10$), reprezentând numărul de vârfuri ale grafului și
- pe următoarele n rânduri cele n linii ale unei matrice pătratice a , reprezentând matricea de adiacență a unui graf orientat.

Să se determine dacă graful dat este graf turneu, afișând pe ecran un mesaj corespunzător.

III. Tare conexitate. Componente tare conexe

4. Fișierul `matrice.txt` conține pe primul rând un număr natural n ($n < 100$), reprezentând numărul de vârfuri ale unui graf orientat, iar de pe următoarele n linii elementele unei matrice pătratice, reprezentând matricea de adiacență a unui graf orientat. Să se scrie un program C++ care:

- să afișeze pe ecran mesajul **Graful este tare conex**, dacă graful dat este tare conex, sau mesajul **Graful nu este tare conex** în caz contrar;
- să afișeze pe ecran componentele tare conexe ale grafului dat; pentru fiecare componentă tare conexă se afișează vârfurile componente.