Teoria grafurilor – rezumat pentru examenul de bacalaureat

Grafuri neorientate

Definiție.Se numește **graf neorientat** o pereche de două mulțimi (X, U) unde X este o mulțime finită și nevidă de elemente numite **noduri** sau **vârfuri**, iar U este o mulțime de perechi **neordonate** de elemente din X numite **muchii**.

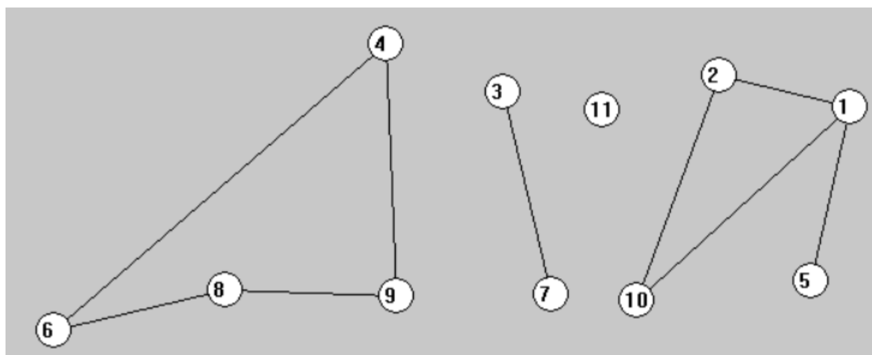
Vom nota cu G=(X,U) un graf neorientat. Mulțimea X se numește **mulțimea nodurilor** sau a **vârfurilor** grafului G iar U, **mulțimea muchiilor**.

O muchie u∈U este deci o submulțime de două vârfuri distincte x și y din X și se notează cu [x,y]. Vom spune despre vârfurile x și y că sunt **adiacente** în G iar despre u și x că sunt **incidente** (analog despre u și y). Se mai spune că x și y sunt **extremitățile** muchiei u. Muchia [x,y] este identică cu [y,x].

Dacă u1 și u2 sunt două muchii care au o extremitate comună, ele vor fi numite de asemenea **incidente**.

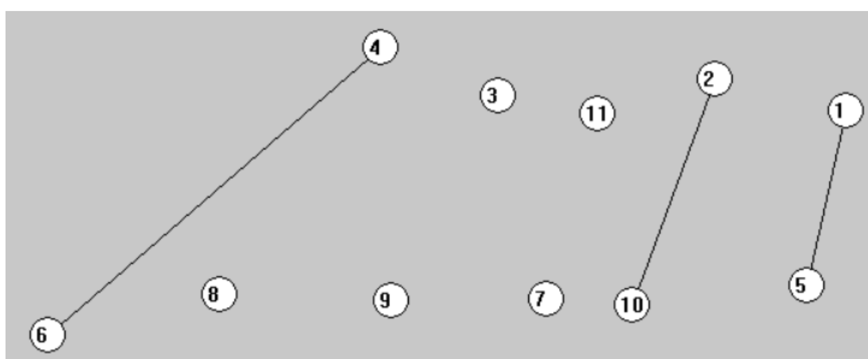
Un aspect important al teoriei grafurilor este aspectul geometirc. Un graf poate fi reprezentat cu ajutorul unei figuri plane în care fiecărui vârf i se poate asocia un punct și fiecărei muchii [x,y] îi putem asocia un segment sau o linie care unește punctele x și y. Evident se naște concluzia că orice graf admite o infinitate de reprezentări planare.

Exemplu.
X={1,2,3,4,5,6,7,8,9,10,11} iar U={[1,5], [3,7], [4,6], [9,8], [10,2], [1,2], [9,4], [1,10], [6,8]}. O posibilă reprezentare planară a grafului G=(X,U) poate fi cea alăturată.



Definiție. Un **graf parțial** al grafului G=(X,U) este un graf G1=(X,V) astfel încât V⊆U.
Cu alte cuvinte, un graf parțial al unui graf se obține păstrând aceeași mulțime de vârfuri și eliminând o parte din muchii. Se mai spune că un graf parțial G1=(X,V) este indus de mulțimea V de muchii.

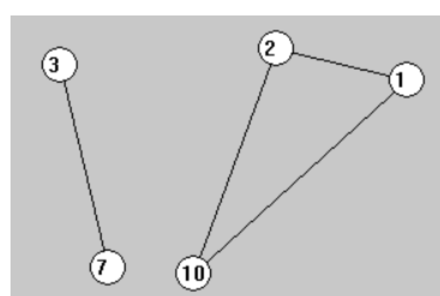
Exemplu. Pentru graful G din figura de mai sus, G1=(X,V) unde V={[1,5], [2,10], [6,4]}, reprezentat în figura alăturată, este un graf parțial.



Definiție. Un **subgraf** al unui graf G=(X,U) este un graf H=(Y,V) astfel încât Y⊆X iar V conține toate muchiile din U cu proprietatea că au ambele extremități în Y. Vom spune că subgraful H este indus sau generat de mulțimea de vârfuri Y.

Prin urmare, un subgraf al unui graf se obține eliminând o parte din vârfuri și toate muchiile incidente cu acestea.

Exemplu. Pentru graful G inițial, dacă Y={1,2,3,7,10} și V={[1,2], [1,10], [2,10], [3,7]} atunci H=(Y,V) este un subgraf al lui G.



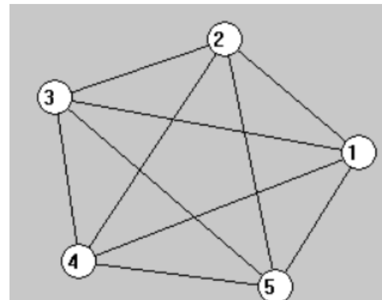
Teoremă. Numărul de grafuri neorientate cu n vârfuri date X={1,2,…n} este $2^{C_n^2}$.

Teorema se poate demonstra folosind faptul că pentru două mulțimi A și B cu a, respectiv b elemente, există b^a funcții definite pe A cu valori în B. Mulțimea A va fi formată din toate submulțimile cu două elemente din X iar mulțimea B={0,1}.

Definiție. **Gradul** unui vârf x este numărul muchiilor incidente cu x (adică numărul de muchii la care una dintre extremități este x). Se notează cu d(x) (din limba franceză „degre”).

Un vârf care are gradul 0 (nu există muchii incidente cu el) se numește **vârf izolat**. Un vârf care are gradul 1 (este extremitatea unei singure muchii) se numește **vârf terminal**. În graful inițial G, vârful 11 este izolat iar 5 vârf terminal.

Consecință. Dacă un graf G=(X,U) are n vârfuri 1,2,...n (sau altfel numerotate) și m muchii atunci **suma gradelor vârfurilor** este 2m.



Graful complet K5

Demonstrația este evidentă observând că fiecare muchie [x,y] contribuie cu o unitate la gradul lui x și cu o unitate la gradul lui y.

Consecință. În orice graf G există un număr par de vârfuri de grad impar.

Definiție. Se numește **graf complet** cu n vârfuri un graf care are proprietatea că oricare două vârfuri diferite sunt adiacente. Un graf complet cu n vârfuri se notează prin K_n.

Consecință. Graful complet K_n are C_n^2 muchii.

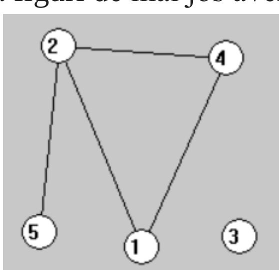
Reprezentarea grafurilor neorientate în memorie.

Fie G=(X,U) un graf neorientat cu n vârfuri. Deoarece între mulțimea X cu n elemente și mulțimea {1,2,…,n} există o bijecție, putem presupune, fără a restrânge generalitatea, că X={1,2,…,n}. Reprezentarea unui graf în memorie înseamnă alegerea unei modalități de a reține anumite date despre graf astfel încât acestea să identifice graful în mod unic.

Matricea de adiacență. Este o matrice pătratică și binară de ordinul n definită astfel:

`a[i][j]=1`, dacă [i,j] ∈U, respectiv `a[i][j]=0` dacă nu există muchia [i,j].

Din modul de definire a matricei de adiacență rezultă că ea este simetrică față de diagonala principală. În primele două figuri de mai jos avem un graf cu 5 vârfuri și matricea lui de adiacență.



Matrice adiacență

0	1	0	1	0
1	0	0	1	1
0	0	0	0	0
1	1	0	0	0
0	1	0	0	0

Vârf	Vecini
1	2, 4
2	1, 4, 5
3	
4	1, 2
5	2

Liste de adiacență. Se precizează numărul n de vârfuri și pentru fiecare vârf i, lista L_i a vecinilor săi, adică lista vârfurilor j pentru care [i,j] ∈U.

Alfel spus, pentru fiecare vârf x se creează o listă cu vârfurile adiacente cu x. În cea de-a treia figură de mai sus este reprezentat graful folosind liste de adiacență.

Vector de muchii. Se definește un tip ”muchie” care să rețină două elemente de tip întreg (extremitățile muchiei) apoi un vector unidimensional de tip „muchie”. În limbajul C++ modalitatea este prin intermediul unei structuri:

```
struct muchie { int x,y; } U[1000];
```

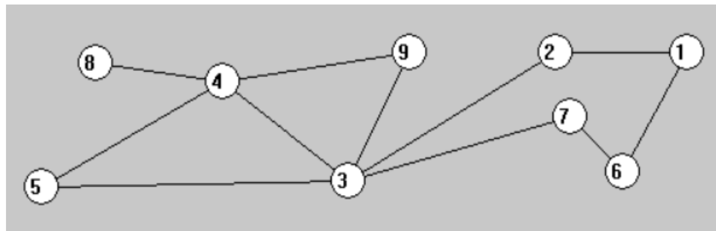
În declarația de mai sus tabloul U poate memora cel mult 1000 de muchii cu extremitățile în x și y.

Conexitate.

Fie G=(X,U) un graf neorientat cu n vârfuri, X={x_1,x_2,…x_n}.

Definiție. Se numește **lanț** în G o succesiune de vârfuri L=[y_1, y_2, ..., y_k] cu proprietatea că oricare două vârfuri consecutive din L sunt adiacente, adică [y_1,y_2], [y_2,y_3],…[y_{k-1},y_k] $\in$U. Vârfurile y_i și y_{i+1} (i=1,2,...,k-1) se numesc **extremitățile** lanțului iar numărul de muchii din care este format lanțul se numește **lungimea** lanțului.

Proprietate. Dacă vârfurile y_1, y_2, ..., y_k sunt diferite două câte două atunci lanțul se numește **elementar**.



Exemplu. În graful din figura alăturată, L1=[1, 2, 3, 5, 4, 8], L2=[1, 2, 3, 2], L3=[9, 3, 5, 4, 3, 2] și L4=[6, 3, 5, 4, 8] sunt lanțuri, L1 și L4 fiind elementare. Lanțul L1 are lungimea 5.

Definiție. Se numește **ciclu** în graful G un lanț L cu proprietatea că y_1=y_k și toate muchiile [y_1,y_2], [y_2,y_3],…[y_{k-1},y_k] sunt diferite două câte două.

Exemplu. În graful din figura de mai sus, C1=[3, 4, 5, 3, 7, 6, 1, 2, 3] este un ciclu.

Definiție. Se numește **ciclu elementar** un ciclu cu proprietatea că oricare două vârfuri, cu excepția primului și a ultimului, sunt diferite două câte două.

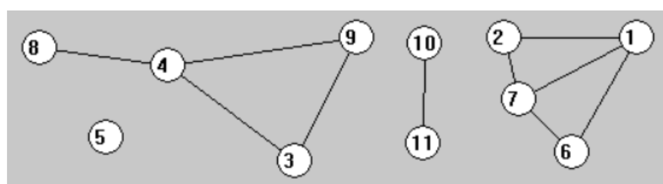
Exemplu. Ciclul C2=[3, 4, 5, 3] din același graf este elementar pe când C1 nu este deoarece vârful 3 se repetă.

Definiție. Două cicluri C1 și C2 sunt **egale/echivalente** dacă muchiile lor induc același graf parțial al subgrafului generat de vârfurile ce aparțin lui C1, respectiv lui C2.

Exemplu. Ciclurile C2 și C3=[4, 5, 3, 4] reprezintă de fapt același ciclu.

Definiție. Un graf G se numește **conex** dacă pentru oricare două vârfuri diferite x și y, există cel puțin un lanț ale cărui extremități să fie aceste vârfuri.

Exemplu. Graful din figura de mai sus este conex iar cel din figura alăturată nu este deoarece nu există niciun lanț care să lege vârfurile 9 și 6.

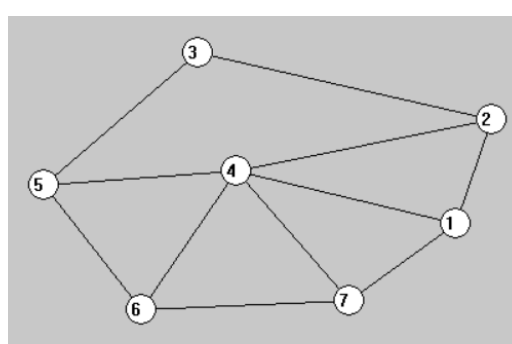


Definiție. Se numește **componentă conexă** a grafului G=(X,U) un subgraf H=(Y,V) conex și cu proprietatea că nu există niciun lanț în G care să lege un vârf din Y cu un vârf din X-Y.

Definiție echivalentă. Se **numește componentă** conexă a grafului G=(X,U) un subgraf H=(Y,V) conex și maximal în raport cu proprietatea de conexitate (adică subgraful H nu se poate extinde prin adăugarea de vârfuri din G).

Exemplu. În figura de mai sus, considerând că Y={3, 4, 8, 9} și V={[8,4], [4,3], [4,9], [3,9]} obținem subgraful H=(Y,V) care este o component conexă. Graful din figură are 4 componente conexe (un vârf sau o muchie definesc o componentă conexă).

Consecință. Dacă un graf are k componente conexe, atunci este nevoie de cel puțin k-1 muchii pentru a-l face conex (câte o muchie între două componente conexe).

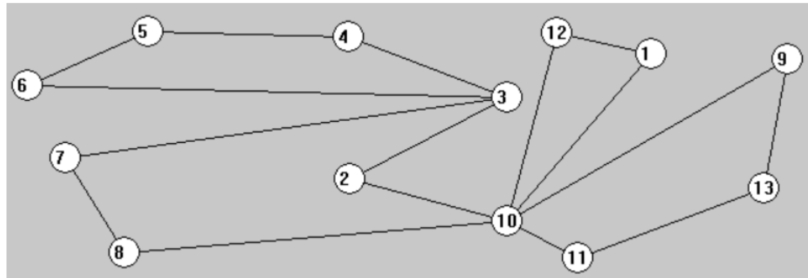


Definiție. Într-un graf G=(X,U) se numește **ciclu hamiltonian** un ciclu elementar care conține toate vârfurile grafului.

Definiție. Se numește **graf hamiltonian** un graf care conține un ciclu hamiltonian.

Exemplu. În graful din figura alăturată, ciclul C=[1, 2, 3, 5, 4, 6, 7, 1] este hamiltonian.

Consecință. Dacă într-un graf cu n vârfuri ($n \geq 3$), gradul fiecărui vârf este cel puțin $n/2$, atunci graful este hamiltonian.



Definiție. Într-un graf G=(X,U) se numește **ciclu eulerian** un ciclu care conține toate muchiile grafului. Un graf care conține un ciclu eulerian se numește **graf eulerian**.

Exemplu. Graful din figura alăturată este eulerian deoarece el conține ciclul eulerian C=[1, 10, 2, 3, 4, 5, 6, 3, 7, 8, 10, 9, 13, 11, 10, 12, 1].

Consecință. Un graf eulerian **poate** să conțină vârfuri izolate.

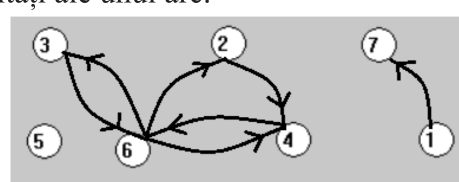
Teoremă. Un graf G=(X,U) care nu conține vârfuri izolate, este eulerian dacă și numai dacă este **conex** și gradele tuturor vârfurilor sunt **numere pare**.

Grafuri orientate

Definiție.Se numește **graf orientat** o pereche de două mulțimi (X, U) unde X este o mulțime finită și nevidă de elemente numite **noduri** sau **vârfuri**, iar U este o mulțime de perechi **ordonate** de elemente din X numite **arce.**

Consecință. Singura diferență dintre grafurile neorientate și cele orientate este că la grafurile orientate, perechile din U sunt ordonate.

Orice arc $u \in U$ va fi notat cu u=(x,y) și spunem că x este extremitatea inițială iar y este extremitatea finală a arcului. Arcul (x,y) este diferit de arcul (y,x). Graful orientat se poate reprezenta planar asociind vârfurile unor puncte iar arcele cu săgeți care să unească cele două extremități ale unui arc.



Exemplu. Graful G=(X,U) unde X={1, 2, 3, 4, 5, 6, 7} iar U={(3,6), (6,3), (6,2),(2,4), (6,4), (4,6), (1,7)} se poate reprezenta planar ca în figura alăturată.

Ca și în cazul grafurilor neorientate, dacă există arcul u=(x,y) spunem că vârfurile x și y sunt **adiacente** și amândouă sunt **incidente** cu arcul u. Se poate observa că pentru grafuri orientate sunt doup tipuri de arce incidente cu un vârf: arce „care intră” și arce „care ies” din vârf. Vom defini deci două concepte distincte care să corespundă numărului arcelor din fiecare dintre cele două categorii.

Teoremă. Numărul de grafuri orientate cu n vârfuri date X={1,2,…n} este $4^{C_n^2}$.

Definiție. Numim **grad exterior** al unui vârf x, notat cu $d^+(x)$, numărul arcelor de forma $(x,y) \in U$. Analog numim **grad interior** al unui vârf x, notat cu $d^-(x)$, numărul arcelor de forma $(y,x) \in U$.

Exemplu. În graful de mai sus: $d^+(4)=1$, $d^-(4)=2$, $d^+(6)=3$, $d^-(1)=0$, $d^-(5)=0$.

Definiție. Un **drum** într-un graf orientat G=(X,U) este o succesiune de vârfuri $D=(y_1, y_2, ..., y_k)$ cu proprietatea că $(y_1,y_2), (y_2,y_3),\dots(y_{k-1},y_k) \in U$ (sunt arce ale grafului).

Exemplu. În graful de mai sus D1=(6, 3, 6, 2, 4) și D2=(3, 6, 2, 4) sunt drumuri. Așadar un drum poate fi privit ca un lanț în care toate arcele au aceeași orientare dată de sensul de deplasare de la extremitatea inițială către cea finală.

Definiție. Numim **drum elementar** un drum cu proprietatea că oricare două vârfuri sunt distincte.

În exemplul de mai sus D2 este drum elementar iar D1 drum neelementar.

Definiție. Dacă D=(y_1, y_2, ..., y_k) este un drum cu proprietatea că y_1=y_k și oricare două arce sunt distincte, atunci drumul poartă numele de **circuit**. Dacă toate vârfurile circuitului, cu excepția primului și a ultimului, sunt distincte, atunci circuitul se numește **circuit elementar**.

Exemplu. În graful de mai sus C1=(6, 3, 6, 2, 4, 6) este un circuit neelementar iar C2=(6, 2, 4, 6) este un circuit elementar.

Observație. Noțiunile de **graf parțial** și **subgraf** la grafuri orientate se definesc analog grafurilor neorientate: graful parțial se obține eliminând o parte din arce iar subgraful, eliminând o parte din vârfuri și toate arcele incidente cu vârfurile eliminate. **Lungimea** unui drum sau circuit este analog asociată numărului de arce care alcătuiesc drumul sau circuitul respectiv.

Definiție. Spunem despre un graf orientat G=(X,U) că este **tare conex** dacă pentru oricare două vârfuri x și y din X există un drum de la x la y și un drum de la y la x (drum dus-întors de la x la y).

Graful din figura de mai sus nu este tare conex (de exemplu nu există drum de la 6 la 7).

Definiție. O **componentă tare conexă** a unui graf orientat G=(X,U) este un subgraf H=(Y,V) al lui G care este tare conex și maximal în raport cu această proprietate (adică subgraful nu se mai poate extinde prin adăugarea de vârfuri din X-Y și să își păstreze proprietatea de tare conexitate).

Exemplu. Graful din figura anterioară are 4 componente tare conexe. Prima este subgraful definit de vârfurile 2, 3, 4 și 6 iar celelalte trei au câte un vârf: 1, 5 și 7.

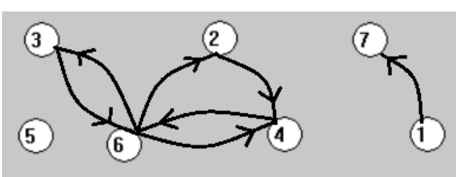
Reprezentarea grafurilor orientate în memorie.

Metodele descrise anterior la grafuri neorientate se definesc asemănător și la grafurile orientate.

Matricea de adiacență. Este o matrice pătratică și binară de ordinul n definită astfel:

`a[i][j]=1`, dacă (i,j) ∈U, respectiv `a[i][j]=0` dacă nu există arcul (i,j).

Din modul de definire a matricei de adiacență rezultă că ea **nu** este simetrică față de diagonala principală. În primele două figuri de mai jos avem un graf cu 7 vârfuri și matricea lui de adiacență.



Matrice adiacență

0	0	0	0	0	0	1
0	0	0	1	0	0	0
0	0	0	0	0	1	0
0	0	0	0	0	1	0
0	0	0	0	0	0	0
0	1	1	1	0	0	0
0	0	0	0	0	0	0

Vârf	Vecini
1	7
2	4
3	6
4	6
5	
6	2, 3, 4
7	

Liste de adiacență. Se precizează numărul n de vârfuri și pentru fiecare vârf i, lista L_i a vecinilor săi, adică lista vârfurilor j pentru care (i,j) ∈U.

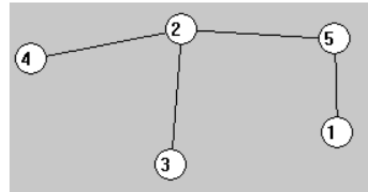
Alfel spus, pentru fiecare vârf x se creează o listă cu vârfurile la care se poate ajunge printr-un arc, plecând de la x. În cea de-a treia figură de mai sus este reprezentat graful folosind liste de adiacență.

Vector de arce. Se definește un tip ”arc” care să rețină două elemente de tip întreg (pentru extremitatea inițială a arcului, respectiv cea finală) apoi un vector unidimensional de tip „arc”. În limbajul C++ modalitatea este prin intermediul unei structuri:

```
struct arc { int x,y; } U[1000];
```

În declarația de mai sus tabloul U poate memora cel mult 1000 de arce cu extremitățile inițiale în x și cele finale în y (sau invers).

Arbori

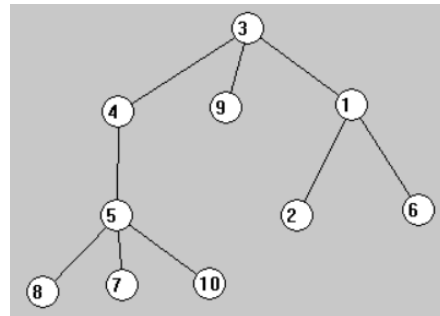


Definiție. Numim **arbore** un graf neorientat, conex și fără cicluri.

Figura alăturată reprezintă un arbore. Se observă că oricum am adăuga o muchie se va forma un ciclu și orice muchie s-ar elimina, se va pierde proprietatea de conexitate. Observația este valabilă pentru orice arbore.

Consecință. Un arbore cu n vârfuri are n-1 muchii.

Definiție. Numim **arbore cu rădăcină** un arbore în care vârfurile se pot distribui pe nivele începând cu un vârf oarecare numit rădăcină și considerat pe nivel 0. Vârfurile adiacente rădăcinii vor ocupa nivelul 1, cele adiacente cu acestea, nivelul 2, ș.a.m.d.



Observație. Oricare vârf al unui arbore se poate alega ca rădăcină pentru a distribui pe nivele celelalte vârfuri. În arborele din figura alăturată rădăcina este 3.

Definiție. Dacă x și y sunt două noduri adiacente într-un arbore cu rădăcină iar x este situat pe un nivel superior atunci y se va numi **fiu** sau **descendent direct** al lui x. Vârful x se va numi **tatăl** sau **părintele** lui y. Dacă două vârfuri y și z au același părinte atunci se vor numi frați. Un vârf care nu are fii se numește **vârf terminal** sau **frunză**.

Exemplu. În arborele cu rădăcină din figura alăturată vârful 4 este părintele (tatăl) lui 5 iar 5 este fiul (descendent direct) lui 4. Vârfurile 7, 8 și 10 sunt frați. Vârfurle 2, 6, 9, 7, 8 și 10 sunt frunze.

Definiție. Dacă există un lanț între nodurile x și y dintr-un arbore cu rădăcină iar x este situat pe un nivel superior atunci y se va numi **descendent** al lui x, situație în care se poate spune și despre x că este **ascendentul** lui y.

Exemplu. În arborele cu rădăcină din figura de mai sus vârful 4 are ca descendenți vârfurile 5, 8, 7 și 10. Rădăcina 3 are ca descendenți toate vârfurile arborelui și este singurul vârf care nu are părinte.

Definiție. Numim **înălțime** a unui arbore cu rădăcină, lungimea (numărul de muchii) celui mai mare lanț care leagă rădăcina de o frunză. De exemplu, arborele din figură are înălțimea 3.

Metode de reprezentare.

Orice metodă de reprezentare în memorie a unui graf neorientat se poate aplica și arborilor deoarece arborele este un graf. Atunci când arborele este transformat într-unul cu rădăcină, se pot folosi metode specifice.

Vector de tați. Se folosește un vector cu n elemente (pentru un arbore cu n vârfuri) care reține tatăl fiecărui vârf. Pentru rădăcină, prin convenție se folosește valoarea 0. De exemplu, pentru arborele din figură obținem următoarele valori:

tata	3	1	0	3	4	1	5	5	3	5
vârf	1	2	3	4	5	6	7	8	9	10

Liste de descendenți. Pentru fiecare vârf se construiește câte o listă cu descendenții direcți (fii). Pentru frunze, listele vor fi evident vide. De exemplu, pentru arborele din figură, L(4)={5}, L(5)={7, 8, 10}, L(2)=Ø iar L(3)={1, 4, 9}.

Marin Cristea
Colegiul Național ”Ion C. Brătianu”