Grafuri – noţiuni teoretice din programa pentru BAC

1. Grafuri neorientate

Cerinţele programei pentru BAC:

- terminologie (nod/vârf, muchie, adiacenţă, incidenţă, grad, lanţ, ciclu, lungime, subgraf, graf parţial)
- proprietăţi (regulat, complet, aciclic, conex, componentă conexă, hamiltonian, eulerian)
- metode de reprezentare în memorie (matrice de adiacenţă, liste de adiacenţă)

Definiţie: Un ***graf*** este o pereche ordonată de mulţimi, notată G=(X,U), unde
X={x|x∈X} este ***mulţimea nodurilor*** (vârfurilor) iar U={(x,y)| x,y∈X}, ***mulţimea muchiilor***.
nod/vârf = element al mulţimii X; poate fi reprezentat în plan printr–un punct (cerc etc.), eventual numerotat.
muchie = pereche neordonată de noduri; poate fi reprezentată în plan printr–un segment de dreaptă/arc
adiacenţă = proprietatea a două noduri de a fi unite prin muchie; dacă [x,y]∈U, spunem că nodurile x şi y sunt adiacente
incidenţă = proprietatea unei muchii de a uni două noduri; dacă [x,y]∈U, spunem că muchia este incidentă cu nodurile x şi y
gradul nodului x = numărul de muchii incidente cu nodul x, notat cu d(x)

- *nod izolat* = nod cu gradul 0; `d(x)=0`
- *nod terminal* = nod cu gradul 1; `d(x)=1`

Propoziţie: În orice graf neorientat cu *n* noduri şi *m* muchii, are loc egalitatea

```
2*m = d(x1)+d(x2)+...+d(xn)
```

(Suma gradelor varfurilor este dublul numarului de muchii)

Consecinta: In orice G exista un numar PAR de varfuri de graf IMPAR

lant = succesiune de noduri cu proprietatea că oricare două noduri consecutive din lanţ sunt adiacente

- *lanţ compus* = lanţ în care muchiile se pot repeta
- *lanţ simplu* = lanţ în care fiecare muchie apare o singură dată dar nodurile se pot repeta
- *lanţ elementar* = lanţ în care nodurile sunt distincte

ciclu = lanţ în care primul nod coincide cu ultimul

- *ciclu compus* = ciclu în care muchiile se pot repeta
- *ciclu simplu* = ciclu în care fiecare muchie apare o singură dată dar nodurile se pot repeta
- *ciclu elementar* = ciclu în care nodurile sunt distincte, cu excepţia primului şi ultimului nod

lungimea unui lanţ/ciclu = numărul de muchii din care este format

graf parţial = graf care se obţine din graful iniţial prin eliminarea unor muchii, nu şi a nodurilor

subgraf = graf care se obţine din graful iniţial prin eliminarea unor noduri şi a tuturor muchiilor incidente cu acestea; nu pot fi eliminate alte muchii decât cele incidente cu nodurile eliminate

Grafuri – noţiuni teoretice din programa pentru BAC

tipuri particulare de grafuri

graf regulat = graf în care toate nodurile au grade egale
graf complet = graf în care orice două noduri distincte sunt adiacente

- numărul de muchii într-un graf complet = n*(n-1)/2
- Numărul grafurilor neorientate cu n vârfuri este $2^{n(n-1)/2}$
- Numărul grafurilor partiale peste un graf cu M muchii 2^M

graf aciclic = graf în care nu există nici un ciclu
graf conex = oricare ar fi două noduri distincte, există lanţ între ele
componentă conexă = un subgraf conex şi maximal în raport cu această proprietate
(nu există lanţ între un nod din subgraf şi un nod care nu aparţine subgrafului)

Obs: un nod izolat constituie o componentă conexă

ciclu hamiltonian = ciclu elementar care trece prin **toate** vârfurile grafului
graf hamiltonian = graf care conţine cel puţin un ciclu hamiltonian

*Condiţie **suficientă** de existenţă a unui ciclu hamiltonian:*
Un graf neorientat cu **n** vârfuri, în care gradul oricărui vârf este mai mare sau egal cu **n/2** este hamiltonian.

ciclu eulerian = ciclu care trece prin toate muchiile grafului

graf eulerian = graf care conţine un ciclu eulerian

*Condiţie **necesară şi suficientă** de existenţă a unui ciclu eulerian*
Th. lui Dirac: Un graf fără vârfuri izolate, este **eulerian** dacă şi numai dacă
- este **conex**
- **gradele** tuturor vârfurilor sale sunt **pare**

OBS: Un graf complet cu numar **impar** de varfuri este hamiltonian si eulerian
Un graf complet cu numar **par** de varfuri este hamiltonian si NU este eulerian (ar avea gradele toate impare) => elimin MINIM n/2 muchii si poate deveni eulerian

Graf hamiltonian si eulerian: un poligon
Graf hamiltonian si NU eulerian: un poligon cu o diagonala
Graf NU hamiltonian si eulerian: o fundita cu 5 noduri (unul e in mijloc)
Graf NU hamiltonian si NU eulerian: ciclu neelementar si grade impare

Metode de reprezentare a grafurilor neorientate în memorie

Matricea de adiacenţă
- matricea este simetrica fata de diagonala principala
- gradul unui nod `d(x)=numarul de valori 1 de pe linia/coloana x`

2. Listele de adiacenţă

```
L1: 2,4
L2: 1,3,4
L3: 2,5
L4: 1,2
L5: 3,6
L6: 5
```

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

$L_i = \{j \in X / [i,j] \in U\}$

3. Lista de muchii

$t \in M_{2xm}$, unde m= numărul de muchii din graf

$t_{1,k}$ şi $t_{2,k}$ = extremităţile muchiei k

```
2 4 4 3 5 6
1 1 2 2 3 5
```

2. Grafuri orientate

Cerinţele programei pentru BAC:

- terminologie (nod/vârf, muchie, adiacenţă, incidenţă, grad intern si extern, drum, circuit, lungime, subgraf, graf parţial)
- proprietăţi (tare conex, componentă tare conexă)
- metode de reprezentare în memorie (matrice de adiacenţă, liste de adiacenţă)

Definiţie: Un ***graf orientat*** este o pereche ordonată de mulţimi, notată G=(X,U), unde X={x|x∈ X} este ***mulţimea nodurilor*** (vârfurilor) iar U={(x,y)| x,y∈ X}, ***mulţimea arcelor***.

nod/vârf = element al mulţimii X; poate fi reprezentat în plan printr–un punct (cerc etc.), eventual numerotat.

arc = pereche ordonată de noduri; poate fi reprezentată în plan printr–o sageata orientata

adiacenţă = proprietatea a două noduri de a fi unite prin arc; dacă (x,y)∈ U, spunem că nodurile x şi y sunt adiacente

incidenţă = proprietatea unei arc de a uni două noduri; dacă (x,y)∈ U, spunem că arcul este incident cu nodul x .

gradul intern al nodului x = numărul de arce care intra in nodul x, notat cu $d^-(x)$ **gradul extern al nodului x** = numărul de arce care ies din nodul x, notat cu $d^+(x)$

- *nod izolat* = nod cu gradul intern si extern 0; $d^-(x)= d^+(x) =0$
- *Propoziţie:* În orice graf orientat cu *n* noduri şi *m* arce, are loc egalitatea

Suma gradelor interioare = suma gradelor exterioare = numarul de arce

drum = succesiune de noduri cu proprietatea că oricare două noduri consecutive sunt adiacente (arcele pastreaza aceeasi orientare)

- *drum simplu* = drum în care fiecare arc apare o singură dată dar nodurile se pot repeta
- *drum elementar* = drum în care nodurile sunt distincte

circuit = drum în care primul nod coincide cu ultimul
- *circuit simplu* = circuit în care fiecare arc apare o singură dată dar nodurile se pot repeta
- *circuit elementar* = circuit în care nodurile sunt distincte, cu excepţia primului şi ultimului nod

lungimea unui drum/circuit = numărul de arce din care este format

graf parţial = graf care se obţine din graful iniţial prin eliminarea unor arce, nu şi a nodurilor

subgraf = graf care se obţine din graful iniţial prin eliminarea unor noduri şi a tuturor arcelor care au o extremitate in nodurile eliminate; nu pot fi eliminate alte arce decât cele cu legatura cu nodurile eliminate

Numărul grafurilor orientate cu n vârfuri este $2^{n(n-1)/2}$

tipuri particulare de grafuri

graf complet = graf în care orice două noduri distincte sunt adiacente (nu este unic, numarul de arce este cel mult n*(n-1)

graf plin = graf în care intre orice două noduri distincte x si y exista arc dus-întors (x, y) si (y, x)

Propoziţie: numărul de arce într-un graf plin = **n*(n-1)**

OBS: Numărul grafurilor orientate cu n vârfuri este $2^{n(n-1)} = 4^{n(n-1)/2}$
Numărul grafurilor orientate COMPLETE cu n vârfuri (exista cel putin un arc intre oricare 2 noduri) este $3^{n(n-1)/2}$

graf tare conex = oricare ar fi două noduri distincte x si y, există drum DUS-INTORS de la x la y

componentă tare conexă = un subgraf tare conex şi maximal în raport cu această proprietate (nu există drum între un nod din subgraf şi un nod care nu aparţine subgrafului)

Obs: un nod izolat constituie o componentă tare conexă

Metode de reprezentare a grafurilor orientate în memorie

1. Matricea de adiacenţă

- matricea nu este simetrica fata de diagonala principala
- gardul exterior = numarul de valori 1 de pe linia x
- gradul interior = numarul de valori 1 de pe coloana x

2. Listele de adiacenţă

`L i = {j∈X/ (i,j) ∈U}`

3. Lista de arce

$t \in M_{2xm}$, unde m= numărul de arce din graf
$t_{1,k}$ şi $t_{2,k}$ = extremităţile arcului k

```
L1: 2,4 L4: 1,2
L2: 1,3,4 L5: 3,6
L3: 2,5 L6: 5
```

3. Arbori

Cerinţele programei pentru BAC:

- terminologie (nod/vârf, muchie, radacina, descendent, descendent direct/fiu, ascendent, ascendent direct/parinte, frati, nod terminal, frunza)
- metode de reprezentare în memorie (matrice de adiacenţă, liste de descendenti, vectori de tati)

Definiţie: Un ***arbore*** este un graf conex aciclic.

Teorema de caracterizare:
Urmatoarele afirmatii sunt echivalente:
1. A este arbore cu **n** varfuri
2. A este conex cu **n-1** muchii
3. A este aciclic cu **n-1** muchii
4. A este conex minimal (daca se elimina o muchie se distruge conexitatea)
5. A este aciclic maximal (daca se adauga o muchie se formeaza un ciclu)

Proprietate: Oricare ar fi doua noduri distincte in arbore exista un lant elementar **unic** intre ele.

Definiţie: Un ***arbore cu radacina*** este un arbore in care exista un nod special numit ***radacina*** iar toate celelalte noduri reprezinta descendenti directi sau indirecti ai radacinii.

descendent al nodului x = nod care se afla pe un lant elementar ce pleaca din x, altul decat cel care uneste radacina de x.

Fiu/descendent direct al nodului x = descendent al nodului x adiacent cu x (nod adiacent cu x care nu se afla pe lantul care uneste radacina de nodul x)

ascendent al nodului x = nod care se afla pe lantul elementar care uneste radacina de nodul x.

Parinte/tata/ascendent direct al nodului x = ascendent al nodului x adiacent cu x.

Frunza/terminal = nod care nu are descendenti (are gradul 1)

Adancime = lungimea lantului elementar maximal care uneste radacina cu o frunza

Arbore degenerat = arbore in care orice nod care nu este terminal are exact un descendent direct/fiu.

Arbore binar – arbore vid sau arbore în care orice nod are cel mult doi fii, intre care se face distinctie clara, fiu stang, fiu drept.

Inaltimea arborelui = numarul de muchii a lantului maxim de la radacina la o frunza

In orice arbore exista un lant elementar unic intre oricare doua varfuri

Metode de reprezentare a arborilor în memorie

Matricea de adiacenţă

1. Reprezentare ascendenta in arbori cu radacina - EFICIENTA

Tata(n), $tata_i$ = Parintele nodului i daca i ≠ radacina; 0, altfel

ex: rad=2, tata = (2,0,2,2,3,5)

2. Reprezentare descendenta in arbori cu radacina

fiu(n), L_i =lista fiilor nodului i, i ≠ frunza; 0, altfel

ex: rad=2,