FIȘĂ DE LUCRU · CLASA A XI-A · FIȘA 5 · ACOPERĂ LECȚIILE XI.9–XI.10

Grafuri neorientate

Durată: 100 min Nivel: mate-info

CE VEI ȘTI DUPĂ LECȚIE

- Citești un graf din matrice sau din lista de muchii și afli rapid grade, muchii, componente.
- Rezolvi itemii de BAC cu formule: muchii de adăugat/eliminat, grafuri parțiale, subgrafuri, șiruri de grade.
- Recunoști și testezi în cod grafurile conexe, bipartite, euleriene și hamiltoniene.

Ce trebuie să știi (o pagină)

Regula	Exemplu	Rezultat
Σ grade = 2m; vârfurile de grad impar sunt în număr **par**	grade 3, 3, 2, 2, 2	m = 6
maxim de muchii (K_n): n(n − 1)/2	n = 7	21
grafuri distincte cu n vârfuri etichetate: 2^(n(n−1)/2)	n = 4	64
grafuri **parțiale**: 2^m · **subgrafuri**: $2^n - 1$	m = 5 · n = 4	32 · 15
matrice: simetrică, diagonala 0, grad(i) = suma liniei i	opt valori de 1	m = 4
p componente: ≥ n − p muchii; ca să fie conex adaugi **p − 1**	n = 10, p = 3	≥ 7; adaugi 2
maxim de muchii cu p componente: p − 1 izolate + K_{n-p+1}	n = 8, p = 2	21
ca să nu mai aibă cicluri, elimini m − (n − p) muchii	K_5	elimini 6
regulat de grad r: n·r/2 muchii · $K_{p,q}$: p·q muchii	$K_{3,4}$	12
bipartit ⇔ fără cicluri de lungime **impară**	triunghi	nu e bipartit
eulerian ⇔ conex (fără izolate) + toate gradele pare	exact 2 impare	doar lanț eulerian
hamiltonian: ciclu elementar prin **toate vârfurile**	K_n: (n − 1)!/2 cicluri	K_5: 12

ATENȚIE — GREȘELI FRECVENTE

Parțial = păstrez toate vârfurile, scot muchii. **Sub**graf = scot vârfuri, iar muchiile lor dispar odată cu ele. **E**uler = muchii, **H**amilton = vârfuri.

Nivelul 1 · Prezice

★ citește și prezice

Graful **B** („funda"), folosit în mai multe probleme: n = 6, muchiile [1,2], [1,3], [2,3], [3,4], [4,5], [4,6], [5,6].

P1. Graful este dat prin matricea de adiacență. Completează gradele, apoi răspunde: câte muchii are? Care vârf e izolat, care e terminal? Câte componente conexe are? Câte muchii îi lipsesc ca să fie complet?

	1	2	3	4	5	6
1	0	1	0	1	0	0
2	1	0	0	1	1	0
3	0	0	0	0	0	0
4	1	1	0	0	1	0
5	0	1	0	1	0	1
6	0	0	0	0	1	0
grad						

P2. Ce afișează secvența (graful B)? Ce îți spune ultima valoare despre un traseu care trece o singură dată pe fiecare muchie?

```
int n = 6, m = 7;
int e[7][2] = {{1,2},{1,3},{2,3},{3,4},{4,5},{4,6},{5,6}};
int g[7] = {0};
for (int k = 0; k < m; k++) { g[e[k][0]]++; g[e[k][1]]++; }
int t = 0;
for (int i = 1; i <= n; i++) if (g[i] % 2) t++;
cout << g[3] << ' ' << g[4] << ' ' << t;
```

P3. (BAC) Care dintre șiruri pot fi gradele unui graf cu 5 vârfuri? Pentru fiecare „nu”, spune motivul în cel mult o propoziție.

a) 4 4 3 2 1	**b) 3 3 3 3 3**	**c) 4 3 2 2 1**	**d) 4 3 2 1 0**	**e) 3 3 2 1 1**

P4. Completează tabelul, fără desen.

Întrebare	Răspuns	Întrebare	Răspuns
muchii maxim, n = 8		vârfurile unui graf complet cu 28 de muchii	
grafuri distincte cu 3 vârfuri		muchii, regulat de grad 4 cu 9 vârfuri	
grafuri parțiale, m = 6		n = 9, 3 componente: muchii minim / maxim	
subgrafuri, n = 5			

P5. Graful H este dat prin listele de adiacență. Este hamiltonian? Eulerian? Are lanț eulerian (între ce vârfuri)? Este bipartit (dacă da, scrie cele două mulțimi)?

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

P6. În graful B, scrie **toate** lanțurile elementare de la 1 la 6. Care e cel mai lung? Câte cicluri elementare are B?

P7. Ce returnează funcția pentru graful B? Dar pentru K_5? Spune **în cuvinte** ce numără.

```
int f(int a[][7], int n) {
    int c = 0;
    for (int i = 1; i <= n; i++)
        for (int j = i + 1; j <= n; j++)
            for (int k = j + 1; k <= n; k++)
                if (a[i][j] && a[j][k] && a[i][k]) c++;
    return c;
}
```

P8. (BAC) Graful are n = 8 și muchiile [1,2], [2,3], [3,1], [4,5], [6,7], [7,8], [8,6]. Câte muchii trebuie **adăugate** minim ca să fie conex? Câte trebuie **eliminate** minim ca să nu aibă cicluri? Câte operații îți trebuie ca să devină arbore?

Nivelul 2 · Găsește și repară

★★ găsește și repară

P9. Pentru n = 4 și muchiile [1,2], [1,3], [2,3], [3,4] programul afișează gradele `2 1 1 0`, dar corect este `2 2 3 1`. Fără să vezi codul: ce rând de la citire a fost scris greșit? Scrie-l corect.

P10. Funcția de mai jos spune dacă un graf (fără vârfuri izolate) este eulerian. Dă un graf cu 6 vârfuri pentru care răspunde greșit „DA". Ce trebuie adăugat?

```
bool eulerian() {
    for (int i = 1; i <= n; i++)
        if (grad[i] % 2 == 1) return false;
    return true;
}
```

P11. *Puzzle cu rânduri amestecate.* Pune în ordine rândurile funcției care colorează graful cu culorile 1 și 2 (testul de bipartit). Un rând este în plus.

```
A)  if (col[y] == 0) coloreaza(y, 3 - c);
B)  col[x] = c;
C)  void coloreaza(int x, int c) {
D)  else if (col[y] == c) ok = false;
E)  for (int y = 1; y <= n; y++) if (a[x][y]) {
F)  } }
G)  else if (col[y] != c) ok = false;
```

P12. Completează golurile. a) `b` (aceleași n vârfuri) este graf parțial al lui `a`; b) câte muchii îi lipsesc lui `a` ca să fie complet.

```
// a)
for (int i = 1; i <= n; i++)
    for (int j = 1; j <= n; j++)
        if (b[i][j] == 1 && __________) return false;
return true;
// b)
int lipsa = 0;
for (int i = 1; i <= n; i++)
    for (int j = ______; j <= n; j++)
        if (__________) lipsa++;
```

P13. *Ce greșeală a făcut elevul?* Pentru fiecare afirmație, spune de ce e falsă și dă răspunsul corect sau un contraexemplu.

a. „$K_{3,3}$ este eulerian: toate vârfurile au același grad."

b. „Un graf cu 12 vârfuri și 3 componente conexe are cel mult (12 − 1)(12 − 2)/2 = 55 de muchii."

c. „Graful nu are niciun triunghi, deci este bipartit."

P14. Programul trebuie să afișeze vârfurile izolate. Pentru n = 5 și muchiile [2,3], [3,4] afișează doar `1`. De ce? Repară mutând un singur rând.

```
int grad = 0;
for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= n; j++) grad += a[i][j];
    if (grad == 0) cout << i << ' ';
}
```

Nivelul 3 · Scrie cod

★★ scrie cod

Toate programele citesc `n m` și apoi `m` muchii. Verifică **toate** testele.

P15. Cel mai popular. Într-o rețea de prieteni, afișează pe primul rând elevii cu cei mai mulți prieteni (crescător), iar pe al doilea, câți elevi nu au niciun prieten.

Teste: `5 4` · `1 2` `1 3` `2 3` `3 4` → `3` / `1`; `4 0` → `1 2 3 4` / `4`; `4 3` · `1 2` `3 4` `2 3` → `2 3` / `0`.

P16. Traseul poștașului. Poștașul vrea să treacă pe fiecare stradă (muchie) o singură dată. Afișează `ciclu` dacă se poate și să revină de unde a plecat, `lant x y` dacă se poate doar pornind din x și terminând în y (x < y), altfel `nu`. Atenție: intersecțiile fără străzi (izolate) nu contează; m ≥ 1.

Teste: `4 3` · `1 2` `2 3` `3 1` → `ciclu`; graful B → `lant 3 4`; `6 6` · `1 2` `2 3` `1 3` `4 5` `5 6` `4 6` → `nu`; `4 3` · `1 2` `1 3` `1 4` → `nu`.

P17. Cablurile. Calculatoarele sunt vârfuri, cablurile sunt muchii. Afișează câte cabluri noi trebuie trase minim ca toate să comunice și câte cabluri sunt de prisos (pot fi scoase fără ca vreun calculator să piardă legătura cu altul).

Teste: graful de la P8 → `2 2`; `5 4` · `1 2` `2 3` `3 4` `4 5` → `0 0`; K_4 (`4 6`, toate muchiile) → `0 3`; `6 0` → `5 0`.

P18. Două echipe. Muchiile sunt perechi de elevi certați. Împarte elevii în două echipe fără ca doi certați să fie în aceeași echipă. Elevul 1 merge în echipa A; din fiecare grup încă neașezat, cel mai mic elev merge tot în A. Afișează echipa A crescător sau `imposibil`.

Teste: `4 4` · `1 2` `2 3` `3 4` `4 1` → `1 3`; `3 3` · `1 2` `2 3` `1 3` → `imposibil`; `5 2` · `1 2` `4 5` → `1 3 4`.

P19. Verificatorul de trasee. După graf se citește `k` și un șir de k vârfuri. Afișează `ciclu elementar`, `lant elementar`, `lant` (un vârf se repetă) sau `nu` (lipsește o muchie). Teste pe graful B:

șir	ieșire	șir	ieșire
`1 2 3 4 5`	`lant elementar`	`1 2 4`	`nu`
`3 4 5 6 4`	`lant`	`1 2 1`	`lant`

șir	ieșire	șir	ieșire
`4 5 6 4`	`ciclu elementar`		

P20. Turneul de șah. Fiecare joacă o dată cu fiecare. Muchiile sunt partidele deja jucate. Afișează câte partide mai sunt, apoi jucătorii care și-au terminat toate partidele (sau `niciunul`).

Teste: `4 4` · `1 2` `1 3` `1 4` `2 3` → `2` / `1`; `3 3` · `1 2` `2 3` `1 3` → `0` / `1 2 3`; `5 0` → `10` / `niciunul`.

Nivelul 4 · Pune mintea la contribuție

★★★ pune mintea la contribuție

P21. Eulerian, dar nu hamiltonian. Construiește un graf cu 6 vârfuri, fără vârfuri izolate, care este eulerian, dar nu hamiltonian, și are **cât mai multe** muchii. Câte muchii poți atinge? De ce nu are ciclu hamiltonian?

P22. Strângeri de mână. La o petrecere sunt 9 persoane. Poate fiecare să fi dat mâna cu exact 5 dintre celelalte? Dar dacă sunt 10 persoane? Justifică doar cu suma gradelor și dă o construcție pentru cazul posibil.

P23. Contraexemplu. Un elev a notat: „cu n vârfuri și m muchii, numărul **maxim** de componente conexe este n − m". Verifică pe n = 20, m = 10: poți obține mai mult de 10 componente? Care este maximul real? Ce reprezintă de fapt n − m?

P24. Câte grafuri parțiale rămân conexe? Graful B are $2^7 = 128$ de grafuri parțiale. Câte dintre ele sunt conexe? Câte sunt arbori? Găsește răspunsul **prin raționament** (ce muchie nu poate lipsi niciodată?), apoi verifică-l cu un program care încearcă toate cele 128 de submulțimi.

P25. Numărătorul de cicluri hamiltoniene. Scrie un program de backtracking care numără ciclurile hamiltoniene ale unui graf dat prin matrice, fiecare ciclu numărat **o singură dată** (indiciu: pornește mereu din 1 și cere `x[2] < x[n]`). Teste: K_4 → 3, K_5 → 12, graful H de la P5 → 1.

Probleme de lucru în clasă (pbinfo)

Le rezolvăm împreună în ședință, în ordinea de mai jos.

- [ ] #412 **adiacenta** — ușoară
  https://www.pbinfo.ro/probleme/412/adiacenta
- [ ] #416 **grade** — ușoară
  https://www.pbinfo.ro/probleme/416/grade
- [ ] #437 **conex** — ușoară
  https://www.pbinfo.ro/probleme/437/conex
- [ ] #438 **componenteconexe** — ușoară
  https://www.pbinfo.ro/probleme/438/componenteconexe

Temă pentru acasă (pbinfo)

Trimite soluțiile pe pbinfo și notează punctajul obținut. Bifează ce ai terminat.

- [ ] #417 **gradmax** — ușoară
  https://www.pbinfo.ro/probleme/417/gradmax
- [ ] #420 **graf-partial** — ușoară
  https://www.pbinfo.ro/probleme/420/graf-partial
- [ ] #471 **bipartit** — ușoară
  https://www.pbinfo.ro/probleme/471/bipartit
- [ ] #441 **componenteconexe1** — medie
  https://www.pbinfo.ro/probleme/441/componenteconexe1
- [ ] #545 **euler** — medie
  https://www.pbinfo.ro/probleme/545/euler
